Australian Government
Department of Defence
Defence Science and
Technology Organisation

A Survey of Electronics Obsolescence and Reliability

R. J. O'Dowd

Air Operations Division
Defence Science and Technology Organisation

DSTO-TR-2437

ABSTRACT

The service life of military assets significantly exceeds design life of commercial electronic systems used within them. Electronic obsolescence is increasingly associated with physical characteristics that reduce component and system reliability, both in usage and storage, with few design margins outside commercial warranty periods. Software content, however, remains a dominant limiting factor for reliability of electronic systems, and emerging commercial trends compound this. Traditional approaches to manage and sustain electronic systems are therefore increasingly ineffective and costly. This report surveys the interrelated concerns of obsolescence and reliability of electronic systems, and describes emerging responses to these concerns.

Published by

Air Operations Division
DSTO Defence Science and Technology Organisation
506 Lorimer St
Fishermans Bend, Victoria 3207 Australia

Telephone: (03) 9626 7000
Fax: (03) 9626 7999

AR-014-805
July 2010

APPROVED FOR PUBLIC RELEASE

A Survey of Electronics Obsolescence and Reliability

Executive Summary

The service life of military capabilities significantly exceeds the design life of commercial electronic systems that implement key functions. As the electronics industry continues to invest in regularly increasing functionality while reducing physical size, obsolescence is also associated with unavoidable physical phenomena and effects that reduce reliability of miniaturised semiconductor technologies and of electronic systems, both in usage and in storage. By design, electronics technologies include few reliability margins outside commercial warranty periods. Highly miniaturised electronics increasingly have failure characteristics in the form of intermittent faults or other counter-intuitive behaviours as components degrade, rather than exhibiting obvious failures. System reliability also reduces with system complexity and software remains a main limiting factor on reliability of electronic systems through life, due to the difficulty of getting software right. Commercial trends towards multicore processors will compound this concern, due to a need to rearchitect software to get desired performance with multicore processors.

The interrelated concerns of obsolescence and reliability affect all commercially available electronic systems. These concerns range on a spectrum between the natural implications of using cutting-edge technologies (immature technology is rarely highly reliable) through to planned obsolescence, in which components are deliberately designed to last only through their warranty period and customers are obligated to buy again. Regardless of cause, Defence bears the impact of increased cost to sustain a viable capability due to a need to implement significant system updates or replacement to ensure long-term viability and affordability of most military capabilities that depend on electronic systems. Simple responses like retiring old capabilities and acquiring new simply magnify the effects because the rate of obsolescence is likely to accelerate, warranty periods are unlikely to increase, manufacturers continue to reduce reliability margins outside the warranty period.

This report surveys some of the inter-related concerns of basic electronics, electronic system reliability, obsolescence, software reliability, the effects these have through long service lives typical of military applications, the limitations of traditional logistical responses (e.g. last-time buys), and some emerging responses to these concerns. The technical emphasis is slightly towards embedded computing systems on aircraft, but discussion is applicable to any military capability that relies on some electronic system. The purpose is to provide some basis for discussion of potential coordinated responses, noting that there are many technical and non-technical factors involved. The current Defence Capability Plan (DCP) includes several projects that seek to address obsolescence, and a significant number that specifically mention concerns related to electronic obsolescence or reliability. Commercial trends are likely to increase the impact of these concerns on the capability development process.

Author

R. J. O'Dowd
Air Operations Division

Robert O'Dowd completed a PhD in Geophysics at the University of Adelaide in 1990, and commenced employment with DSTO. He has worked in various areas, including prediction of tactical towed array performance, tactical aids for submarine command teams, undersea warfare operations research, surface combatant combat systems, and support of acquisition projects. He currently works in the airborne mission systems area.

Contents

1. Introduction

Defence acquisition programs traditionally relied on assurances associated with Military Standards to more easily meet requirements for extended operating range, quality and reliability, and could achieve economic advantages of reduced parts inventories and better quantity discounts if military grade components were used across multiple programs. A number of legal and policy changes in the US led to the "Perry Initiative" of 1994 which mandated that Military Standards only be used if an adequate commercial specification was unavailable. This increased usage of Commercial-Off-The-Shelf (COTS) products within military acquisitions. Most electronic component manufacturers then reviewed viability of low-volume component production lines and switched capacity to more lucrative higher volume commercial projects. Many commercial products offer adequate alternatives to military specifications, in the sense of offering a similar or greater feature set, but are often not designed for the range of environmental conditions associated with military application, so have lower shelf lives or operational lives, and shorter manufacture cycles.

Regearing of component manufacturers to high volume consumer products (mobile phones, games consoles, home computing, digital video products, etc) has substantially reduced Defence market share and influence. In practice, complex analogue and discrete semiconductor devices are very difficult to replace, as original technologies and geometries are obsolete, documentation is incomplete, and methods used to characterise components do not map well into modern practice, making it difficult to even identify a suitable replacement part. Replacement devices also exhibit characteristics that necessitate different integration approach and setup procedures or can cause new instabilities of the original system. Reactive responses to these concerns, such as "last time buy" of components, purchasing original dies or wafers before components are removed from standard parts catalogues, or integration of new components to provide a similar function, are sometimes employed. Each has significant cost or time implications, related to quantities and, particularly in aircraft systems, re-qualification.

The MIL-STD-217 series, the original standard concerned with traditional reliability methods, included several caveats about its applicability. There is evidence that the "bathtub curve", traditionally employed to forecast component or system failure rates during their life cycle, is inapplicable to electronics. Some practitioners have proposed, based on analysis of available data sets, a multi-hump "roller-coaster" curve, suggesting the dominant causes of failures vary with time. The electronics industry is increasingly employing "Physics of Failure" techniques to minimise impacts of various physical phenomena during warranty periods. Some failure modes appear independent of usage rates, suggesting some physical mechanisms leading to failures remain active while components are in storage. Not withstanding these concerns, traditional techniques are easier to apply, so are used regularly and are often specified in acquisition contracts.

Industry is increasingly catering to consumer expectations that performance or features of computing equipment will improve with each generation, and an increasing tendency to treat electronic devices as disposable items. Consistent with this, industry increasingly follows the strategic economic practice of "value engineering", which seeks ongoing evolution that maximises value, with value specifically defined as a ratio of function to cost. All US Government departments, by law, are required to support and encourage industry practice of

value engineering. From a system engineering perspective, overall system quality attributes such as reliability and supportability are generally considered non-functional requirements, so are not typically primary considerations during value engineering processes. Value engineering is the underpinning of planned obsolescence, in which products increasingly become obsolete or non-functional after a time or usage selected in advance by the manufacturer. Consistent with the practice of value engineering, electronics manufacturers now routinely employ "Physics of Failure" techniques to minimise failures during warranty period, while reducing design margins that might contribute to reliability after warranty.

Technology projections suggest new generation electronic devices will continue to become smaller, faster, and consume less energy. These trends may have significant impacts on lifetime reliability because of scaling concerns, increased electric fields or power densities, increasing transistor count, increasing variability, and other design features.

Even with these physical concerns, software remains a dominant cause of unreliability of electronic systems. Development and verification of software remains difficult, and software is often used to implement functionality that cannot be effectively achieved in any other manner. Recent industry trends towards multicore processors place significant demands on software developers. Effective exploitation of multicore requires a shift away from traditional sequential software development techniques towards more difficult, rarely practiced, techniques for developing concurrent or parallel software.

After an overview of basic theory of electronics and of reliability methods, this report surveys a range of factors that affect both reliability and obsolescence characteristics of electronic components and electronic systems. Some suggested elements of a way forward that might be considered by the Australian Defence Organisation are presented.

2. Background

2.1 Basic electronics

Any electronic system may be described physically as a network of interconnected electrical elements that affect voltage or currents between those elements. An electrical circuit is a network with a closed loop that gives a return path for current.

The four fundamental properties that characterise any electrical network are **current**, I, **voltage**, V, **charge**, Q, and **magnetic flux**, Φ_m. These properties all vary with **time**, t, at any point in the network. All electronic networks may be represented using six abstract elements that affect these fundamental properties, as follows:

- A current source that produces a current I, measured in amperes, in a conductor;
- A voltage source that produces a potential difference, V, measured in volts, between two points;
- **Resistance**, R, measured in ohms, which produces a voltage proportional to the current flowing through an element;

- **Capacitance**, C , measured in farads, which produces a current proportional to the rate of change of voltage across the element (i.e. ability to hold electric charge);
- **Inductance**, L , measured in henries, which produces a voltage proportional to the rate of change of current through the element (i.e. a change in current induces an electromotive force (EMF) that opposes that change);
- **Memristance**, M , that produces a rate of change of current proportional to the rate of change of voltage across the element.

All of these elements, except memristance (§3.1.4), are the basis of traditional linear circuit theory. Relationships between the fundamental properties and abstract elements are;

$$I = -\frac{dQ}{dt}$$

$$V = \frac{d\Phi_m}{dt}$$

$$R = \frac{dV}{dI}$$

$$C = \frac{dQ}{dV}$$

$$L = \frac{d\Phi_m}{dI}$$

$$M = \frac{d\Phi_m}{dQ}$$

Gain is a measure of ability of a circuit or system to increase power or amplitude of a signal. It is usually defined as the mean ratio of the output signal to the input signal for a circuit or system, and often presented on a decimal logarithmic scale. The term "gain" in isolation is ambiguous, as it may refer to a ratio of voltage, power, or current increase.

A **passive component** consumes energy and is therefore incapable of power gain. Otherwise a component is described as **active**. A passive electronic circuit consists entirely of passive components, and has the same defining properties as a passive component.

Analogue electronics work with a continuously variable signal, while **digital** electronics typically involves only two different levels (on and off).

2.2 Systems engineering

Systems engineering is an interdisciplinary field of engineering concerned with design and management of complex engineering projects. Within this field, there are various definitions of "system", including:

- "A set or arrangement of elements and processes that are related and whose behaviour satisfies customer/operational needs and provides for life cycle sustainment of the products" [95];

- "An aggregation of end products and enabling products to achieve a given purpose" [107];
- "An assemblage or combination of things or parts forming a complex or unitary whole" [108];
- "A homogeneous entity that exhibits predefined behaviour in the real world and is composed of heterogeneous parts that do not individually exhibit that behaviour and an integrated configuration of components and/or subsystems" [255]
- "The combination of elements that function together to produce the capability to meet a need. The elements include all hardware, software, equipment, facilities, personnel, processes, and procedures needed for this purpose" [268];
- "A combination of interacting elements organised to achieve one or more stated purposes" [280].

Software engineering is the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, and the study of these approaches. The field is relatively young compared to other fields of engineering.

A **functional requirement** describes the set of system inputs, behaviours, and outputs required of a system or of its components. Functional requirements may be calculations, technical details, data manipulation and processing and other functionality that define *what* a system is to accomplish. The system **design** is a description of how functional requirements are implemented.

Non-functional requirements (also known as **quality requirements**) specify overall system characteristics, and impose constraints on the design or implementation. Examples of non-functional requirements include [291,330] usability, user-friendliness, flexibility, predictability, robustness, performance, interoperability, reliability, reparability, adaptability, understandability, enhanceability, supportability, and security.

2.2.1 Basic system concepts

This section is drawn from [177] and preceding articles [27,72].

The **function** of a system is what the system is intended to do and is described by the functional specification in terms of functionality and performance.

The **behaviour** of a system is what the system does to implement its function, and is described by a sequence of states.

The **total state** of a system comprises the set of states related to computation, communication, stored information, interconnection, and physical condition.

The **structure** of a system is what enables it to generate its behaviour and, structurally, a system is viewed as being composed of a set of components bound together in order to interact. This view is recursive, with components being further decomposable into contributing components and systems.

A component is considered **atomic** when its internal structure cannot be seen or recognised, or if that structure is not of interest, in the system design and can be ignored. The total state of a system is therefore the set of external states of its atomic components.

The **service** of a system is the set of helpful or useful functions it delivers [108]. Systems may have roles as a provider or as a receiver of service. A **user** is a system, or other entity, that receives service from a provider. The **service interface** is the part of a provider's system boundary where service delivery takes place, and the **user interface** is the part of the receiver's system boundary through which the user receives service. The part of provider's total state that may be perceived at the service interface is the **internal state**, and other parts are its **external state**.

A service **failure** is a transition from correct service to incorrect service, in which the service does not implement the system function. Service failure results in a deviation of the external state of the system from the correct service state. That deviation is referred to as an **error**, and the judged or hypothesised cause of an error is called a **fault**. Faults may be internal to or external of a system. A **vulnerability** is an internal fault that allows an external fault to harm the system. In practice, a fault first affects internal system state but does not necessarily have immediate effect on the external state.

Dependability is the ability to deliver service that can justifiably be trusted. It may also be defined as the ability to avoid service failures that are more frequent or errors more severe than acceptable. Dependability is an integrating concept that encompasses attributes of *availability* (readiness for correct service); *reliability* (continuity of correct service); *safety* (absence of catastrophic consequences on user(s) and environment); *integrity* (absence of improper system alterations); *maintainability* (ability to undergo modifications and repairs); and *confidentiality* (absence of unauthorised disclosure of information) [27,72,177,323].

Security is generally a composite of confidentiality, integrity, and availability. System properties that affect dependability and security include usability, manageability, and cost.

Computing systems are characterised by five fundamental properties: functionality, usability, performance, cost, and dependability. Failures, errors, and faults are the main categories of threats to dependability and security of systems.

2.2.2 Real-time systems

A **real-time** system is one in which correctness of behaviour depends on the time at which required effects are produced [157]. Typically a window of opportunity exists, and results must be produced in the interval between some start point and a deadline. Results produced too early or too late therefore represent errors, regardless of their logical correctness.

Periodic tasks are regularly repeated at some fixed interval, while **aperiodic** tasks respond to randomly arriving events. Such tasks are considered real-time if they must be initiated and completed within a specified time window after the triggering event.

A **hard real-time** system is one in which one or more activities must never violate a timing constraint, because violations of timing constraints may cause significant unwanted effects (equipment damage, loss of revenue, injury, death).

A **soft real-time** system is one that has timing requirements, but requirements of the application continue to be met if those requirements are missed occasionally. Systems that regularly sample some input may have soft real-time characteristics if they can withstand missing some samples but the likelihood of some failure increases with the number of missed samples.

A **non-real time** system is one in which only logical correctness of results is of concern, with no specified timing constraints.

A **predictable system** is one with timing behaviour that is always within an acceptable defined range. Requirements for predictability are specified on a system-wide basis, such as "all tasks will meet all deadlines". Generally, the period, deadline, and worst-case execution time of each contributing task needs to be known to create a predictable system. Predictability is ensured by use of an appropriate scheduling algorithm supported by a corresponding schedulability analysis.

A **deterministic system** is a special case of a predictable system, in which the timing behaviour can be explicitly pre-determined. A deterministic system is therefore one in which all tasks are executed only within pre-determined timeslots in a fixed schedule. Determinism requires that every task has a known execution time, and there are no timing anomalies that might cause deviation from pre-determined system requirements.

Avionics (airborne electronic) systems often have real-time characteristics [157]. Critical requirements, such as safety of flight, are often associated with hard real-time constraints of some key system tasks (e.g. engine control). Mission systems may have hard or soft real-time characteristics. Qualification or certification of such systems is typically assisted by greater technical rigour in the system design and certification process, so predictable or deterministic systems are more likely to be produced to meet more critical requirements.

2.3 System reliability

Reliability methods date from concerns in the 1950s by US military about reliability and readiness of electronic systems [101]. Two major approaches to reliability assessment are traditional methods, based upon probabilistic assessment of field data [176], and methods based on analysis of failure mechanisms and physics of failure [216]. Traditional methods were most commonly used to the 1970s and continue to be frequently used because they are easier to implement [176].

2.3.1 Traditional reliability concepts

Reliability is defined [46,102] as "the duration or probability of failure-free performance under stated conditions" or, alternatively, as "the probability that an item can perform its intended function for a specified interval under stated conditions" [117]. For non-redundant items these alternative definitions are equivalent. For redundant items, the second definition expresses mission reliability [117].

A **catastrophic failure** of a component occurs when it ceases to deliver required function, and repair or recovery is either impossible or does not contribute to completion of the intended mission [117]. Such failures are modelled based on life test data. The "lifetime" or "time to failure", T, is represented as a continuous random variable, such that;

$$P(survival\ to\ time\ t) = P(T > t) \equiv R(t)$$

where $R(t)$ is referred to as the **reliability function** and $R(t) \to 0$ as $t \to \infty$ since cumulative probability of failure increases with time of operation [117]. The probability of a failure having occurred at time t is then;

$$P(\ failure\ at\ t) = P(T \le t) \equiv Q(t) = 1 - R(t)$$

$Q(t)$, the **unreliability function**, is the distribution function for T [117]. The failure density function is;

$$f(t) = \frac{dQ(t)}{dt}$$

and the **hazard rate function** is;

$$\lambda(t) \equiv \lim_{\Delta t \to 0} \left[\frac{1}{\Delta t} \left(probability\ of\ \ failure\ in\ (t, t+\Delta t) : survival\ to\ time\ t \right) \right] = \frac{f(t)}{R(t)}$$

The four functions described above are fundamental tools of basic reliability analysis [117].

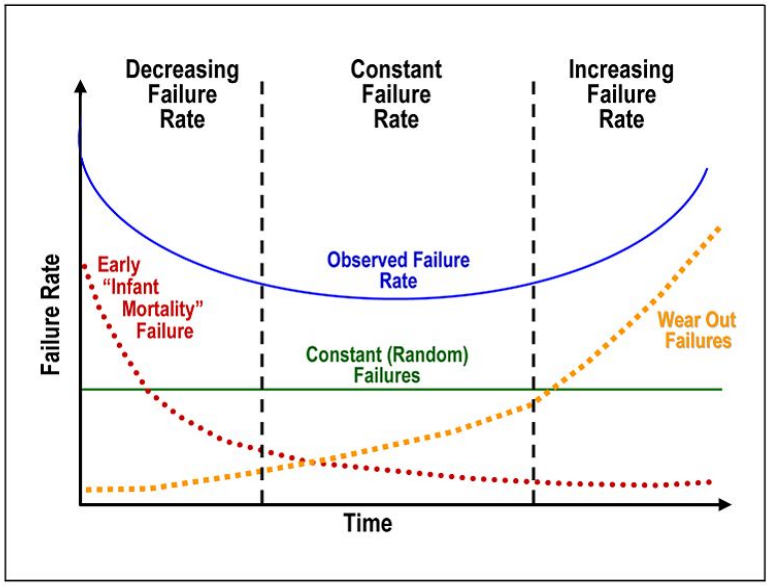

Figure 1: Bathtub-shaped hazard function

The hazard rate function is often plotted as a "bathtub curve", illustrated as the blue "Observed Failure Rate" curve in Figure 1. The "Decreasing Failure Rate" region corresponds to a "wear in" or "infant mortality" resulting from early failures during debugging [117]. The second "Constant Failure Rate" region is considered to consist of a low rate of essentially random

failures, during the useful life of the product [117]. The third "Increasing Failure Rate" region corresponds to a wearout or fatigue phase [117].

"**Burn-in**" is a practice of subjecting components to an initial operating period in order to reach the constant failure rate region before delivery to the customer [117]. This is considered to eliminate initial failures for customers requiring high reliability [117]. Similarly, component replacements are planned as they approach the wearout region. Traditionally, electronic components have been considered to have a long useful life (constant failure rate period) [117].

Measures of reliability include [117];

- The expected value of the continuous random variable "time to failure" is the Mean Time to Failure, $MTTF = \int_0^\infty tf(t)dt = \int_0^\infty R(t)dt$;
- Average Failure rate over interval T is, $AFR(0,T) \equiv AFR(T) = -\frac{\ln R(t)}{T}$;
- A Posteriori Failure probability is the probability of failure during the interval $(T, T+t)$, calculated as;

$$Q_c(t) = \frac{\int_T^{T+t} f(\xi)d\xi}{\int_T^\infty f(\xi)d\xi}$$

Derived or related quantities include [117];

- Mean Time to Repair ($MTTR$);
- Mean Time Between Failures[1] ($MTBF = MTTF + MTTR$). Since $MTTR$ is relatively small, $MTBF$ and $MTTF$ are often used interchangeably;
- Probability of survival in the interval $(T, T+t)$ is;

$$R(t \mid T) = 1 - Q_c(t) = \frac{\int_{T+t}^\infty f(\xi)d\xi}{\int_T^\infty f(\xi)d\xi} = \frac{R(T+t)}{R(T)} = e^{-\int_T^{T+t}\lambda(\xi)d\xi}$$

Distributions typically employed in traditional reliability analysis

The **Poisson distribution** is used to represent the probability $P_x(t)$ of exactly x occurrences in the time interval $(0,t)$ [117] and is given by;

$$P_x(t) = \frac{(\lambda t)^x e^{-\lambda t}}{x!}$$

[1] *MTBF* is sometimes misinterpreted as the life of a product whereas, in reality, it describes the mean number of total operating hours of a population before a failure occurs [133].

The **Weibull distribution**, with a scale parameter α and shape parameter β, are often used to fit a curve to experimental data in system reliability studies [117]. The representative functions are;

$$\lambda(t) = \frac{\beta t^{\beta-1}}{\alpha^{\beta}}; \alpha > 0, \beta > 0, t \geq 0$$

$$f(t) = \frac{\beta t^{\beta-1}}{\alpha^{\beta}} e^{-\left(\frac{t}{\alpha}\right)^{\beta}}$$

$$R(t) = e^{-\left(\frac{t}{\alpha}\right)^{\beta}}$$

And the associated mean is;

$$MTTF = \mu = \alpha\Gamma\left(1 + \frac{1}{\beta}\right)$$

The **Exponential distribution** is the particular case of the Poisson distribution for $X = 0$ or, alternatively, of the Weibull distribution for $\beta = 1, \lambda = 1/\alpha$ [117]. This is also known as the constant-hazard model, and is represented by;

$$f(t) = \lambda e^{-\lambda t}, t > 0$$

$$R(t) = e^{-\lambda t}$$

$$Q(t) = Q_c(t) = 1 - e^{-\lambda t}$$

The *a posteriori* failure probability $Q_c(t)$ is independent of prior operating time, T, reflecting an assumption that the component or system does not degrade during operation [117]. This assumption is considered to correspond to the Constant Failure Rate portion of the Bathtub Curve. The mean and standard deviation of the random variable "lifetime" are $\mu \equiv MTTF = \frac{1}{\lambda}$ and $\sigma = \frac{1}{\lambda}$ respectively.

The electronics industry often uses $MTBF$ as a measure of reliability, based on an assumption of an exponential failure distribution [133].

A random variable is **lognormally distributed** if its logarithm is normally distributed, as represented by;

$$f(T') = \frac{1}{\sigma_{T'}\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{T' - \bar{T}'}{\sigma_{T'}}\right)^2}, t > 0$$

$$R(T) = \int_{T'}^{\infty} \frac{1}{\sigma_{T'}\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{t - \bar{T}'}{\sigma_{T'}}\right)}$$

where T is the random variable representing time to failure, $T' = \ln(T)$, $\overline{T}'$ is the mean value of T' (i.e. the mean of the natural logarithms of time to failure), and $\sigma_{T'}$ is the standard deviation of natural logarithms of times to failure.

System analysis

Analysis of a system is facilitated by its decomposition into functional entities (subsystems or components) and by application of combinatorial considerations [117].

A serial or chain structure is one in which entities are linked in series, and the system only functions correctly if all elements in series also function correctly [117]. If R_i is probability of success of element i then, assuming independence between the elements, the serial system reliability of n elements is;

$$R_s = \prod_{i=1}^{n} R_i$$

A parallel structure is one in which the system will function correctly if any one of the n elements functions correctly [117]. If Q_i is probability of failure of element i then, assuming independence between elements, the system reliability of n parallel elements is;

$$R_p = 1 - \prod_{i=1}^{n} Q_i = 1 - \prod_{i=1}^{n} (1 - R_i)$$

2.3.2 Physics of Failure methods

Despite widespread usage of empirical methods, the Rome Air Development Center of USAF has documented [3,4] an interest in Physics of Failure since the 1960s. These methods study and correct the root cause of each individual failure mechanism in order to achieve required lifetime of electronic components [216].

The goal of reliability engineers using these techniques is to design electronic components so they perform with required life and with no single dominant failure mechanism [216]. Theoretically this means that wearout phenomena are unlikely to occur during service life of microelectronic devices, and reliability prediction techniques must allow for the presence of multiple competing failure mechanisms that potentially limit life of electronic devices [216]. In practice, statistical models of device reliability have become an integral part of the design process, with considerable advances made in understanding of the reliability physics of semiconductor devices, but there has been relatively little advance in techniques for failure rate qualification (e.g. accelerated testing), where a constant failure rate model is often still assumed [216].

During design of electronic components, physics of failure approaches involve [216];

- identifying potential failure mechanisms (chemical, electrical, physical, mechanical, structural, or thermal processes that lead to failure) and failure sites;

- identifying failure modes that result from activation of failure mechanisms. These are usually shorts (abnormal low-resistance connections), opens (abnormal high or infinite-resistance connections), or electrical deviations beyond specifications;
- identifying appropriate failure models and their input parameters. These may represent material characteristics, damage properties, relevant geometry at failure sites, manufacturing flaws and defects, and environmental and operating loads;
- determining distribution functions for each design parameter, if possible;
- computing the effective reliability function;
- identify and modify the reliability inhibitors to make the design acceptable; and
- accepting the design, if the reliability function meets or exceeds the required value over the required time period.

Physics-of-failure analysis seeks to determine or predict when a specific failure mechanism will occur for each component in a specific application. This analysis requires knowledge of all material characteristics, geometries, and environmental conditions [216].

The advantage of physics-of-failure approaches is that they allow accurate predictions based on known failure mechanisms. The disadvantages are that considerable knowledge is required of materials, processes, and failure mechanisms and there is a need for access to manufacturer material, process, and design data – such data is often unavailable to system designers or integrators [216].

2.3.3 Software reliability

Software reliability is defined as "the probability of failure free software operation for a specified period of time in a specified environment" [46,76,225,237]. Other software quality attributes include functionality, usability, performance, serviceability, capability, installability, maintainability, and documentation [76]. Reliability is one of the few measures that may be objectively measured, subject to clear articulation of requirements [46]. User requirements are a key determinant of reliability [76,184], but this is sometimes misinterpreted so software that fails to report a critical error to the user may be deemed reliable [184]. IEEE has sought to standardise various software quality metrics [104], but that standard does not require validation of direct measures used [185].

Software reliability, like hardware reliability, is often considered a stochastic process and therefore described by probability distributions but, unlike hardware, software does not wear out, burn out, or degrade with time in use [76]. Software generally exhibits reliability growth during testing and operation, as faults may be detected and removed when failures occur but, conversely, can exhibit a decreasing reliability due to abrupt changes of operational usage or incorrect software modifications [76]. While hardware faults are often physical faults, all software faults are design faults that are more difficult to visualise, classify, detect, and correct [202]. Even relatively small software programs can have large combinations of inputs and states that are difficult, or not cost effective, to exhaustively test [202].

Traditional reliability theory, which assumes that stationary processes affect reliability, cannot represent non-stationary phenomena associated with reliability growth or reduction experienced in software designs [76]. Software does not wear out, but some in the software reliability engineering community consider that traditional reliability models can represent software reliability, as the probability of encountering a latent defect that results in failure increases with the time that software is run in an untested manner [202].

Logical designs of modern VLSI systems (e.g. modern microprocessors) have several attributes and complexities in common with software development, with the final fabrication occurring in hardware [99].

Evaluation of software reliability relies on several basic definitions and concepts. Some of these are equivalent to definitions in §2.2.1, but some are adjusted slightly when applied to software.

Software is the collection of programs that directs operation of a computer, and documentation giving instructions on their use [108].

A **system** is an assemblage or combination of things or parts forming a complex or unitary whole [108].

A **subsystem** is a secondary or subordinate system [108].

A **software system** is an interacting set of software subsystems that is embedded in a computing environment that provides inputs to the software system and accepts outputs from the software [76].

A **service** is a time-dependent sequence of outputs from a software system that agrees with the initial specification from which the software implementation has been derived (for verification purposes) or which agrees with what system users have perceived the correct values to be (for validation purposes) [76].

A **failure** is an event in which the user perceives that the program ceases to deliver the expected service [76].

An **error** is a discrepancy between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition [76].

A **fault** is the cause of a failure or internal error [76].

A **mistake** is a human action that results in software containing a fault [76].

A **defect:** a generic term used to refer to either a fault or a failure (when the distinction is not critical) [76].

An **operational profile** is a characterisation of the set of operations that software can execute associated with the probability that each will occur [76].

Estimation is determination of current software reliability by applying statistical inference techniques to failure data obtained during system test or system operation [76].

Prediction is the determination or estimation of future software reliability based upon available software metrics and measures [76].

A **software reliability model** is a specification of the general form of the dependence of the failure process on the principal factors that affect it (fault introduction, fault removal, and operational environment) [76]. In practice, reliability and dependability after often considered synonymous terms for software [228].

Software reliability modelling

Developers use software reliability models to answer two basic questions about their software systems: "When will the software be reliable enough to ship?" and "What will be its expected reliability at that time?" [76,237]. Software reliability modelling has occurred since the 1970s [17], usually concerned with use of past failure data to predict future behaviours. Several models are based on traditional techniques for modelling hardware reliability, and use input data concerning failures per time period or predict time between failures [76]. As software complexity has grown, models and methods for architecture-based software reliability analysis have been emerging [226,239].

The basic assumptions of most models are [76]:

- The software is operated in a similar manner as that in which predictions are made;
- Every fault in a particular severity class has equal likelihood of being encountered;
- The failures, when faults are detected, are independent.

Most models also assume software failure is a stochastic process [76,237]. There is no quantitative evidence for this assumption: it may also be argued that software failure is a chaotic process, as software failures often result from human mistakes that are deterministic and not completely random in nature [237].

Models often make criteria assumptions such as [132,272] there being a fixed and finite number of potential faults, detected faults are fixed immediately, and that individual fault occurrence times are recorded (i.e. fault occurrences are not grouped in time).

Specific assumptions made by various models also include [132,272] a fixed number of errors in the code; bug fixing processes that do not introduce new errors; a constant program size; that error detection is an independent process; that testing is representative of intended usage; and that the error detection rate is proportional to the number of errors existing in the code.

The following sections describe classes of models that seek to predict software reliability in later stages of the software lifecycle (integrated test and beyond). There are also models that seek to predict reliability in earlier lifecycle phases, when design changes are less costly, that employ statistics based on analysis of requirements, design, and coding phases in order to predict subsequent reliability [76].

Traditional failure time models

Exponential failure time models are the subject of the most significant body of literature [76], and are overtly based on traditional modelling approaches for hardware failure [272]. Binomial models are based on a per-fault constant failure rate, and Poisson models are based on per-fault constant failure rate and an exponential time to failure of an individual fault. Time-between-failure models assume an exponential distribution.

The next most commonly used models assume the per-fault failure distribution to be the Weibull or Gamma models [76].

Infinite failure time models

Infinite failure-time models assume the software will never be fault free, as additional faults will be introduced during error correction processes [76]. These models require input of either actual times or software failure or elapsed time between failures.

Bayesian models

Bayesian models take the subjective viewpoint that software reliability should increase in periods when no fault is observed, reflecting increasing user confidence. These models therefore reflect both a prior distribution that incorporates past data and a posterior distribution that incorporates both past and current data [76]. Bayesian models consider the number of faults less significant than their impacts: a program with a fault in a frequently executed part of code is less reliable than a program with faults in rarely executed area of code [272].

3. Overview of electronic components and systems

3.1 Electronic components

An electronic component is a basic electronic element that is usually packaged in a discrete form with two or more connecting leads or metallic pads. Components are connected together, usually by soldering to a printed circuit board, to create a circuit with a particular function (e.g. an amplifier, radio receiver, or oscillator). Components may be packaged singly (e.g. resistor, capacitor, transistor, diode) or in more complex groups as integrated circuits (e.g. operational amplifier, resistor array, logic gate).

3.1.1 Resistors

A resistor is a two-terminal passive electronic component that produces a voltage across its terminals proportional to electric current flowing between them. Resistors are ubiquitous in most electronic circuits and equipment and are made of various compounds and films, as well as resistance wire (wire made of a high-resistivity alloy).

Primary characteristics of a resistor are resistance, tolerance (allowed uncertainty, often expressed as a percentage of resistance), and power rating (upper limit on the power of the device or its ability to dissipate energy).

Other characteristics include temperature coefficient (as resistance is often a linear function of temperature), noise characteristics (levels of random fluctuations of current/voltage), inductance, capacitance, and critical resistance (the value below which power dissipation limits the maximum permitted current flow and above which the limiting factor is applied voltage).

Fixed or variable resistors are typically employed to limit current in circuits. They may be integrated into hybrid and printed circuits, as well as integrated circuits.

3.1.2 Capacitors

A capacitor (or condenser) is a passive electronic component consisting of a pair of conductors separated by a dielectric or by a vacuum. When a potential difference exists between the conductors, an electric field is present in the dielectric. This field stores energy and produces a mechanical force between the conductor plates. The effect is greatest between wide, flat, parallel, narrowly separated conductors.

The primary property of a capacitor is capacitance, the ratio of the electric charge on each conductor to the potential difference between them. An ideal capacitor is characterised by a single constant capacitance. In practice the dielectric between the plates passes a small amount of leakage current as all devices have finite resistance.

Capacitors are used in electronic circuits to filter or smooth current in a circuit.

3.1.3 Inductors

An inductor (or a reactor) is a passive electrical component that can store energy in a magnetic field created by electric current passing through it.

The primary property of an inductor is inductance, which results from the magnetic field which forms around a current-carrying conductor that tends to resist changes in the current. Electric current through the conductor creates a magnetic flux proportional to the current. Any change in this current creates a change in magnetic flux that, according to Faraday's law, generates an electromotive force (EMF) that opposes the change in current. Inductance is a measure of the amount of EMF generated for a unit change in current.

An “ideal inductor” has inductance, but no resistance or capacitance, so does not dissipate or radiate energy. Real inductors have resistance (due to the resistivity of the wire and losses in core material) and capacitance.

Inductors are used, often in conjunction with capacitors and other components, to form tuned circuits that can emphasise or filter out specific signal frequencies.

3.1.4 Memristors

Memristors are a class of passive two-terminal circuit elements that maintains a functional relationship between the time integrals of current and voltage. The definition of the memristor is based solely on fundamental circuit variables, similarly to the resistor, capacitor, and inductor. Unlike resistors, capacitors, or inductors, memristors have nonlinear characteristics so there is no such thing as a generic memristor. Instead, each device implements a particular function. A linear time-invariant memristor is simply a conventional resistor [8].

Memristance, a contraction of "memory resistance", is a mathematical construct first proposed as a fundamental circuit element in 1971 [8] but a working memristor was not found until 2008 [298]. This switching memristor, based on a thin film of titanium dioxide has been presented as an almost ideal device, is much simpler than current transistor devices, and able to implement one bit of non-volatile memory in a single nanoscale device [339]. Memresistance is inversely proportional to the square of component size, so is an insignificant effect except with highly miniaturised electronic designs where nonlinear effects become more significant [298].

3.1.5 Diodes

Diodes have two active electrodes between which a unidirectional electric current, or rectifying, property is exhibited. The most common function of a diode is allowing an electric current in one direction (called the *forward biased* condition) while blocking current in the opposite direction (the *reverse biased* condition). The varicap diode (also referred to as a tuning diode) has properties of an electrically adjustable capacitor.

Real diodes do not display perfect on-off directionality but have nonlinear electrical characteristics that depend on the type of diode technology. Diodes also have many other functions in which they are not designed to operate in this on-off manner.

Most modern diodes are made from semiconductor materials such as silicon or germanium. Semiconductor diodes are discussed further in §3.2.1.

3.2 Semiconductor materials and technology

The term semiconductor refers to the class of materials with resistivity between those of conductors (e.g. metals) and insulators. Semiconductor materials are insulators at absolute zero, and have limited conductivity at room temperatures. Most commonly employed semiconductor materials are crystalline inorganic solids. Silicon is the most common semiconductor material used in commercial applications. Germanium, gallium arsenide, and silicon carbide are also commonly employed.

Electronic properties of a semiconductor may be altered in a controlled manner by introducing impurity into its crystal structure, in a process known as **doping**. This allows production of components with predictable and reliable electrical properties, making them suitable for mass production. Doping levels of the order of one per 100 million atoms is described as *low* or *light* doping. Doping levels of one per ten thousand atoms are described as *heavy* or *high*.

Semiconductor materials, depending on choice of doping material, may carry current as a flow of electrons in N-type semiconductors or of positive holes in the valence band for P-type semiconductors [1].

The term semiconductor also refers to any electronic device made from semiconductor material.

Semiconductor devices are fabricated using one or more layers of semiconductor material. The region where two types of material meet is referred to as a **junction**. A PN-junction refers to a junction between a P-type and an N-type semiconductor.

3.2.1 Diodes

A PN-diode is a device made from a PN-junction. Connecting the P-type region to the positive terminal of a power source and the N-type region to the negative terminal is described as forward-bias usage. Conversely, connecting the P-type region to the negative terminal and the N-type region to the positive terminal is described as reverse-bias usage.

The **depletion zone**, a region in the vicinity of the PN-junction, blocks current conduction from the N-type region to the P-type region, but allows current to flow in the opposite direction. This rectifying effect means that a diode conducts current easily when forward biased, and allows little current when reverse biased.

Electromagnetic radiation of a semiconductor crystal may break valence bonds linking neighbouring atoms, producing a hole and an excess electron [1], increasing the number of free carriers and therefore its conductivity. Photodiodes are optimised to exploit this phenomenon. When an electron and a hole recombine, energy is released, either as a quantum of electromagnetic radiation or as thermal vibrations of the crystal lattice [1]. This is the basis for operation of light-emitting diodes (LEDs) and laser diodes. High efficiency LEDs are replacing incandescent bulbs, particularly in applications requiring durability and low power consumption [327].

A PIN diode is a device in which a lightly doped or pure (intrinsic) semiconductor is sandwiched between heavily doped P-type and N-type semiconductor layers. PIN devices function poorly as rectifiers and are used as attenuators, fast switches, photo-detectors, and high voltage power electronics applications.

Some metal-semiconductor contacts also exhibit rectifying characteristics, and are referred to as a Schottky barrier. A Schottky diode uses such a metal-semiconductor contact and is characterised by both very fast switching times and low forward voltage drop.

3.2.2 Transistors

Transistors are semiconductor devices commonly used to amplify or switch electronic signals. A transistor is made of a solid piece of semiconductor material, with at least three terminals. A voltage or current applied to one pair of the transistor's terminals changes the current flowing through another pair of terminals. Transistors are a key active component in most modern electronics, as they may be mass-produced in a highly automated process (fabrication) that achieves low per-transistor cost.

Transistors have various characteristics, including; semiconductor material, structure, polarity, maximum power rating (high, medium, low), maximum operating frequency (low, medium, high, radio frequency (RF), microwave), application (switch, general purpose, audio, high voltage, super-beta, matched-pair), physical packaging (e.g. power modules, ball array, pin array, surface mount, through hole plastic, through hole metal).

A Field Effect Transistor (FET) is constructed with two layers of semiconductor material. Electricity flows through one of the layers, called the channel, which may consist of either N-type or P-type semiconductor. A voltage connected to the other layer, called the gate, interferes with the current flowing in the channel (from source to drain). Thus, the voltage connected to the gate controls the strength of the current in the channel as the resultant electric field controls the shape and hence the conductivity of the channel. Accordingly, the terminals of a FET are labelled gate, source, and drain. A Junction Gate FET (JFET, JGFET, or JUGFET) is typically used as an electrically controlled switch or voltage-controlled resistance. Most transistors contained in modern integrated circuits are Metal Oxide Semiconductor FET (MOSFET) devices.

A bipolar junction (BJT) transistor, named as its operation involves flow of both electrons and holes, consists of a layer of semiconductor material, named the base, sandwiched between two thicker layers of opposite type. Polarity of a BJT may therefore be NPN (P-type layer between two N-type layers) or PNP. The outside layer with (slightly) heavier doping is named the emitter and the other is named the collector. Terminals are named by the layer they are connected to. Usage requires connecting the emitter of a NPN-type transistor to negative, and the emitter of a PNP-type to positive. BJTs are used in amplifying or switching applications as a small current at the base terminal (flowing from base to emitter) can control or switch a much larger current between the collector and emitter terminals. Insulated gate bipolar transistors (IGPT) are characterised by high-efficiency and fast switching, and are used in medium- to high-power applications such as switched-mode power supply, traction motor control and induction heating.

3.2.3 Thyristors

A thyristor (thyratron transistor) is a solid-state device with at least four layers of alternating N-type and P-type material (versus a transistor having no more than three layers). Thyristors conduct when their gate receives a current pulse, and continue to conduct for as long as they are forward biased (i.e. while voltage across the device is not reversed). A conventional thyristor, once switched on by the gate terminal, remains latched in the on-state (i.e. does not need a continuous supply of gate current to conduct) once the anode current has exceeded the latching (or holding) current.

While the anode remains positively biased, a thyristor cannot be switched off until the anode current falls below the latching current. It may be switched off if the external circuit causes the anode to become negatively biased. In some applications this is achieved by forced commutation (switching a second thyristor to discharge a capacitor into the cathode of the first thyristor) after which a finite time delay (the commutated turn-off time) must pass before the anode can be positively biased into the off-state.

Thyristors are used for control of AC and DC motors, choppers (switches for switched-mode power suppliers), optimisation measures, and protection measures.

3.2.4 Logic gates and Integrated circuits

Logic gates act on one or more logic inputs to produce a single logic output. They are usually implemented electronically using diodes or transistors, but may also be constructed using electromagnetic relays, fluidics, optics, molecules, or mechanical elements. A logic level is usually represented by either a voltage or current, so each gate requires power to allow source and sink of currents or to achieve correct output voltage.

An integrated circuit (IC) is a miniaturised electronic circuit produced on a single crystal, or chip, of semiconductor material – usually silicon. It may contain many millions of components. ICs are typically encapsulated within a plastic or ceramic case, and linked via gold wires to metal pins with which it is connected to a printed circuit board and other components that make up electronic devices such as computers and calculators. Logic gates are one of the classes of basic building blocks of integrated circuits.

3.3 Electronic Packaging

Assembly of devices and components started shortly after invention of the transistor by using solder (primarily tin-lead), eutectic alloys (gold-silicon, gold-tin), and wire (copper, aluminium, or gold) [201]. During the 1960s, organic materials in hermetically sealed electronic packages were not allowed in military programs as they were associated with failures due to corrosion, metal migration, and out-gassing [201]. Metals are still used, but polymer adhesives are increasingly used in commercial applications.

Integrated circuit packaging is the final stage of semiconductor device fabrication, and includes both assembly processes and encapsulation or sealing processes [256,300]. Die attachment is the assembly process in which a die is attached to support structure. Bonding is the assembly process that establishes interconnections between circuits and devices. Encapsulation is the process of producing protective packages for circuits, in order to prevent physical damage or corrosion.

Packaging levels are generally defined in terms of the system elements being connected, rather than by the method of connecting them, as follows [56];

- *Level 1.* Chip pad to package lead (e.g. wire bonds in integrated circuits) interconnections are usually made by automated means, are highly specialised, generally not separable or repairable, are enclosed within a device package, and are required to be highly reliable.
- *Level 2.* Component to circuit board interconnections (e.g. DIP sockets) must typically withstand soldering, are relatively small, have contacts that are not individually repairable, have low mating cycle requirements, and are serviced by trained personnel.
- *Level 3.* Circuit board to circuit board connectors (e.g. edge connectors) have high pin counts, high pin density, must survive tens of mating cycles, support high speed electrical switching, be repairable, and be robust to user abuse.
- *Level 4.* Sub-assembly to sub-assembly connections (e.g. ribbon cables) typically facilitate cable applications, must survive hundreds of mating cycles, be robust to handling by untrained users, and often have shielding characteristics.

- *Level 5.* Sub-assembly to input/output connections (e.g. D sub cable assembly) are typically part of external system interfaces, so must meet the Level 4 requirements plus more demanding requirements for standardisation, interoperability, cosmetics, ease of use, robustness, shielding, filtering, and interference.
- *Level 6.* System to System connection (e.g. coaxial cable assembly) must satisfy the requirements of Levels 4 and 5, with additional demands for high robustness, support of many mating cycles, and standardisation. Shielding and filtering are also often important due to lengths of exposed cable.

Packaging has emerged as the limiting factor in cost and performance for many types of electronic devices [256], with near-term difficult challenges affecting all phases of assembly and packaging processes, including manufacturing design, test, and reliability [256,300,302]. Packaging materials are therefore forecast to evolve substantially over the next decade [302].

Traditional CMOS scaling is nearing physical limits, so there has been an increase in the rate of systems packaging innovation in order to continue growth rates associated with Moore's Law (§ 4.1). Functional Diversification, also known as "More than Moore" (MtM), refers to a longer-term emerging trend, in which the electronics industry seeks to provide progress without miniaturisation [300]. The electronics industry considers system level integration through System in Package (SiP) to be the most important trend in packaging to achieve this [302]. A consequence of these trends is a need for improved cooling to reduce operating junction temperatures due to large leakage currents [302].

Wafer Level Packaging (WLP) extends wafer fabrication processes to include device interconnection and device protection processes. Demand for WLP is rapidly growing due to demand for portable consumer products. WLP offers inherently lower cost, improved electrical performance, lower power requirements, and smaller size than other packaging techniques [302].

3.3.1 Bonding

After mechanical attachment to a substrate or to the inside of a package, individual bare die or chip devices are electrically connected [256]. Wire bonding is the attachment of fine wires from semiconductor chips to their substrates [256,300] and may be performed at high speed using thermo-compression, thermo-sonic, or ultrasonic equipment.

Flip-chip bonding and its variants replace wire-bond pads at the perimeter of the die with solder or metal bumps [256,300]. The die is flipped face down to mate with corresponding solder pads formed on the interconnect substrate. Solder is then reflowed to form electrical connections. Flip-chip bonding allows a large number of connections per unit area and allows short electrical paths. Flip-chip bonding technology has recently matured and is now considered a viable alternative to wire bonding [302]. Ball-grid arrays (BGA) are similar to flip-chip devices except that solder balls are formed or attached to the package or the chip carrier. BGA packages may contain flip-chip devices and wire-bonded devices.

Tape-automated bonding (TAB) positions the semiconductor die, fabricated with bumped interconnect pads, into apertures of a polymer-film tape and then gang-bonds it to cantilevered beams formed by metal pads on the tape [256,300].

Surface-mounting attaches and connects components to the board surface using batch solder reflow processes. This achieves higher packaging densities, greater reliability, and lower cost than the plated through-hole insertion process, so surface-mounting is widely used for low-cost, high-production consumer electronic assemblies [256,300].

3.3.2 Adhesives and coatings

The main polymer types used in electronics packaging are epoxies, silicones, acrylics, polyurethanes, polyimides, and cyanate esters. These types share some generic properties but there are many minor and major variations (e.g. curing schedule) affecting their usage in packaging processes and in reworking. Curing transforms low-to-moderate molecular-weight resins (monomers or oligomers) into high molecular-weight solid polymers. Most curing mechanisms require a catalyst or hardener and are initiated by some form of energy. Variation in the resin portion of adhesive and coating formulations is limited, and the hardener or catalyst added to cure the resin typically determines the final properties.

Adhesives are used in assembling semiconductor dies, both in single-chip packages and in multi-chip assemblies [201]. Both bare-chip devices and pre-packaged components are attached and electrically connected with adhesives to produce electronic circuits such as printed-wiring assemblies, thin- and thick-film hybrid microcircuits, and multi-chip modules. Adhesives, as pastes or as solid films, are also used in fabricating high-density multilayer interconnect substrates, flexible circuits, flat-panel displays, optoelectronics, high-speed high-frequency circuits, sensors, and smart cards. Adhesion is also a fundamental property of coating materials both initially and during the operation and lifetime of the hardware. If a coating does not adhere well to all surfaces and then maintain its adhesion under storage and duty-cycle conditions it will not perform its intended function and may result in electrical failure of the entire system [165].

Adhesives are also used for mechanical attachment of dissimilar materials, electrical connections, thermal dissipation, and stress dissipation [201]. Die-attachment adhesives are used to attach semiconductors (e.g. transistors, diodes, and integrated circuits) or other unpackaged devices (e.g. capacitors and resistors) into a single package or onto an interconnect substrate. These adhesives may be either electrically conductive where ohmic contact is required (to attach transistors, capacitors, resistors, etc) or electrically insulative (to attach ICs, stacked chips, connectors, heat sinks, and substrates) [201].

Conformal coatings such as polyurethanes, acrylics, epoxies, and silicones are used to protect printed wiring assemblies from moisture, handling, ionic contaminants, and particulates [165]. Organic coatings were initially used for protection of bare-chip devices, when chip passivation layers did not offer complete protection, and epoxy types that do not produce sodium chloride as a by-product of synthesis were subsequently introduced. With advancements in very high density, high-speed devices and circuits, coatings have been developed and applied as interlayer dielectrics for multichip modules, chip-scale packages, and optoelectronic packaging. These include polyimides, benzocyclobutenes (BCB), fluoroparylene, and several photo-

imageable coatings. Depending on the application, coatings are required to meet a broad range of adhesion, environmental protection, electrical, and thermal functions.

The two key functions of coatings used in electronic circuits are environmental protection and electrical insulation or isolation. Environmental protection includes shielding from moisture, chemicals, and contaminants that result in corrosion and electrical failures, and protection addresses physical abuse, such as handling and abrasion, temperature extremes, and radiation. Other functions include protection against handling conditions, abrasion resistance, friction resistance, protection from particles, and resistance to micro-organism. Applications include interlayer dielectrics for high density interconnect packaging, particle immobilisation or gettering, Electromagnetic Interference/Radio Frequency Interference (EMI/RFI) shielding, electrostatic discharge (ESD) protection, photoresists, and solder masking. Adhesion of the coatings is critical to fulfilling these functions. Organic coatings generally offer little benefit related to radiation protection.

3.3.3 Single chip packages

Integrated chips and packages may have their I/O connections (pins) formed throughout their surface area (area arrays) or on the chip periphery. Area arrays support large numbers of connections, and therefore increased functionality. Bare dies (e.g. integrated circuits) may be assembled in plastic or ceramic carriers, called chip-scale packages, which are slightly larger than the chip. Various types of chip-scale packages offer trade-offs of size, electrical performance, thermal characteristics, manufacture costs, and yield.

The dual-in-line package (DIP) is a commonly used rectangular package in which leads emanate from two parallel sides of the package, and may be constructed of moulded plastic, primarily epoxy, or of ceramic.

Quad-flat packs may be metal or ceramic cavity types or plastic-moulded packages where the leads emanate from all four sides. They may be of various sizes, and may package either single chips or multiple chips in one package. The leads may be shaped as “gull wings” so that the packages can be surface mounted onto an interconnect substrate.

3.3.4 Multichip packaging

Multichip packaging involves attachment and interconnection of a variety of chip devices onto single-layer or multilayer substrate. Two basic types of multichip packages are hybrid microcircuits and multichip modules.

Hybrid microcircuits are high-density circuits produced by directly attaching and connecting bare chips to a substrate. A significant variety of devices, connection techniques, and interconnect substrates can be used.

Multichip modules (MCM) are extended hybrid microcircuits, offering higher density (typically characterised by silicon to substrate density exceeding 30%) and greater electrical performance [144]. Modern MCM substrates consist of both inter-bedded connection lines and integral (embedded) passive devices [144].

Chip-on-board (COB), also known as chip-on-substrate (COS), is a cross between high-density hybrid microcircuits and low-cost Printed Wiring Assemblies (PWAs) that allows assembly of active and passive chip devices in hybrid and multichip modules. COB allows higher densities than Printed Wiring Boards (PWBs). Chip-on-flex (COF) is similar to COB except that the die, chips, or Chip Sized Packages (CSPs) are wire bonded, flip-chip attached, or epoxy connected to a flexible interconnect substrate.

Chip stacks allow increased component density, by assembling devices in a stack orthogonal to the substrate rather than on the substrate. Stacked chips of different sizes may be connected by wire bonding or flip-chip bonding, first to each other and then to the substrate. If the chips are of the same size, they must be staggered to allow wire bond connections to be made from one side of each device to the base substrate.

Flexible circuits are similar to rigid printed-circuit boards, but are fabricated from a thin copper foil adhesively bonded to a flexible dielectric film. The copper is photo-etched to form a circuit pattern using photolithography processes. A plastic film is then adhesive-bonded to the etched copper circuitry for moisture, contaminant, and handling protection, with open areas left for subsequently attaching components.

3.4 Programmable and designable electronics

A hardware item is simple only if a comprehensive combination of deterministic tests and analyses appropriate to the design assurance level can ensure correct functional performance under all foreseeable operating conditions with no anomalous behaviour [115]. If an item is not simple, it is complex [115]. An item comprised entirely of simple components may be complex [115]. Conversely, items that contain a complex device, such as an ASIC or PLD, may be categorised as simple [115].

Complex electronics comprises programmable and designable complex integrated circuits, and also systems built from significant numbers of interconnected basic components and integrated circuits [259]. “Programmable” logic devices may be programmed by the user [259]. “Designable” logic devices are typically not programmable by the user, and their design is submitted to a manufacturer for implementation [259].

Firmware (essentially software stored on a read-only device), and components on which it is stored (e.g. EEPROM, SRAM, Flash Memory) are not generally considered, in themselves, to represent complex electronics [259].

3.4.1 Programmable Logic Devices

A Programmable Logic Device or PLD is an electronic component used to build reconfigurable digital circuits. Unlike a logic gate, which has a fixed function, a PLD has an undefined function at the time of manufacture. Before the PLD can be used in a circuit it must be programmed or reconfigured.

Programmable Array Logic (PAL) is a family of PLD devices used to implement logic functions in digital circuits. The programmable logic plane is a programmable read-only memory (PROM) array that allows the signals present on the devices pins (or the logical complements of those

signals) to be routed to an output logic device. PAL devices have arrays of transistor cells arranged in a "fixed-OR, programmable-AND" plane used to implement "sum-of-products" binary logic equations for each of the outputs in terms of the inputs and either synchronous or asynchronous feedback from the outputs.

3.4.2 Complex Programmable Logic Devices

A Complex Programmable Logic Device (CPLD) contains a set of simple PLD blocks with their inputs and outputs connected by a global interconnection matrix. A CPLD therefore typically has two levels of programmability: programming of each PLD, and of interconnections between them. A common CPLD architecture is an arrangement of logic cells around a central shared routing resource. Interconnect information is typically stored in EEPROM, SRAM, or Flash memory.

3.4.3 Field Programmable Gate Arrays

A field-programmable gate array (FPGA) is a semiconductor device that may be configured by the customer or designer after manufacture. FPGAs are programmed using a logic circuit diagram or source code in a hardware description language (HDL) to specify how the chip will work. A FPGA can be used to implement any logical function that an ASIC could perform but, unlike an ASIC, the functionality may be updated after shipping.

FPGAs use a grid of logic gates and an array of simple configurable logic blocks with interspersed switches that can rearrange interconnections between logic blocks. Each logic block is individually programmed to perform a logic function. The switches are then programmed to connect logic blocks in order to implement complete logic functions. Unlike a CPLD, the FPGA architecture allows programming of individual logic blocks.

Programmable logic devices like FPGAs are increasingly being used in high-integrity and safety-critical domains, but there is currently lack of consensus on how they may be safely deployed and certified [306]. One issue is the question of whether the device should be treated as hardware or software during the certification process [306].

3.4.4 Application Specific Integrated Circuits

Application Specific Integrated Circuits (ASICs) are integrated circuits designed for specific applications. ASICs are typically designed by the end user and produced in volume. They allow a user to combine several parts and functions into a single chip, reducing cost and increasing reliability. ASICs may include programmable logic (FPGA, CPLD, or PAL). If the ASIC includes a microprocessor or other computer peripherals, it may also be categorised as a System-on-chip device.

3.4.5 System-on-Chip

System-on-Chip (SoC) combines all electronics for a complete product onto a single chip. The implementation typically includes a microprocessor and all ancillary electronics including switches, comparators, resistors, capacitors, timing elements, and digital logic.

SoC designs typically include a number of functional hardware blocks including;

- One or more processing elements (microcontrollers, microprocessors or DSP cores);
- ROM, RAM, EEPROM, or Flash memory;
- Timing sources including oscillators and phase-locked loops;
- Peripherals including counters, timers, real-time timers, and power-on reset generators;
- External interfaces such as USB, FireWire, Ethernet, or UART;
- Analogue interfaces and converters;
- Voltage regulators and power management circuits;
- A bus to connect the other components.

SoC designs also include software executed on the processing elements to control the peripherals and interfaces. Most SoC designs use a platform-based solution, based on reuse of pre-qualified hardware blocks and software drivers that control their operation. The hardware blocks are assembled using CAD tools and the software modules are integrated using a software development environment. SoCs that do not allow user-configuration are usually implemented as ASICs.

A reconfigurable SoC provides similar support of design customisation, except that devices and peripherals are implemented using a reconfigurable matrix such as a FPGA. The software typically sets up the hardware before use. This allows chip functionality to be updated by changing the configuration code.

SoC designs usually consume less power and have a lower cost and higher reliability than the multi-chip systems they replace. With fewer packages in the system, assembly costs are reduced as well. However the total cost is typically higher for one large chip than for the same functionality distributed over several smaller chips, because of lower yields and higher non-recurring engineering (NRE) costs. In particular, functional verification of chip designs accounts for a significant proportion of effort in a chip design before it is submitted to a foundry.

3.4.6 Microprocessors

A microprocessor incorporates most or all functions of a central processing unit (CPU) on a single integrated circuit (IC). Computer processors were originally constructed from small and medium-scale ICs containing up to a few hundred transistors. From the 1980s, the integration of the whole CPU onto a single VLSI (Very Large Scale Integration) chip greatly reduced cost of processing capacity.

While performance characteristics like processing power or speed of a microprocessor increase with number of transistors, the relationship is not proportional. Factors such as clock speed and microarchitecture, within each processor family, have a greater impact on performance by various measures than number of transistors. For example, AMD64 processors have better overall performance by several measures than Pentium 4 processors, despite the Pentium 4 family having more transistors.

Microarchitecture (or "computer organisation") is a description of the electrical circuitry of a computer, processor, or digital signal processor sufficient to completely describe operation of the hardware. Instruction Set Architecture (ISA) is the architecture that is visible and documented for programmers. Microarchitecture and Instruction Set Architecture together make up the field of computer architecture.

The ISA approximately represents the programming model of a processor as seen by an assembly language programmer or compiler writer, and includes the execution model, processor registers, address and data formats. The microarchitecture is a lower level structure than the ISA that represents details hidden in the programming model. It specifically describes constituent parts of the microprocessor and how these interconnect and interoperate to implement the architectural specification.

Microcode is a layer of lowest-level instructions involved in the implementation of machine code instructions in many computers and processors. It provides an abstraction to separate the machine instructions from the underlying electronics. Many computers use a microcoding approach to implement their control logic, in which programming instructions are decoded and executed by triggering electrical signals.

A single microarchitecture may be used to implement many different instruction sets. If the microarchitecture includes microcode, changing of instruction sets may be simplified by means of changing the control store. Two machines may also have the same microarchitecture, and hence the same block diagram, but radically different hardware implementations. Conversely, machines with different microarchitectures may have the same instruction set architecture, and thus be capable of executing the same programs. New microarchitectures and/or circuitry solutions, along with advances in semiconductor manufacture techniques, underpin performance gains between generations of processors.

Since about 2007, a shift towards using throughput rather than frequency to measure microprocessor performance has led to increased use of multicore processors [234,302] that distribute computing load across multiple cores, rather than operating a single core faster [289]. Single-thread performance has levelled off, and is no longer correlated with transistor count due, at least partially, to microprocessor design elements (out of order execution, prefetching, pipelining) reaching points of diminishing return [278]. Multi-core processors exhibit better performance per watt than single core processors, offer faster communication between threads running on different cores due to fast high-bandwidth networks and cache-coherency protocols, and can allow weight and space benefits through reducing the total number of processors and boards [289]. The cores are often simplified versions of their single-processor counterparts, so also take up less integrated circuit real-estate [288]. However, to fully exploit multicore processors, application software needs to be explicitly designed to be highly parallel or multi-threaded [278,288,289]. Such software development is difficult and error-prone [289], and can result in reduced performance due to resource contention and other phenomena that occur in a physically parallel processing environment [288]. Current software for aerospace applications is often intentionally designed for serial execution with limited threads, in order to ensure determinism and to aid verification and certification processes [289].

3.4.7 Microcontrollers

A microcontroller is a small computer on a single integrated circuit consisting of a relatively simple CPU and a set of support functions (e.g. crystal oscillator, timers and watchdogs, serial and analogue I/O). Program memory and a typically small amount of RAM are often included on chip. Microcontrollers are designed for small or dedicated applications, so simplicity is emphasised. Some microcontrollers operate at clock rate frequencies as low as 4 kHz, which is adequate for many applications and enables low power consumption (order of milliwatts or microwatts). They will generally have the ability to retain functionality while waiting for an event, such as a button press or other interrupt. Power consumption while sleeping (CPU clock and most peripherals off) may be of the order of nanowatts, making many microcontrollers well suited to long lasting battery applications. Other microcontrollers serve performance-critical roles, functioning more like a digital signal processor, with higher clock speeds and power consumption.

3.5 Other physical elements

This section describes some broad classes of parts that are used to build a computing system, and therefore contribute to delivering required functions and to its operating and non-operating reliability. There are many devices and variations, to meet many possible requirements, so this does not represent a comprehensive survey.

A *fastener* is a hardware device that mechanically joins or affixes two or more objects together. Electronic fasteners are generally small components for spacing, positioning, shielding, or physically reinforcing electronic devices. They may be components of other devices, such as connectors.

A *connector* provides a separable connection between two elements of an electronic system without unacceptable signal distortion or power loss [56]. A *terminator* is a connector placed at the end of a cable or wire to reduce unacceptable signal reflections and interference that would affect other connected devices. A separable connection allows for easy repair, upgrading, maintenance, or interconnection. Connectors and terminators must typically meet both mechanical requirements (mating force limitations, meeting a specified number of mating cycles, etc) and electrical requirements (impedance, etc) [56].

In computing, an electrical connector may be known as a physical interface, for example the Physical Layer in the OSI networking model [80].

A *cable* is two or more wires that are bonded, twisted or braided together to form a single assembly. Electrical cables are used to carry electrical currents. *Cords* are cables with connectors or terminators at their ends.

A *switch* is an electrical component that can break an electrical circuit, interrupting or diverting current. Digital active devices such as transistors (§3.2.2) and logic gates (§3.2.4), which change output state between two logic levels or connect different signal lines, are also described as switches. All switches control a binary state (on versus off, closed versus open, connected versus disconnected). Switches are categorised by the physical stimulus to which they respond.

Protection devices are either active or passive components that protect circuits from excessive current or voltage. Classes of protection devices include over-current protection devices, over-voltage protection, and protection against inrush current (high initial draw of a device).

A *power supply* is a source of electrical power, and a *power supply unit* (PSU) is a device or system that supplies electrical or other types of energy to other devices via an output terminal. A *battery* is a power supply that uses one or more electrochemical cells. A *switched-mode power supply* is a device that incorporates a switching regulator in order to provide a required output power when connected to an input supply.

A *heat sink* is an environment or object that absorbs and dissipates heat from another object using either direct or radiant thermal contact. Heat sinks function by transferring thermal energy ("heat") from an object at a relatively high temperature to a second object at a lower temperature that has a greater heat capacity. In electronics, a heat sink is usually a metal object placed in contact with the hot surface of an electronic component. Heat sinks are commonly used to cool modern integrated circuits such as microprocessors, digital signal processors, and graphics processing units.

Enclosures or *cases* are physical containers that contain and protect the main components of a computer. A computer *fan* is a mechanical device placed inside a computer case for active cooling purposes, to draw cooler air into the case from the outside, expel warm air from inside, or move air across a heat sink to cool a particular component.

3.6 Embedded Systems

A **computer system** is a computer plus any peripheral devices and software that enable the computer to function. The term **embedded system** is a closely related and frequently used term that is not uniformly defined in literature. Some available definitions include;

- "… a combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a dedicated function" [163].
- "… information processing systems that are embedded into a larger product and are normally not directly visible to the user" [164].
- "… an engineering artefact involving computation that is subject to physical constraints" [260].
- "… a microprocessor based system that is embedded as a subsystem in a larger system (which may or may not be a computer system)" [282].

Common characteristics of embedded systems include [163,164,282];

- Connection to a physical environment through sensors or actuators;
- Dependability, with attributes including reliability, maintainability, availability, safety, and security;
- Purpose built for some specific tasks or applications;
- A user interface that is specific to the task or application;
- Designed to tight deadlines by small teams;
- Real-time constraints that will cause system failure if not met.

A number of these characteristics are examples of non-functional requirements, which are concerned with the system's interaction with the physical world [291].

These characteristics often result in attributes such as [163, 164, 282] small code size, low weight, limited hardware resource (memory and processor capacity) usage, low supply voltages, low microprocessor or microcontroller clock frequencies, and only those hardware components required being present.

General-purpose computers may be adapted for low-volume embedded applications by selecting specific hardware, limiting what programs may be run, and utilising a real-time operating system [282].

Shortcomings of current design, validation, and maintenance processes make software the most costly and least reliable part of embedded applications [260], limiting ability to exploit potential of emerging hardware and communications technologies [260]. Current processes focus mainly on functional requirements (i.e. on a description of how the system responds to particular inputs or stimuli) while neglecting or deferring non-functional requirements (related to quality or cost of the system) [261,291,330]. This is considered to be partly due to demands of getting systems running quickly to fulfil basic needs, and partly due to the "soft" nature of non-functional requirements that leads to quality attributes being treated as technical issues affecting only detailed design or test [330]. In recent years, embedded systems have been increasingly required to implement advanced user interface, security, or networking functionalities, so the size and complexity of embedded software is increasing [283,309], and becoming an increasing threat to reliability [283].

4. Commercial and market effects

The global semiconductor industry has experienced decades of rapid technological change, rising costs for production capacity, and declining prices for final products [156]. There has also been an increase of vertical specialisation in semiconductor design and manufacture, with the growth of "fabless" design and marketing firms and of foundries, the manufacturing counterparts who contract for production [156].

Worldwide production in the electronics industry during 2007 totalled US$1.3 trillion, most driven by business and individual consumer spending [302]. Computers and office equipment was the largest segment, totalling US$446 billion or 35% of total production, followed by smaller industry segments including portable and consumer electronics (US$300 billion), personal communications equipment (US$176 billion), medical electronics (US$66 billion), and automotive electronics (US$79 billion) [302].

Global sales of semiconductors for 2008 totalled US$248.6 billion compared to US$255.6 billion in 2007, a decrease resulting from the global economic recession and weakening demand for major drivers of semiconductor sales: automotive products, personal computers, cellular phones, and corporate information technology products [305].

Consumers drive over half the worldwide demand for semiconductors, so profitability of the chip industry is increasingly linked to macroeconomic conditions such as GDP, consumer confidence, and disposable income [305]. Due to global economic pressures, device fabrication foundries have spread around the world and moved from high-cost to low-cost locations [292,332], with increasing outsourcing of high-skill, high-technology industries and processes [292]. In the US, exceptions to this trend are that design activities and manufacture of "core technologies" (e.g. cutting edge microprocessors) are not yet typically outsourced [292]. Weak intellectual property protection laws is considered a factor in this but, as technical skills increase in low-cost locations, outsourcing of design activities may also increase [292].

4.1 Moore's Law

Moore's Law describes a long term trend, first observed in 1965 [6], in which the number of transistors that can be placed inexpensively on an integrated circuit doubles approximately every two years. Although initially expressed as an observation or empirical forecast, semi-conductor manufacturers accepted Moore's Law as a business goal, and continue to expend significant effort to achieve the specified growth in processing power that, it is presumed, their competitors will eventually achieve. In this respect, Moore's Law has become a self-fulfilling prophecy [93].

Other measures associated with digital technology exhibit exponential growth, such as;

- Number of transistors per integrated circuit;
- Manufacturing cost per unit area of semiconductor;
- Power consumption of computer nodes;
- Random Access Memory (RAM) storage capacity;
- Pixels per unit price in a digital optical sensors;
- Affordable network capacity (sometimes described as Moore's Law for Data Traffic or Moore's Law of Fibre);
- Capital cost of semi-conductor fabrication plants (doubles with the introduction of each new generation of technology [133]).

Conversely, unit cost per transistor and storage cost per unit of information exhibit exponential decline.

The growth predicted by Moore's Law has traditionally been achieved through "geometric scaling", associated with ongoing miniaturisation of selected technologies such as CMOS transistors [300]. This is expected to continue for some aspects of chip manufacture [300]. Manufactured silicon MOSFET devices have been formally classed as nanotechnology (defined as feature sizes less than 100 nanometres) since about the year 2000 [221]. The ability to miniaturise further is increasingly being limited by concerns such as parasitic resistance and parasitic capacitance, which were traditionally considered to be insignificant in scaling theories [221]. As of early 2009, the state-of-the-art "node" for CMOS transistors was 45 nanometres, which refers to the standard gate length for a single transistor [307]. As gate widths are scaled further below 100 nanometres, classical laws of physics increasingly give way to quantum

mechanics, and CMOS devices become increasingly susceptible to quantum tunnelling effects and thermal leakage currents, resulting in reduced performance [307].

In future, it is expected that the semiconductor industry will increasingly utilise "equivalent scaling", in which performance will be increased through innovative design, software solutions, and innovative processing [300]. The ongoing practices of geometric scaling and equivalent scaling are described by industry as "More Moore" [300].

Functional Diversification, also known as "More than Moore" (MtM), describes a longer-term emerging trend, in which the semiconductor industry seeks to provide additional value to customers in manners that don't involve miniaturisation, such as support for non-digital functionalities (RF communication, power control, increased usage of passive components, sensors, or actuators) and increasing system level integration in chip, circuits, and system packages associated with evolution to package-level (SiP), chip-level (SoC), and Stacked Chip SoC (SCS) solutions [300]. It is forecast that capabilities offered by CMOS transistor processes will need augmentation by integrating multiple new devices, at either chip or package level, around a CMOS core [300]. MtM results in a number of technology and material challenges due to trends towards higher interconnect densities in packages, increasing thermal densities, increased challenges gaining access to test components within a package, and increasing challenges of ensuring reliability [302].

4.2 Market segmentation of the semiconductor industry

Before World War II, U.S. firms led the world in manufacture of consumer electronics [74], consisting primarily of phonographs and radios. Many of these firms helped develop and produce electronic equipment for the war and, subsequently, most returned to producing consumer products. After a sharp decrease between 1947 and 1950, military spending increased rapidly with the advent of the cold war. Military-related research and development rose to nearly 75% of the US total research and development expenditure, with electronics and aerospace receiving a large share of that expenditure: electronic content of US DoD's expenditure in equipment, research and development grew from US$3.2 billion in 1955 to US$7.8 billion in 1964, before declining to US$7.3 billion in 1966 [74]. Military spending on electronics during the 1960s and 1970s consistently represented over 80% of the product value of shipments in the electronic systems and equipment industry segment [74]. However, subsequently the market share declined. In 1984, aerospace and defence sectors reflected 7% of the total market, most of that in military grade components [134].

In response to already reducing influence of the military on commercial electronic trends, the US Secretary of Defense William Perry [63] instructed the US armed services in 2004 to adopt commercial products and standards in order to reduce costs [158]. This memo stated that military standards could only be used if an adequate commercial specification was unavailable [74,134,158]. This contributed to a further increase of COTS usage within Defence acquisitions, and caused electronic component manufactures to review viability of low-volume component production lines and switch capacity to more lucrative higher volume commercial projects. By 1999 the Aerospace and Defence sectors only represented 0.3% [134] or 0.4% [129] of the total market.

Defence acquisition programs traditionally relied on assurances associated with Military Standards to more easily meet requirements for extended operating range, quality and reliability, and could achieve economic advantages of reduced parts inventories and better quantity discounts if military grade components were used across multiple programs [134]. With reduced market share, the ability to obtain military grade component and potential to achieve economic advantages has reduced. However, military systems are also becoming increasingly complex, with demands for command and control functions, communications, sensors, and other requirements [158].

As the electronics industry matures, several market segments are entering the commodity phase of the life-cycle, in which breakthrough technology is no longer sufficient to ensure business success [302]. The electronics industry is currently completing a major restructuring, in which the centre of manufacturing competence is moving away from vertically integrated OEMs (Original Equipment Manufacturers) towards a multi-firm supply chain consisting of EMS (Electronics Manufacturing Services) and ODM (Original Design Manufacturers) in low cost geographic areas [302,307]. There is an increasing OEM focus on time-to-market and increasing complexity of emerging technology [302].

Although the industry is developing new technologies, the focus is on sustaining innovation, or bringing better products to established markets [307]. At the high performance end, barriers to new market entrants are high and increasing to the point where only a few oligopolies exist [307]. At the low performance end, where the industry focuses on marginal improvements to existing products, there is significant competition and scope for disruptive innovation in which innovative design houses and reconfigurable foundries allow greater flexibility to build custom Systems On Chip (SoC) on demand [307]. The semiconductor industry, as a whole, remains focused on building better integrated circuits and participates little in emerging nanotechnology fields such as micro-machines, solar cells, and flexible displays [307].

4.3 Lead Free Solder

Eutectic or near-eutectic tin/lead (Sn/Pb) solder was used by the electronics industry for decades due to solderability and reliability characteristics [273]. However, legislation mandating the banning of lead in electronics has been actively pursued worldwide since the mid 1990s due to the environmental and health concerns [112,273]. The European Union Reduction of Hazardous substances (RoHS) Directive [158] has contributed to reduced usage of lead-free solder by most segments of the electronics industry, and required development of lead-free replacement of the tin-lead coating used in lead-frames and printed circuit boards [273]. Usage of lead in electronics is also reducing due to market differentiation and advantage being realised by companies producing "green" products that are lead-free [273]. Plating with tin is therefore increasingly prevalent, even in high-criticality applications [192].

No "drop in" replacement for Sn/Pb solder has been identified for all applications. Several potentially promising solder alloys have higher melting temperature, so some components or substrates cannot reliably sustain the soldering process [112] and the solder processing window for lead-free solders is narrower than for solders containing lead [329]. However, Sn-Ag, Sn-Ag-Cu (SAC), and other alloys involving elements such as Sn, Ag, Cu, Bi, In, and Zn have been identified as promising replacements for standard 63Sn-37Pb eutectic solder [273,318]. The favoured lead-free solder varies with geographic region but, generally, high-tin alloys are

preferred [112,318] with Sn-Ag-Cu (SAC) most prevalent worldwide [318]. Industrial consortiums have proposed several SAC alloys that offer benefits of relatively low melting temperatures (compared with the 96.5Sn-3.5Ag binary eutectic alloy) and superior mechanical and solderability properties in comparison with other lead free solders [273].

Some classes of long life and high-reliability products were granted exemption under the RoHS Directive [158,191], due to insufficient data to determine if SAC alloys were sufficiently reliable in these applications [251,290]. Avionics was not the subject of any exemption, but the transition to lead-free avionics introduced concerns as avionics are subject to harsh temperature, radiation, and vibratory environments and have service lives exceeding 20 years [191]. Despite exemptions, however, other high-reliability communities experienced difficulty by 2007 in procuring components with traditional SnPb surface finishes, and therefore unknown reliability or supply risks [290]. Most component suppliers have switched over to tin finishes for I/O terminals, so tin whiskers (§5.3.5) remain a risk for high-criticality applications [251,290]. Most lead-free BGAs are incompatible with tin-lead assembly processes, resulting in a need for ongoing availability of tin-lead based BGAs or development of a robust process for mixed (lead-free and lead-based) assembly [290]. Sn-Ag-Cu solders are more tolerant of lead contamination, and therefore more compatible with existing lead-solder based systems, than other lead-free solders [112]. The reliability of lead-free solder joints is generally considered acceptable for thermal cycling applications, but fragility of joints remains a concern [329].

The lead-free transition is continuing to present cost, reliability, and manufacture process compatibility concerns for the electronics industry, particularly related to high-reliability applications [302].

4.4 Electronic part life cycle

The commercial trends that underpin the progress of Moore's Law (§4.1) are associated with high rates of change of electronic components within commercial products and systems. A consequence is that many electronic components are likely to have a life cycle significantly shorter than the life cycle of the products they are part of [124]. This concern is prevalent in airborne and military systems that are characterised by long developmental lead times and field lives, which may encounter obsolescence concerns before being fielded and are likely to experience such concerns during field life [124,158]. The effort for qualification or certification of these systems is also often significant, making subsequent obsolescence treatments an expensive and time-consuming process [124]. The lifecycle mismatch between a system and its components also affects widely used "workhorse" consumer products that have a minimal feature set but are expected to function reliably (e.g. pagers used in restaurants to advise patrons their table is ready) [124].

Standard models [89] include five lifecycle stages of electronic parts (introduction, growth, maturity, decline, and phase-out). Practically, a final discontinuance or obsolescence stage may also be described. Typical descriptions of these stages [124] follow.

- *Introduction stage*. Often characterised by high production costs associated with recently incurred design costs, low yields, frequent modifications of design or implementation, low or unpredictable production volumes, lack of specialised equipment for production or manufacture, and high marketing costs. Customers who buy in this stage tend to value performance over price or reliability.
- *Growth stage*. Follows market acceptance of the part. Increased sales and competition justify development and use of specialised equipment for production or manufacture, leading to benefits through economies of scale due to mass production, mass distribution, and mass marketing yielding price reductions.
- *Maturity stage*. Characterised by high-volume sales, and increasing dominance of manufacturers with lower cost of production.
- *Decline stage*. Characterised by decreasing demand and often with decreasing profit margin. The manufacturers in the market reduce in numbers and may become more specialised. Few modifications of the part specification occur.
- *Phase-out stage*. Occurs when the manufacturer sets a date for ceasing production of the part. The manufacturer possibly provides discontinuance notice to customers, and may also advise a last-time buy date and suggest alternative parts or after-market manufacturers.
- *Discontinuance and obsolescence stage*. Occurs when manufacture ceases at the part number or other manufacture-specific level. The part may remain in the market if the production line or part stocks were acquired by an aftermarket source. Technology obsolescence is the particular form of obsolescence in which the basic technology that defines a component is discontinued and no longer manufactured.

Not all components follow this sequence due to economic, social, or environmental concerns [89]. Some parts undergo a false start and die out. Some parts are associated with a niche market, with some unique applications, and hold a relatively low and constant sales level over an extended period. The "Decline" stage may be delayed or reversed by the definition of new market segments, new applications, or perceptions causing an increase in demand late in the product life cycle.

Average introduction rates for new generations of commercial integrated circuits, in 2000, were approximately [129];

- Logic families: six years;
- Memory families: nine months;
- Microprocessors: two years;
- Digital Signal Processing (DSP): three years;
- Programmable Logic Devices (PLD): one year;
- Linear Interfaces: eight years;
- Gate Arrays: two years.

Any system or application that experiences growth encourages demand for parts required in its manufacture, and is therefore exposed to risks associated with availability of those parts [124]. Manufacturers may discontinue a product line for business reasons that are unrelated to technical concerns. There may therefore be mismatch between the lifecycle of a system and for components of that system. Similarly, there may be mismatch between the lifecycle stage of a component and the technologies that define those components. Using simple measures such as number of manufacturers of a part may therefore lead to optimistic assessments of the future risk with using that part. In particular [124];

- Many market sources do not necessarily infer market health. In particular, there may be several manufacturers of a device after manufacture of an underlying defining technology ceases;
- A small number of sources, or existence of a large market player, does not imply a risk of losing a device or technology group, as manufacturers still in business may profitably command most of the market share. This is particularly true when after-market manufacturers continue manufacture after the original manufacturers have discontinued their product lines.

Although it is considered that uncertainty makes vertical integration approaches more effective, uncertainty introduced by technological obsolescence works the other way [28]. Vertical integration of the semiconductor industry means it would take considerable time to introduce new types of devices (e.g. non-silicon technology) at all levels of the semiconductor industry, so it is considered likely that silicon-based CMOS technology will remain the dominant form of nanotechnology for roughly another 30 years [221].

4.5 Value Engineering

Value engineering [252], also known as value management, is a systematic business management method to improve the "value" of goods or products and services by examining and evolving function. Value is defined as the ratio of function to cost [252], and may therefore be increased by either improving the function or reducing the cost of products. In the United States, the practice of value engineering is enshrined in Public Law affecting all government acquisition executives [77].

Value Engineering follows a structured thought process based exclusively on "function", i.e. what something "does" not what it is [252]. "Functions" are always described in a two word abridgment of an active verb and measurable noun (what is being done - the verb - and what it is being done to - the noun) in the most non-prescriptive way possible. This is the basis of "function analysis" [252]. A goal of function analysis is to understand something with such clarity that it can be described in two words, the active verb and measurable noun abridgement [252]. For example, the function of a pencil is to "make marks". This description then allows considering what else can make marks. From a spray can, lipstick, a diamond on glass to a stick in the sand, one can then identify which alternative solution is most appropriate [252].

Function analysis uses rational logic (a "how" – "why" questioning technique) to identify relationships that increase value. It is considered a quantitative method which focuses on hypothesis-conclusion approaches to test relationships, and operations research, which uses model-building to identify predictive relationships [252].

The use of value engineering techniques has led to planned obsolescence being associated with product deterioration and inferior quality, and it is claimed [2] that this could give engineering a bad name, because it directs creative engineering energies toward short-term market ends rather than more significant engineering goals. Companies are often considered to employ value engineering solely to cut costs [2]. It is considered that a focus on functional requirements can result in neglecting or deferring non-functional requirements (related to quality of the system) [261,291,330] and assigning them as technical issues affecting only detailed design or test [330]. Value Engineering Change Proposals (VECP) processes are, however, considered to offer several advantages over traditional Engineering Change Proposal (ECP) for the US military in addressing DMSMS (Diminishing Manufacturing Sources and Material Shortages) problems [340].

5. Physical phenomena and failure mechanisms of electronics

5.1 Physical phenomena that affect electronics reliability

Electronic boxes used in aircraft and missiles often have odd shapes that permit them to make maximum use of the volume available in tight spaces and a high packing density [118]. This results in a number of distinct responses to physical phenomena.

5.1.1 Electromagnetic interference

Any current-carrying conductor radiates an electromagnetic field and is also affected by any existing electromagnetic field around it. Devices may therefore be adversely affected by nearby components or equipment. This may be seen in the form of increased noise levels or compromise of power-supply or control voltages in a manner that causes equipment malfunction.

Electromagnetic interference (EMI) is a disturbance to an electrical circuit due to either electromagnetic conduction or electromagnetic radiation emitted by an external source. The disturbance may interrupt, obstruct, degrade, or limit circuit performance. The source may be any artificial or natural object that carries rapidly changing electrical currents.

Narrowband interference usually arises from intentional transmissions. Broadband interference usually comes from incidental radio frequency emitters, including power lines, electric motors, computers, and other digital equipment. The spectra of these sources generally decrease with frequencies, but there is significant variation with the originating device. Broadband interference is typically difficult to filter out once it has entered the receiver chain. Conducted Electromagnetic Interference is associated with physical contact between the conductors.

Integrated circuits are often a source of electromagnetic interference, but the radiation is usually only significant if there is coupling with larger objects such as heat sinks, circuit boards, and cables. Means of reducing electromagnetic interference with integrated circuits include the use

of bypass or "decoupling" capacitors on each active device (connected across the power supply, as close to the device as possible), rise time control of high-speed signals using series resistors, and positive supply voltage filtering.

5.1.2 Vibration and shock

Vibration is an oscillation in a mechanical system [143] where some structure or body moves back and forth [118]. Shock is an aspect of vibration in which the excitation is non-periodic (e.g. in the form of a pulse, step, or a transient vibration).
Electronic equipment may be subjected to many different forms of vibration over wide frequency ranges and acceleration levels. These may be due to an active association with a machine or moving vehicle or due to transporting the equipment from manufacturer to customer [118].

The vibration frequency for aircraft varies between 3 Hz and 1 kHz, with acceleration levels from about 1 G to 5 G peak [118]. The greatest accelerations appear to occur in the vertical direction in the frequency range of about 100–400 Hz. The lowest accelerations appear to occur in the longitudinal direction, with maximum levels of about 1 G in the same frequency range.

The vibration frequency for helicopters varies from 3 Hz to 500 Hz and acceleration from about 0.5 to about 4 G [118]. The greatest accelerations appear to occur in the vertical direction at frequencies near 500 Hz and large amplitude at low frequency.

The frequency of vibration for missiles varies from 3 Hz to 5 kHz, with the lower frequency vibrations appearing due to bending modes in the airframe structure [118]. Acceleration levels range from about 5 to about 30 G peak, with the maxima occurring during power-plant ignition at frequencies above 1 kHz.

The vibration environment in supersonic aircraft and missiles appears more random in nature than periodic. However, sinusoidal vibration tests are still being used to evaluate and to qualify electronic equipment that will be used in these vehicles [118].

High forcing frequencies in aircraft and missiles increases difficulty designing resonance-free electronic systems for these environments [118]. Complete encapsulation of an electronic box in some expanding rigid type of foam, to increase the resonant frequency for a small box, is generally considered impractical because of expense to maintain, troubleshoot, and repair such a system [118], while heat would limit usage. Forcing frequencies present in aircraft and missiles will therefore excite many resonant modes in every electronic box, and the electronic support structure must be dynamically tuned with respect to the electronic components to prevent resonances that can lead to rapid fatigue failures.

5.1.3 Temperature

Temperature changes introduce distortion stress on the junction between two materials with different expansion rates. With repeated temperature changes, material fatigue can damage hermetic seals, affect adhesion of joints, and can cause opening of connective materials. Increased temperature accelerates several chemical reactions and material changes, which in turn can accelerate failure mechanisms. If a device is connected inappropriately, heat generated

by the equipment can accelerate the temperature change and further accelerate some failure mechanisms.

5.1.4 Humidity

Humidity leads to condensation adhering to the surface of devices so can increase electrical conductivity of surface materials. This increases current leakage, resulting in change in operational characteristics. Humidity can also accelerate chemical and electrical reactions, contributing to corrosion.

5.1.5 Atmospheric pressure

Atmospheric pressure affects devices used at high altitude, both in mountainous regions and in aircraft applications. Low pressure permits corona discharge between electrodes, and reduces the ability to radiate heat, resulting in internal thermal generation and accelerating increase in component temperature.

5.1.6 Salinity

Salinity greatly affects electronic devices used in coastal regions, ships, and marine applications. Salt that adheres to an element surface causes deterioration of insulation between electrodes and increases rate of damage due to corrosion.

5.2 Failure mechanisms related to electronics device design

Electronic component or system reliability is primarily determined during product design [301]. Within integrated circuits, dimensions of transistors and other factors affecting both performance and reliability are derived from required device functional characteristics. The manufacturing process is broadly divided into two processes. The wafer process places transistors, diodes, and resistors onto silicon substrate in accordance with the design pattern. The assembly process consists of dicing of the pattern developed on the wafer, die bonding, wire bonding, and sealing to form the final product.

5.2.1 Dopant variability

Random dopant fluctuations cause variability in the threshold voltages of transistors, affecting static memory stability [222]. This results from discrete distribution of dopant atoms in the channel of a transistor [257]. With increasing miniaturisation, the number of dopant atoms in the transistor channel reduces exponentially and two transistors sitting side by side may therefore have different electrical characteristics due to random variation in the number of dopant atoms [208,257].

5.2.2 Migration

Increased integration and miniaturisation of semiconductor devices increases current density in metal wiring, while larger circuit scales increase power consumption [122]. Metal wiring used in semiconductor devices has a polycrystalline structure, with each crystal grain differently oriented. Many defects are therefore located around grain boundaries. Some metal atoms caught in these defects have weak inter-atom bonds, so can migrate in conditions of current density or

thermal stress. This allows atom voids to form and grow, contributing to increased resistance of wiring and electrical disconnection.

5.2.2.1 Electromigration

Electromigration occurs when momentum transfer due to impacts with electrons transports conductor metal atoms within circuit interconnects. Atoms migrate from one end of each interconnect to the other, eventually leading to increased resistance and short circuits [7,11,111,181,195,265]. Electromigration damage ultimately leads to failure of the affected IC, but the first symptoms are intermittent behaviours that are challenging to diagnose. As some interconnects fail before others, the circuit exhibits seemingly random errors, which may be indistinguishable from other failure mechanisms.

The likelihood of IC failure due to electromigration effects can be reduced by proper semiconductor design practices. Some effects of electromigration may be reversed, or even healed, by reversing current flow in the affected wire [351]. Electromigration is not a concern for bulk electrical wiring (eg used in home circuitry) but is one of the most significant sources of degradation in integrated circuits [351]. However, miniaturisation increases the probability of failure due to electromigration because of increased power and current densities [181,301,351].

One frequently used [11,122,181,220,297,301,351] model suggests that the mean time to failure associated with electromigration is proportional to;

$$J^{-n} e^{\frac{E_{aEM}}{kT}}$$

Where J is the current density in the interconnect material (and depends on feature size), k is Boltzmann's constant, and T is absolute temperature. The parameters n and E_{aEM} (the activation energy for electromigration) characterise the interconnect material.

While increasing wire width may mitigate effects of electromigration, the trade-off in miniaturised semiconductor design is that other sources of failure, such as NBTI and oxide breakdown, would become, relatively, more significant [351].

5.2.2.2 Stress Migration

Stress migration, also known as Stress Induced Voiding (SIV), occurs when mechanical or thermal stress causes metal atoms in interconnects to migrate with no electric current applied [111,181,195,265,297]. When semiconductors are subjected to high temperatures, difference in thermal expansion between materials causes the stress on the wiring to increase further [122]. One model [122,181,297,301] suggests that the mean time to failure associated with stress migration is proportional to;

$$\left|T_0 - T\right|^{-m} e^{\frac{E_{aSM}}{kT}}$$

where T is absolute temperature and T_0 is the metal deposition temperature and is dependent on manufacture technique. The parameters m and E_{aSM} (the activation energy for stress migration) characterise the interconnect material.

5.2.3 Hot Carrier Degradation

Hot Carrier Injection (HCI) in solid-state devices or semiconductors is the phenomenon where either an electron or a hole gains sufficient kinetic energy to overcome a potential barrier and migrate to a different area of the device. Some hot carriers are scattered by phonons, quantised vibration modes in the device crystal lattice. Others lose energy due to impact ionisation, a process in a material by which an energetic charge carrier can lose energy by creation of other charge carriers [301]. The rate of HCI is related to channel length, dielectric thickness, and operating voltage [216]. Operational conditions and toggling frequencies of CMOS transistors also contribute to the HCI rate [78].

Hot Carrier Degradation (HCD) is the accumulation of damage associated with HCI in solid-state devices or semi-conductors [66]. Hot carriers can degrade gate dielectrics causing electron and cause hole traps to form which increase sub-threshold leakage current and cause shifts in threshold voltage that contribute to eventual device instability or failure [78].

MOSFET devices are now submicron sizes [220] and electric fields in the devices have grown, resulting in increased generation of hot carriers due to acceleration of electrons near a drain [78,122,216,220,297,301]. This causes fluctuations of threshold voltage and/or current characteristics of the device [122,220,301]. Unlike other failure modes of semiconductors, hot carriers cause higher degradation at lower temperatures [78,301] and cause the greatest degradation of device characteristics in the normal operating temperature range [122,220]. Two models are used in the JEP-122A Standard [141, 216];

- The N-channel model for NMOS devices, in which substrate current is an indicator of hot carriers, is characterised by;

$$MTTF = B(i_{sub}) - Ne^{E_a/kT}$$

  where B is a function of doping profile and sidewall spacing, i_{sub} is the substrate current, N is in the range 2 to 4, and E_a is the activation energy (typically in the range -0.1 eV to -0.2 eV);

- The P-channel model for PMOS devices, in which substrate current is an indicator of hot carriers, is characterised by;

$$MTTF = B(i_{gate})^{-M} e^{E_a/kT}$$

  where i_{gate} is the peak gate current, M is also in the range 2 to 4.

There is little discussion in literature concerning an appropriate distribution model for HCI, but as a device becomes more complex, with millions of gates, it may be viewed as a system so a constant failure rate (exponential) model may be assumed in practice [216].

Degradation of device characteristics due to hot carriers also occurs in bipolar transistors when a reverse bias is applied across emitter and base [301]. With a trend towards shallow junction devices, there is a tendency towards increased reverse leakage current between emitter and base, causing device degradation to occur readily as a result of hot carrier effects [301].

Hot carrier degradation contributes to a significant aging effect on transistor performance in which transistor saturation current degrades over time [257].

Hot carrier effects may be suppressed by moderating the electric field using a Lightly Doped Drain (LDD) or Deeply Doped Drain (DDD) structure which prevents a high-resistance structure near the drain [220,301]. Lower supply voltages through circuit design are also promoted as reducing hot carrier effects [122,220].

HCI is also the basis for operation of some non-volatile memory technologies such as EEPROM cells and NAND flash memory. HCD is one of the factors limiting write/erase cycles of those memory technologies. Impact ionisation also causes a related failure mechanism in memory cells and charged coupled devices (CCD) that are often used in photoelectric light sensors. Energetic charge carriers generated by impact ionisation are accelerated towards the semiconductor substrate, causing secondary impact ionisation. The resultant hole and electron pairs reach adjacent memory cells and CCD cells as noise charge that fills storage capacitors inverts their memory states. This failure mechanism is expected to increase with increasing device scaling and miniaturisation [301].

5.2.4 Dielectric breakdown

Dielectric Breakdown refers to destruction of an insulating layer in a semiconductor device. Oxide dielectric layers are used in many parts of electronic devices, including; as gate oxide between the metal and the semiconductor in MOS (Metal Oxide Semiconductor) transistors; as dielectric layer in capacitors; and as inter-layer dielectric to isolate conductors from each other.

Oxide breakdown is also referred to as "oxide rupture" or "oxide punch-through". The related phenomena of oxide wearout, along with hot carrier degradation, contributes to a significant aging effect on transistor performance [257].

Dielectric thickness reduces as device features are miniaturised, resulting in increased vulnerability to device voltages [111]. Device reliability may limit the amount of scaling or miniaturisation that may occur, because increases of direct tunnelling current may decrease time to oxide breakdown [103,116]. Electrical Overstress or Electrical Discharge events can cause dielectric breakdown due to high voltage across the oxide layer, allowing current flow. This current flow causes localised heating, thereby inducing larger current and therefore localised heating and eventual meltdown of the silicon, dielectric, and other materials. This meltdown creates a short circuit between the layers supposedly isolated by the oxide, and occurs due to randomly located latent defects in the gate dielectric [19].

Physical and predictive models of oxide reliability in CMOS devices and circuits have been extensively reviewed in literature [125]. The breakdown process is generally considered to occur in two stages [216]. In the first stage, the dielectric is damaged by a localised hole and bulk electron trapping within it and at its interfaces. The second stage is reached when the increasing density of traps within the dielectric form a percolation (conduction) path, resulting in a short circuit between the substrate and the gate electrode. This is essentially the process of "soft" dielectric breakdown, and can lead to "hard" breakdown [196] in which silicon melts, oxygen is released, and a silicon filament is formed. Contrary to some claims that hard breakdown is a fast thermally driven process, it has been found to be a gradual process in which gate current

increases at a predictable rate that is exponentially dependent on instantaneous stress voltage and oxide thickness [155].

Both early-life and time-dependent dielectric breakdowns are primarily due to the presence of weak spots within the dielectric layer arising from poor processing or uneven growth during manufacture. They result in the same failure attributes, except that the former occurs early in the life of the device and the latter occurs after a greater period of use (in the "wearout" stage). Both categories of breakdown involve destruction of the dielectric while under normal bias or operation. The risk of dielectric breakdown generally increases with the area of the dielectric layer, since a larger area means the presence of more defects and greater exposure to contaminants. Understanding of time-dependent dielectric breakdown mechanisms remains a key concern affecting efforts to enhance reliability of recent capacitor technology [336], such as those based on PZT (Lead Zirconate Titanate) materials that are of increasing interest because of their outstanding ferroelectric and dielectric properties [98].

One model [181], based on experimental work [139], suggests that the mean time to failure associated with time dependent dielectric breakdown is proportional to;

$$\left(\frac{1}{V}\right)^{bT-a} e^{\frac{\left(X+\frac{Y}{T}+ZT\right)}{kT}}$$

where V is the applied voltage, and the parameters a, b, X, Y, and Z are empirically determined through experimentation.

5.2.5 Voltage Stress (Bias Temperature Instability)

Biasing refers to the method of establishing predetermined voltage or current at various points in a circuit. Practically, bias often refers to application of a fixed DC voltage to the same point in a circuit as an AC signal, to select operating response of an electronic component. Linear circuits involving transistors, in particular, typically require specific DC voltages and currents to operate correctly.

Fixed charge and interface state in the gate oxide film increases when a bias is applied under high temperature to the gate or drain of a MOS transistor. This causes shifts of voltage and current, resulting in degradation of device characteristics through Bias Temperature Instability (BTI) [253,301]. Negative and positive bias temperature instabilities (NBTI and PBTI respectively) can both be significant in Static Random Access Memories (SRAM), with effects including degradation of both read and write times due to NBTI and PBTI individually, and in combination [316]. NBTI of PMOS devices and PBTI of NMOS devices are major contributors to circuit aging effects [347]. NBTI of PMOS devices became a major reliability concern as semiconductor devices were miniaturised below 90 nm, and PBTI of NMOS devices is increasingly a concern below 45 nm [352].

Negative Bias Temperature Instability

Negative Bias Temperature Instability (NBTI) refers to the generation of positive oxide charge and interface traps in all MOS structures under negative gate bias, in particular at elevated temperature [217,247]. It is particularly associated with upward shifts in transistor threshold

voltage (and hence reduction in current) in p-channel MOS (PMOS) devices stressed with negative gate voltages at higher temperature [199,217] which, in turn, causes degradation of circuit performance metrics such as speed, power, and leakage [199,250,295]. Similar amounts of positive charge and interface state generation occur for both n- and p-type silicon substrates. However, the charge in the interface states depends on bias so the net effect on threshold voltage is greater for p-FETs, because the positive oxide charge and positive interface charge are additive [217]. NBTI is therefore, practically, a greater concern for PMOS devices. It has become more significant with increasing miniaturisation [297], particularly with 3 nanometre (or less) thin film PMOS transistor gates [301]. NBTI failures appear characterised by lognormal statistics, combined with a slower degradation rate [284].

Despite being first reported in the 1950s, the phenomenon of NBTI is not well understood and many fundamental and practical questions remain [216, 217], such as explaining why stress damage builds up slowly but partial or complete recovery then occurs rapidly after removal or reduction of then stressing voltage [247]. The three most prominent models in literature feature holes injection into the oxide dielectric, electron tunnelling, and electrochemical reactions [216].

The importance of NBTI is that it can determine useful lifetime of all CMOS transistors [204]. The effect increases with reducing transistor dimension, with the electric field applied to the gate oxide, and with reduced operating voltage (as a given threshold shift causes a larger relative impact on circuit behaviour) [204].

NBTI is one of the most significant factors affecting microprocessor reliability [284] and can lead to microprocessor failure because of timing constraint violations [195,265]. It is a dominant contributor to circuit aging in advanced CMOS technology [250,295].

The effect of process conditions on NBTI has been the subject of extensive but largely empirical study, with observations including;

1. Hydrogen and/or water play a large role in NBTI and water released from inter-metal dielectrics in the upper layer of an integrated circuit can increase NBTI [217].
2. A Silicon Nitride barrier above the transistors and other barrier materials can be used to control the effect [217].
3. Chlorine impurity can increase the effect, but Fluorine has a beneficial effect [217].
4. The addition of nitrogen into the gate dielectric to reduce gate leakage, and control of boron penetration, has had a side effect of increasing NBTI [199, 217]. This effect is not well understood [247].

It is well established [217] that the NBTI phenomenon is independent of current flow through the oxide dielectric, in contrast to oxide breakdown (§5.2.4). However, as oxides have become thinner, the increasing tunnelling current can lead to dielectric breakdown defects exhibiting characteristics similar to NBTI, which confuses efforts to collect and interpret data on the phenomenon [217].

5.2.6 Soft Errors

The passage of nucleons (deuterons, protons, neutrons, or alpha particles) can produce changes in conductivity of semiconductors, and even of conductivity type (N-type or P-type) [1]. Contributing phenomena include transmutation (e.g. exposure of silicon to thermal neutrons leads to production of phosphorous atoms while fast neutrons produce aluminium atoms) and lattice defects produced by fast neutrons and charged particles [1].

Semiconductor memory defects that can be recovered by rewriting affected data are referred to as soft errors. If these errors that are not easily detected or corrected, and are not of a transient nature, they are described as firm errors. They may be caused by power supply line or ground line noise [122] and radiation [21]. Soft errors may also be caused by alpha rays emitted by trace amounts of radioactive materials in the immediate chip environment [16] or packing materials [19] such as Uranium and Thorium contained in package or wiring materials [122,220,297,301]. Soft errors may also be caused by neutrons [60,65,170]. Neutron flux has some dependence on geographic latitude and is more prevalent at altitude [173], making it an important consideration for aircraft systems. Data from military aircraft, experimental flights, and laboratory testing indicates that non-radiation-hardened SRAM can experience a significant soft upset rate at aircraft altitudes due to energetic neutron rays created by cosmic ray interactions in the atmosphere [65].

When alpha rays penetrate silicon, electron and hole pairs are generated along the path of the ray. When this occurs within a DRAM or SRAM memory cell, the result is a soft error. A memory cell mode error occurs when an alpha particle impinges on a memory cell area, and the electrons generated flow into the memory cell area and corrupt the data it stores [301]. Within DRAM, for example, a cell containing charge has a data value of zero, while an empty or discharged cell has a value of 1 [301]. Therefore a data $1 \rightarrow 0$ change occurs when electrons impinge on the memory cell area. The effect becomes more significant with reducing device geometry and when memory elements hold less charge [173].

A bit line mode error occurs when the electron-hole pair generated by the alpha particle produces an electrical current, affecting the bit line electric potential [36,297,301]. The bit line's electric potential varies with the data of the memory cell, and is compared with the reference potential to read out the value of that cell. A sense amplifier is used to amplify the measured change. If alpha particles impinge near the bit line during the (extremely short) period between memory readout and sense amplification, the bit line potential changes. The result is a $1 \rightarrow 0$ data change if the bit line potential falls below the reference potential and a $0 \rightarrow 1$ change, otherwise. Since a bit line mode error only occurs during data readout, the frequency of soft errors is directly proportional to frequency of data readout (or inversely proportional to cycle time).

A Combined Cell Bit (CCB) error results when both the memory cell and the bit line collects radiation-induced charge, with neither sufficient to cause an error, but which cause an error in combination. At short cycle times it has been reported [301] that CCB errors can be the dominant cause soft error in high-density high-speed dynamic memories.

Neutron-induced errors have significant implications for applications based on SRAM FPGA [65,152,173]. Existing detection techniques, which rely on periodically reading back the FPGA

configuration, may allow corrupt data to remain in the system for a significant period of time, and the read-back circuits are themselves susceptible to single-event upsets or damage. Schemes to detect and correct FPGA soft errors increase complexity of the system design and significantly increase board space materials cost. Antifuse and Flash based FPGA technologies are immune to neutron-induced errors.

Although it is often assumed that soft errors are uncorrelated events (each event only causes one error), some studies suggest that Multiple Bit Upset events are a significant concern for advanced memory technologies [210,233,267].

While the majority of soft errors in single-core microprocessor systems appear to be masked at the architectural level, soft errors have emerged as an increasing vulnerability of multicore microprocessor systems [248]. A soft-error within one core may spread to other cores in various ways, depending on processor topology and on methods of communication within the processor [248].

5.3 Failure mechanisms related to electronics manufacture process

Manufacturing processes include various steps, such as heat treatment, chemical treatment, processing, testing, and inspection. These steps introduce factors that affect reliability, including processing variances (dimensions, property values, etc), defects and damage that occur in the manufacture process, handling errors, and equipment operation errors. For highly miniaturised semiconductor products, these processes are adversely affected by presence of dust. The manufacture process related to the wafer (silicon substrate) is fundamental to reliability of the product, which is affected by crystal defects, resistivity dispersion, surface contamination, and surface flaws.

The assembly process for semiconductors begins with dicing, the process of separating individual silicon chips or integrated circuits following wafer processing. Die bonding and wire bonding processes, which secure chip and bond electrodes to the exterior, form junctions between different materials where changes in temperature and physical forces can result in die cracks or open faults, which can compromise behaviour of the product.

Impurities in sealing compound (e.g. sodium, potassium, or chlorine), moisture absorption, thermal expansion, and mould shrinkage can result in failures of resin encapsulation, such as corrosion, bonding wire breakage, and die cracks. For hermetic sealing, moisture content, impurities in the sealing gas, and conductive foreign matter can adhere to the chip surface can cause increase in leakage current or faulty operation.

5.3.1 Wire bonding

To assemble a semiconductor device, a semiconductor chip is bonded onto the die pad of a package [84]. Modern assembly processes often bond an Aluminium surface electrode and an inner silver or gold plated lead using a fine metal wire, typically gold or aluminium [256]. Five distinct inter-metallic compounds may be formed between gold and aluminium. Differences in diffusion coefficients, and rapid diffusion of aluminium into gold at higher temperature, results in Kirkendall [223] voids forming at the aluminium- Au_2Al interface or at the gold- Au_2Al_5 interface [328]. This causes unavoidable long-term life degradation in the gold-aluminium joint

[297]. Long-term storage of the semiconductor device, particularly at high temperature, causes contact resistance of the inter-metallic joint to increase, eventually resulting in breaks in the joint [297]. This phenomenon is associated with visible discolourations known as "purple plague" and "white plague", which are associated with Au_2Al and Au_2Al_5 inter-metallic compounds respectively [328], but the visual phenomena may occur without structural weakening of the metal joint [9]. These and other inter-metallic compounds (e.g. Au_5Al_2 and Au_4Al [297]) are generally brittle, and may break due to metal fatigue or stress cracking under conditions of vibration or flexing [328]. Bromine, used to achieve flame retardation in resin material, may catalyse oxidation of the Au_4Al alloy layer, leading to high resistance [297]. Impurities in the bonding wire, on the pad metallisation, or at the wire-bond-pad interface have further been shown to cause rapid inter-metallic growth and Kirkendall voiding at lower temperatures than associated with normal inter-metallic formation [328].

Similar concerns of inter-metallic compounds affect bond strength with metals other than aluminium and gold, for example in lead free solders [197,223].

5.3.2 Metal Ion migration

Metal ion migration is a phenomenon in which metal ions move under an electrochemical effect. This is a distinct phenomenon from electromigration (§5.2.2.1) or stress migration (§5.2.2.2) in wiring. The phenomenon can occur with solder materials or with gold, but silver and copper pose more problems in practice [297]. As an electrolytic reaction, the migration only occurs when DC voltage is applied between electrodes. The time to short-circuit is (roughly) inversely proportional to the potential difference and proportional to distance between the electrodes [297].

Silver ion migration occurs when silver, in the form of foil, plating, or paste is subjected to a voltage under high humidity and temperature. Electrolytic action causes silver to migrate and grow as a blot or dendrite on the surface of insulating material. This may reduce resistance of the insulator or cause a short circuit. Generally, the effect does not occur if relative humidity is 50% or less, but accelerates rapidly when relative humidity is over 70%. The effect of temperature is less significant, but it does accelerate the process. Presence of dust tends to accelerate ion migration, as it both retains moisture and contains water-soluble matter that increases concentration of electrolytes.

In multi-pin plastic packages, lead frames are bonded with heat-resistant polyimide tape to prevent deformation during production and associated short-circuiting between leads. Applying a voltage to a multi-pin plastic package with copper lead-frames at elevated temperature causes ionisation of copper because of reaction with solvent elements in the adhesive. The effect is accelerated by higher temperature and increased concentration of anions in the taping adhesive.

5.3.3 Shear force under temperature cycling

In a temperature cycle environment, shearing force can cause the semiconductor chip surface to push toward the centre of the chip surface can cause failure phenomena due to the contraction stress of the mould resin at a low temperature [297].

Metallic wiring materials, such as aluminium, are deformed easily by an external force and therefore do not tolerate external stresses [297]. Al sliding is a failure mechanism by which thermal stress produced by the difference in the coefficient of thermal expansion with the mould resin causes the Al wiring materials on the semiconductor chip to deform and slide [122]. If a horizontal stress is applied to the centre of the chip surface and to the wiring around it, the passivation film on the surface withstands the stress rather than the Al wiring [297]. Therefore, when the Al wiring is wide, a protective film with structurally low strength is destroyed and the Al wiring slides [297]. This failure phenomenon is known as Al sliding and causes cracks in the passivation film [122,297]. These cracks can lead to circuit damage [122,297] and subsequently device malfunction as moisture and other matter then enters through these cracks and leads to Al corrosion and other reliability failures [122].

5.3.4 Filler induced failure

Mould resins include fillers that secure strength and have a thermal expansion coefficient close to that of the chip [297]. When a filler of about 100 micrometres is located on the chip surface by moulding, the filler presses the chip surface due to the temperature cycle and other factors, damaging the chip surface and possibly causing a failure [297]. This is sometimes described as filler attack [122].

5.3.5 Metal Whiskers

Metal whiskering is a crystalline metallurgical phenomenon involving the spontaneous growth of tiny, filiform hairs from a metallic surface. The effect is primarily seen on elemental metals but also occurs with alloys. Cadmium, Tin, Zinc, and silver are well known as whisker-forming materials [192]. Gold whiskers have also been observed [162]. Metallic film deposits also exhibit other eruptions that are quite different in appearance from the whisker eruption (flowers, extrusions, volcanoes, etc).

A metallic whisker is a single crystalline filamentary surface eruption from a metal surface [192]. Whiskers are usually found on relatively thin (0.5 to 50 micrometre) metal films that have been deposited onto some kind of substrate material. A typical whisker is 1–5 micrometre in diameter and 1-500 micrometres in length. Whiskers may be straight, kinked, and even curved. There is still no consensus concerning the specific growth mechanism of whiskers [236].

At frequencies above 6 GHz tin whiskers can act like miniature antennas, affecting the impedance of digital circuits [190]. Whiskers can also break off, leading to debris that contaminates equipment physically separated from the originating site [182,251]. In some circumstances, a short circuiting tin whisker ionises into plasma that can conduct hundreds of amperes of current [182,190], increasing damaging effect of the short circuit. Whiskers are typically able to carry current of 10-35 milliamps but current up to 75 milliamps has been observed [190]. They can, depending on diameter and length of the whisker, form stable short circuits in low-voltage high impedance circuits or transient short circuits [182]. Whiskers can occur under protective coatings, leading to distortion or puncturing of coating materials. Temperature cycling appears to accelerate whisker growth when there is a large mismatch in thermal expansion [190]. To date, no plating parameter and no additive other than lead has been identified that will prevent whiskers growing on plated tin [190].

There are no conclusive means to mitigate growth and effects of tin whiskers, but commonly applied methods include [182]:

- Avoid tin-plated parts;
- Strip and replate;
- Solder-dip plated surfaces using a tin-lead solder;
- Select a matte or low-stress tin finish;
- Select under-plating to reduce inter-metallic formation;
- Vary thickness of tin plating;
- Reflow of pure tin-plated surfaces;
- Anneal;
- Avoid applying compressive loads on plated surfaces;
- Apply conformal coat.

5.4 Failure mechanisms related to usage of electronics

External stresses in the operating environment can accelerate various failure mechanisms that are associated with device design (§5.2) or manufacture (§5.3). Stresses may be associated with human actions or with the natural environment, including temperature, humidity, atmospheric pressure, salinity, over-voltage due to lightning, and cosmic rays. Of these, temperature and humidity are often the most significant [301].

A change in temperature introduces distortion stress on the junction between two materials with different expansion rates. Repeated changes in temperature, material fatigue can damage hermetic seals, affect die bond adhesion, and can cause opening of bonding wire. A rise in temperature speeds up several chemical reactions and accelerates material changes, which in turn can accelerate failure mechanisms. Additionally, if the device is connected inappropriately, heat generated by the equipment can accelerate the temperature change and further accelerate some failure mechanisms.

Humidity leads to condensation adhering to the surface of the device and therefore can increase electrical conductivity of the surface material. This increases current leakage, resulting in change in operational characteristics of the device. Humidity can also accelerate chemical and electrical reactions, contributing to corrosion.

Atmospheric pressure affects devices used at high altitude, both in mountainous regions and in aircraft applications. Low pressure induces corona discharge between electrodes, and reduces the ability to radiate heat, resulting in internal thermal generation and accelerating increase in component temperature.

Salinity greatly affects devices used in coastal regions, ships, and marine applications. Salt that adheres to an element surface causes deterioration of insulation between electrodes and increases rate of damage due to corrosion.

Lightning causes significant voltage surges.

Cosmic rays or alpha rays from radioactive isotopes can increase incidence of soft errors (§5.2.6).

Human actions affecting reliability include vibration during transport, shock during handling (e.g. dropping), heat during soldering of printed circuit boards, voltage surges due to use of switches, noise from poor relay contacts or motor devices, electrostatic damage caused by use in low-humidity environments, malfunction due to proximity to a transmitter, and ultrasonic vibration during cleaning of printed circuit boards after soldering. Stresses also occur when a product is used outside specified operating range (exceeding rated voltage, malfunction due to use at low voltage). Breakdown may occur due to excessive load, and a device may malfunction or break down due to use outside specified operational timings.

5.4.1 Electrostatic discharge

Electrostatic discharge (ESD) refers to the sudden and short-duration electric current that flows between two objects at different electric potential, either caused by direct contact or induced by an electrostatic field. ESD can cause complete semiconductor device failure or degradation of device characteristics such as increasing leakage current, degrading voltage that can be withstood, or maintenance of open state. There are a number of models that describe causes of ESD, and form the basis for test methods, including [122,220,301];

- The Human Body Model (HBM) models the discharge of electrostatic change accumulated in a human body to a semiconductor device. Capacitance and resistance of the human body are represented as $100\ pF$ and 1500Ω respectively within MIL-STD-883F [188], method 3015.7. The HBM discharge waveform is prescribed by that standard;
- The Machine Model (MM) models objects with higher static charge capacity than the HBM, such as device handling equipment made of metal, and a discharge under low-resistance conditions to devices. This model was originally based on worst-case values for the HBM ($200\ pF$ and 0Ω) but testing methods did not specify a discharge waveform, allowing considerable variation between manufacture and user tests. Accordingly, this model is deprecated;
- The Charged Device Model (CDM) models the device itself as the source of static electricity, due to friction produced when the device approaches a charged object, resulting in discharge of through leads. This model is considered to reproduce the discharge mechanism in the form closest to field conditions.

5.4.2 Junction breakdown

Junction breakdown [220,297,301] is a thermal breakdown phenomenon that occurs when excessive current flows in a reverse bias direction relative to the PN-junction. This increases temperature of the junction, and thermal breakdown occurs when the melting point of the semiconductor material (1415 Celsius for silicon) is exceeded.

5.4.3 Metallisation breakdown

Metallisation breakdown [297,301] is also caused by thermal destruction, when the power density applied is sufficient to fuse metal.

5.4.4 Latchup

CMOS integrated circuits include a PNPN thyristor-type structure comprised of parasitic transistors between the source and drain pins. Latchup occurs when a parasitic thyristor is activated by electrical noise, which causes a short circuit between the source and drain pins [220,301]. This allows a large current to flow until between the pins until the circuit breaks down or power is cycled.

5.4.5 Thermal Cycling

Thermal cycling causes permanent damage that accumulates with each temperature cycle experienced by a component [111,181,195,265]. Large cycles occur at a low frequency because of changes like powering up and down. Small cycles occur at a higher frequency because of changes in workload and fine-grained power management techniques.

One model [181], based on the Coffin-Manson equation that relates fatigue life to strain amplitude, suggests that the mean time to failure associated with large thermal cycles is proportional to;

$$\left(\frac{1}{T_{average} - T_{ambient}} \right)^{q}$$

Where $T_{ambient}$ is ambient temperature and $T_{average} - T_{ambient}$ is the average large thermal cycle a structure on a chip experiences and q is the material-dependent Coffin-Manson exponent.

6. Electronic component and system reliability

The numbers of failures of electronic systems or components in any particular environment are typically small, so field-failure data provides little information for determining the actual causes of recorded failures and typically did not include or address information such as failure site, failure mechanism, load, environment history, materials, or geometrics [58]. In practice, components were attributed incorrectly as the cause of problems despite 30%-70% of components retesting OK [54]. This means cause-and-effect relationships are not captured, limiting insight or control over actual causes of failure, and limiting ability to use design and usage data within a traditional reliability assessment. Traditional reliability approaches are therefore inapplicable for assessing product reliability in a test chamber or in the field, reliability design guidance, or for comparing reliability of competing designs, despite being widely used for these applications [58].

Electronic systems use large numbers of similar components over which the designer has little control. Quality control methods may be used in procurement and manufacture phases, but

circuit designers have no control over design reliability of devices unless they are custom-designed. Electronic components often cannot be inspected easily because of encapsulation. Other than gross defects that are readily detected in testing, unreliability of electronic components is generally considered due to defects that are not detected by simple inspection and do not have immediate effect. Temperature and voltage are considered the dominant failure accelerating stresses for most electronic components. As components fail and are replaced, the percentage of defects in the remaining components reduces. Wearout is rarely considered a significant concern for electronic components in part because the commercial electronics market often focuses on satisfying warranty period so products are often obsolete before wear-out effects become dominant [150].

Parameter drifts and accidental short circuits at connections are common causes of observed failures of electronic systems. Traditional system designs therefore focus on ensuring that voltage, current, and temperature remain within rated values, and seek to minimise hot-spots and significant temperature gradients. Designers also apply basic rules, such as minimising number of adjustable components, selecting components based on parameter values obtained by testing, assembling systems so components are easily accessible for adjustment, and partitioning designs into subassemblies for easy testing and problem diagnosis.

6.1 Application of traditional reliability methods

6.1.1 Historical perspective

Traditional empirical reliability assessment methods are the basis of MIL-HDBK-217 [48] and several commercial derivative products concerned with predicting electronic reliability [25,55,67,114,128]. MIL-HDBK-217F [48§3.3] itself noted several limitations affecting its applicability. These products employ curves fitted to field-failure data, so may be characterised as bottom-up statistical methods [142]. Although lacking physical justification, the constant failure rate model was considered sufficient until the 1980s, as devices to that time were fragile and had several intrinsic failure mechanisms so their observed failures approximated the constant failure rate model [216].

During the 1980s and early 1990s, evidence increasingly suggested that the constant failure rate model was not applicable to integrated circuits [216]. Research concerned with updating MIL-HDBK-217 suggested that the constant failure rate model not be used [62]. Studies found that predictions based on MIL-STD-217 and derivative products disagree with each other, lack accuracy in predicting product reliability, and were often conservative; in many cases, actual product reliability significantly exceeded that predicted [176]. Other criticisms of traditional empirical methods include;

- "*Much of the available information on electronic part reliabilities has been obtained by means of statistically derived behaviour patterns of specific parts, operated in the laboratory under controlled environmental conditions, or operated in actual equipments under field service environmental conditions. The resulting data are then extrapolated to indicate part behaviour under other sets of stress conditions. Such extrapolations most often yield suspect results, because the mechanisms responsible for the bulk of failures under one set of stress conditions may very well not be responsible for a simply determinable proportion of failures under other sets of conditions.*" [4];

- "They do not account for temperature cycling" [133];
- "The method does not reflect modern manufacturing trends" [133];
- "The method does not differentiate good quality and design practices" [133];
- "The method penalises system level factors, such as transient protection circuits, that influence reliability" [133].

In 2003, US Army policy [175] directed that;

> "*Solicitations should require access to information adequate for evaluating the source data, models and reasonableness of modeling assumptions, methods, results, and risks and uncertainties. Requirements to use particular models or statistical test plans are not to be specified. Solicitations should not cite any language, specification, standard, or handbook that specifies "how to" design, manufacture, or test for reliability. MIL–HDBK–217 or any of its derivatives are not to appear in a solicitation as it has been shown to be unreliable, and its use can lead to erroneous and misleading reliability predictions.*"

In 2004, the US Secretary of Defense William Perry [63] instructed the US armed services;

> "*Performance specifications shall be used when purchasing new systems, major modifications, upgrades to current systems, and non-developmental and commercial items, for programs in any acquisition category. If it is not practicable to use a performance specification, a non-government standard shall be used. Since there will be cases when military specifications are needed to define an exact design solution because there is no acceptable non-governmental standard or because the use of a performance specification or non-government standard is not cost effective, the use of military specifications and standards is authorised as a last resort, with an appropriate waiver.*"

This reduced the usage of military standards, including the MIL-HDBK-217 series, in favour of commercial specifications and standards and contributed to an increased commercial focus on physics of failure methods [216]. However, MIL-STD-217 based reliability predictions are still used and required contractually by many customers [176]. Practically, such empirical models are employed for early trade-off of competing designs during the system design process [219]. FIDES [183] is a recently developed hybrid methodology, with some elements based upon physics of failure and underpinned by assumptions such as the constant failure rate.

6.1.2 Concerns with application

Reliability requirements or goals are historically driven by the economic rationale of the product manufacturer, usually regarding warranty, or are dictated by the customer [303]. MTBF and MTTF are often used as an attribute or measure of hardware or software reliability, or as a basis for logistics planning of spare parts needs for repairable items, planning replacement of non-repairable items, specifying warranty periods, and planning for part obsolescence [303]. MTTF and MTBF are often used interchangeably and presented with no reference or relationship to how the product is used or the environment in which it is used other than limits of environmental extremes [303]. MTTF and MTBF are often specified for components without considering its actual function, and assumed to be constant; a real example was a Hard Disk Drive manufacturer that determined MTBF by powering and spinning the devices up, but without performing any loading, unloading, reading, writing, or searching operations [303].

The "infant mortality" period on the "bathtub curve" is a primary justification for Environmental Stress Screening (ESS) techniques such as burn-in, which continues to be used as a means of optimising reliability to satisfy warranty periods [198,244,245,355]. The intent of ESS techniques is to precipitate infant mortality (latent manufacturing) flaws so the fielded item will be at the beginning of the "flat portion" of the bathtub curve [355]. An industry "rule of thumb" is that ESS should not consume more than 5% of demonstrated endurance, durability, or life of the item [355].

Stress during burn-in accelerates defect mechanisms responsible for early-life failure but, for scaled semiconductors, also increases junction temperatures and result in accelerated aging [284]. Elevated junction temperature, in turn, causes leakage currents to increase and can result in thermal runaways. This can increase the cost of burn-in substantially, due to a need to optimise the burn-in environment to minimise thermal runaway while maintaining effectiveness of the burn-in procedure [284].

There is a body of literature providing specific criticisms or observations concerning applicability of traditional reliability methods to electronic (and other) systems, including;

- A statement [24] that "… the bathtub curve can model the reliability characteristics of a generic piece-part type, but not of an assembly, a circuit, or a system";
- Philosophical and analytical justifications [33,34,37,39,47] for adopting a "roller-coaster" curve for electronic components, with multiple humps suggestive of the dominant failure mechanism varying with time;
- Presentation [51] of field data showing different characteristics of the "bathtub curve" (this article sought to demonstrate applicability of traditional methods, but data presented did not achieve that to a high confidence level);
- Descriptions [85,159] of ESS and burn-in as expensive techniques with both benefits and impact on surviving components poorly understood;
- A statement [85] that thermodynamic considerations make it unlikely that the decreasing hazard region near zero is plausible for manufactured devices;
- A description [96] of ESS techniques like burn-in as an "art" based on engineering judgement and statistical analysis;
- An assertion [140] that the bathtub curve is applicable to "only 10% or 15% of applications";
- An observation [159] that no standard text provides compelling evidence of applicability of the bathtub curve for any manufactured products;
- Observations [284] that, with increased miniaturisation (or scaling) of semiconductor technologies, constant failure rates increase and wearout phases commence earlier.

6.2 Component reliability

6.2.1 Resistors

Resistors represent about 40-50% of all components in electronic circuits, but only 2-5% of the value of most circuits, so there is relatively little basic work to characterise their reliability, in comparison with work concerned with active components [325].

As dissipative elements, even ideal resistors will exhibit a fluctuating "noise" voltage across their terminals. Johnson–Nyquist noise is electronic noise generated by the thermal agitation of the charge carriers (usually electrons) inside an electrical conductor at equilibrium, which happens regardless of any applied voltage.

Resistors rarely fail unless physically or electrically overstressed. The most common failure mode is to an "open" state, in which its resistance increases significantly rather than to a "short" state corresponding to a reduced resistance. Some failure modes include;

- Carbon composition resistors and metal film resistors typically fail as open circuits;
- Carbon-film resistors may decrease or increase in resistance;
- Carbon film and composition resistors can open if approaching their maximum dissipation. This effect is possible but less likely with metal film and wire-wound resistors.
- If not enclosed, wire-wound resistors can corrode;
- Deposited metal film resistors aged in absence of oxygen show a reduction of resistance [4];
- Deposited metal film resistors exposed to oxygen at high temperature show an increase of resistance [4];
- The resistance of carbon composition resistors may drift over time and are easily damaged by excessive heat in soldering (the binder evaporates);
- Variable resistors become electrically noisy as they wear.

Drifts of resistance or tolerances directly affect stability and reliability of a resistor. Several ageing phenomena and other influences affect these parameters, including [325]; thermal influences from environmental conditions; chemical reactions associated with transport of load by resistive films, contacts, or coatings; chemical interactions between films, contacts, and coatings; and chemical reactions due to presence of water or humidity, particularly corrosion. The primary factors that influence reliability of resistors are temperature, power dissipation, and resistor type [102].

Failure or degradation modes associated with resistors in storage include [102];

- The values of composition-type fixed resistors drift, and these resistors are not suitable at temperatures above 85°C;
- Enamel and cement-coated resistors have small pinholes which bleed moisture, accounting for eventual breakdown;

- Precision wire-wound fixed resistors fail rapidly when exposed to high humidity or to temperatures at about 125°C.

6.2.2 Capacitors

Properties of capacitors in a circuit can determine the resonant frequency and quality factor of a resonant circuit, power dissipation and operating frequency of a digital logic circuit, energy capacity in a high-power system, and many other system characteristics. Real capacitors deviate from the ideal capacitor equation in a number of ways. Some of these, such as leakage current and parasitic effects are approximately linear and may be dealt with by adding virtual components to the equivalent circuit of the capacitor. All real capacitors have imperfections within the material that create resistance. This is specified as the equivalent series resistance (ESR), and adds a real component to the impedance. Similarly, leads of a capacitor add equivalent series inductance (ESL) to the component. These effects are usually significant only at relatively high frequencies.

If the conductors in a capacitor are separated by a material with small conductivity rather than by a perfect dielectric, then a small leakage current flows directly between them. The capacitor therefore has a finite parallel resistance and slowly discharges over time.

Tantalum capacitors are characterised by high charge per volume, low ESR and ESL, high stability with respect to voltage and temperature, and stability over long intervals [264]. These stability and reliability characteristics make them attractive for use in critical applications. However, the Ta_2O_5 dielectric is inherently thermodynamically unstable. Feasibility of stabilising both the dielectric and the Ta/Ta_2O_5 interface through suitable manufacturing technique has been demonstrated [264], suggesting such capacitors may be used at higher voltages or temperatures [264].

The predominant failure mode experienced with solid tantalum capacitors is the electrical short caused by impurities in the tantalum slugs and imperfections in the dielectric. These defects result in a phenomenon called scintillation, involving momentary shorts at dielectric imperfections, which can result in the capacitor healing itself, forming high leakage current, or permanently shorting [10].

Breakdown voltage

Above a particular electric field, the dielectric strength, the dielectric in a capacitor becomes conductive. This occurs at the device breakdown voltage and determines the maximum energy that can be stored safely in a capacitor. Capacitance and breakdown voltage therefore scale with dielectric thickness.

Geometry of the capacitor conductive parts (plates and connecting wires) also affects breakdown voltage. Sharp edges or points are associated with increased electric field strength, which can lead to a local breakdown which then "tracks" through the dielectric and causes a short circuit when it reaches the opposite plate.

A common breakdown mechanism is that the field strength becomes sufficient to pull electrons in the dielectric from their atoms thus causing conduction. Impurities in the dielectric, or

imperfections in the crystal structure of crystalline dielectrics, can cause an avalanche breakdown. Pressure, humidity and temperature also affect breakdown voltage.

Ripple current

Ripple current is the AC component of an applied source (such as a switched-mode power supply) with constant or varying frequency. Capacitors with high ESR ratings, such as electrolytic tantalum capacitors, are affected by ripple current frequency and magnitude, which cause heat to be generated within the capacitor due to current flow across resistive imperfections within the capacitor. The heat generated increases as the capacitor ages, eventually leading to an abnormal temperature rise and a chain reaction that generates more heat [79]. This causes an increase in vapour pressure of the electrolyte solution, emission of decomposition gases, and therefore pressure increase inside the capacitor case leading to failure with rapid release of high-temperature vapour and potentially further secondary damage [79]. Ceramic capacitors generally have no ripple current limitation, consistent with their having some of the lowest ESR ratings.

Capacitance instability

Capacitance of some capacitors decreases with component age. In ceramic capacitors, this is caused by degradation of the dielectric. The type of dielectric and the ambient operating and storage temperatures are the most significant aging factors, while the operating voltage has a smaller effect. The aging process may be reversed by heating the component above the Curie point. Aging is fastest near the beginning of life of the component, and the device stabilises over time. Electrolytic capacitors age as the electrolyte evaporates. In contrast with ceramic capacitors, this occurs towards the end of life of the component.

Capacitance is often assumed linear with temperature, but this breaks down at the higher temperatures. The slope may be positive, negative, or zero and may vary in a range for a given type of capacitor.

Capacitors, especially older components, can absorb sound waves. Vibration moves the plates, causing capacitance to vary and, in turn, inducing AC current. Some dielectrics also generate piezoelectricity and associated feedback effects. Conversely, the varying electric field between the capacitor plates can cause them to vibrate.

Capacitor failure

High-voltage capacitors may catastrophically fail when subjected to voltages or currents beyond their rating, or as they reach their normal end of life. Dielectric or metal interconnection failures may create arcing that vaporises the dielectric, resulting in case bulging, rupture, or even explosion. Capacitors used in RF or sustained high-current applications can overheat, especially in the centre of the capacitor rolls. Capacitors within high-energy capacitor banks can violently explode when a short in one capacitor causes sudden dumping of energy stored in the rest of the bank to the failing unit. High voltage vacuum capacitors can generate soft X-rays even during normal operation.

Proper containment, fusing, and preventive maintenance can reduce these hazards. High-voltage capacitors can benefit from a pre-charge to limit in-rush currents at power-up of HVDC circuits. This will extend component life and may mitigate high-voltage hazards.

Multilayer ceramic capacitors, being brittle material, are sensitive to failures due to mechanical or thermal stress with growth of flaws dependent on chemistry and microstructure [41]. It has also been demonstrated [41] that flaws can lead to electrical failures even if mechanical integrity of the capacity is unaffected.

6.2.3 Inductors

At some frequency, usually higher than the working frequency, some real inductors behave as a resonant circuit due to self-capacitance and at some frequency the capacitive component of impedance dominates. In addition to dissipating energy in the resistance of the wire, magnetic core inductors may dissipate energy in the core due to hysteresis, and at high currents show gradual departure from ideal behaviour due to nonlinearity caused by magnetic saturation.

At higher frequencies, resistance and resistive losses in inductors grow due to skin effect in the inductor's winding wires, in which current density near surface of the conductor exceeds current density at its core.

Real-world inductors act as antennas, radiating energy into surrounding space and circuits, and accepting electromagnetic emissions from other circuit.

6.2.4 Memristors

The development of memristors and considerations of potential applications are recent topics of research, so there is relatively little consideration in literature of their reliability characteristics.

A potential application of memresistors is in building nanoscale high density non-volatile memories and FPGAs [304,312]. Of these, resistance switching RAM (RRAM) is considered a promising candidate for future non-volatile memory [312,324,335]. In comparison with other non-volatile memories, RRAM promises advantages such as fast writing times, high densities, and low operating voltages [324]. Arrays using RRAM may be feasible as an ultra-high density synapse circuit for future large-scale neural networks [335]. RRAM does not suffer from some scaling limits associated with DRAM or flash memory, but is susceptible to some problems associated with device scaling, such as high defect rates, high device variability, and device ageing [335].

Memristors are also considered to be a means of obtaining equivalent circuit functionality of a transistor, using fewer basic devices or components, and therefore may provide a means of further miniaturising integrated circuits beyond what is possible with scaling of transistors [338].

6.2.5 Semiconductor technologies

Technology scaling associated with the progress of Moore's Law, §4.1, continues to provide performance benefits with increasingly smaller feature sizes and increasing power densities [195,265]. Microprocessors represent a leading edge of this progress.

New technologies and fabrication techniques increase sensitivity of manufactured product to process variations (spatial, temporal, within die, and between dies) that cause variability of manufacture process and of semiconductor material characteristics, such as random dopant fluctuations. Variations in operating voltage and temperature also affect key performance characteristics such as delay and energy consumption and can cause a range of intermittent, transient, and permanent faults. Future devices and systems are anticipated to be more vulnerable to transient faults due to radiation particle strikes [265]. Voltage variations are examples of intermittent faults that can appear or disappear seemingly randomly, with potentially high duration. Other effects like electromigration cause transient and permanent faults that persist until specific corrective action is taken.

Scaling within microprocessors accelerates onset of wear-out based failures and therefore shortens microprocessor lives [195]. Microarchitects traditionally have treated processor lifetime reliability as a manufacturing problem, and only considered reliability of mission-critical systems [265], while manufacturers qualify reliability during device design, circuit layout, manufacture, and chip test using application-independent methods based on worst case temperature and processor utilisation [195]. Conventionally it has been considered that monolithic processors implemented in silicon are reliable internally, with errors occurring only at the pins but, as feature sizes have dropped below 65 nanometre feature sizes, semiconductor devices increasingly exhibit high soft and hard error rates [208,194,234].

Ideally, device scaling would maintain a constant electric field by reducing element dimensions and voltage by the same factor $1/k$ ($k > 1$) [122]. This would mean [111] that each generational doubling of transistor density per unit area would be associated with;

- Gate delay reduced by about 30% (due to dividing physical separations by $\sqrt{2}$));
- Operating frequency increased by about 43% (multiplying by $\sqrt{2}$);
- Supply voltage reduced by about 30%;
- Reduced energy per transition by about 65% (dividing by $2\sqrt{2}$), corresponding to 50% reduction in power usage (assuming increased operating frequency and reduced supply voltage to maintain constant electric field);
- Reduced area and fringing capacitance (and total capacitance) by 30%, corresponding to an increase of capacitance per unit area by 43%

Supply voltages and threshold voltages, however, do not scale well with technology and manufacture techniques because of leakage, power voltage, and parasitic capacitance concerns [111,195,265]. For these reasons, scaling has actually been performed by reducing element dimensions while maintaining constant voltage [122]. This means that electric fields have grown with miniaturisation [122,194], which increases power densities and temperatures and therefore exponentially accelerates wearout failures within circuits [111,194]. Lower operating voltages

further contribute to increased sensitivity of component behaviour to manufacturing variations [88]. These problems are potentially exacerbated by increasing transistor count due to reducing thickness of gate and inter-layer dielectrics and interconnect current density. Such effects are having a strong impact on reliability margins as CMOS technologies have reduced below 45 nm [194]. Reliability margins are also being strongly affected by introduction of new materials and device technologies that seek to meet more demanding performance requirements, with several newer materials and devices having unknown reliability behaviours [194]. There is also a change from abrupt failure modes, in which failures are readily detected, towards more gradual failure mechanisms that involve some measurable parameter shifting over time, making it more difficult to identify failures [194]. These factors cause shifts of failure statistics, making it more difficult to apply conventional reliability techniques or to extrapolate from test results for predictive purposes [194].

Reliability of individual transistors is therefore reducing, while the number of transistors within a component or circuit that can fail is increasing [270]. To cope with this, modern processors employ some form of gating, most commonly clock gating. Dynamic (workload-driven) adaptation of processor resources and bandwidth, performed as part of on-chip power management [270], contribute to reductions of average power and temperature with the trade-off that they can introduce new on-chip effects, such as thermal cycling, that can again reduce reliability [265,270]. Reliability of LEDs depends mainly on chip quality, encapsulation type, and wire bonding reliability between the chip and anode terminal [327].

There are several unanswered questions about semiconductor reliability, including [265];

- What is the magnitude of faults due to process variations associated with increased technology scaling?
- What opportunities are possible by exploiting process variations?
- What are the causes of wearout in the typical lifetime of electrical components?
- What is the contribution of the microprocessor versus other components to total faults within a long-life system?
- Are system-level, software-level, or circuit-level solutions more technically feasible and cost-effective for addressing these concerns?
- Are schemes to anticipate faults preferable to schemes to detect and correct faults?

Degradations in electronics are often more difficult to detect or inspect than in most mechanical systems, due to device scaling and increasingly complex architecture of electronic products [213]. Furthermore, faults in electronic products may lead to degradation of functionality rather than simple failure or loss of that functionality. This increases the difficulty of detecting product degradation, tracking the progression from faults to failures, or implementing diagnostic or prognostic systems that can monitor either faults or even conditions in which faults occur [213].

Significant long-term reliability concerns affecting scaled semiconductor technologies include time-dependent dielectric breakdown (TDDB) of gate dielectrics, hot carrier injection, negative bias temperature instability (NBTI), electromigration, and stress-induced voiding [284]. Of these, TDDB and NBTI appear to be major reliability concerns affecting scaled semiconductors with TDDB, NBTI, or both considered to contribute to a number of concerns including digital

circuit speed degradation, FPGA delay increase, and analogue circuit mismatch [284]. Available data also suggests that technology scaling may cause wear-out failures much earlier than older technologies and an increase of the constant failure rate (the bottom portion of the "bathtub" curve [284]).

6.2.6 Connectors and fasteners

The reliability of electronic assemblies depends on the reliability of passive electrical connections between active components, as well as on reliability of the components [49]. Degradation or failure mechanisms of a contact or connector include corrosion, loss of normal force through stress relaxation, excessive heating leading to temperature related degradation, contamination, application of currents outside product specifications, and contact abuse (mating at inappropriate angles, pulling on cables, forced insertion, etc) [49]. Connectors may exhibit both mechanical failure modes (broken latches, bent pins, etc) and electrical failure modes (cross talk, leakage, change of resistance, etc) [49]. Data suggests, unsurprisingly, that the most common point of failure is with the contacts [45]. Common degradation mechanisms for connectors include corrosion, wear, and loss of contact normal force [56]. In particular, connectors that are repeatedly subject to mating/unmating cycles may experience significant reduction of contact force [275]. Nobel metal finishes typically provide greater durability than tin finishes, due to greater hardness and to requiring a lower normal force [56].

Fastening joint failure is also a significant failure mode in electronics packages [180].

6.3 Software reliability

Reliability practitioners traditionally describe software failures as "systemic", in contrast to describing hardware failures as "random". Flaws in software, with a possible exception of some security vulnerabilities, arise more from higher level design processes than from minor bugs in code [287]. If software fails on particular inputs due to a software fault, it will fail on those inputs until the fault is removed. Although failures are certain, given a specific set of circumstances, those circumstances have a probability of occurrence so software reliability is often expressed probabilistically [346]. The software failure process actually arises from random uncovering of faults during execution of successive inputs so software failures are characterised by inherent uncertainties and random characteristics, as it is not possible to predict all possible future inputs to the software under operating conditions [99]. This imposes severe limitations on ability to measure reliability through direct observation of program behaviour, due to costs of running a large number of possible test cases. This means considerable practical difficulty measuring and predicting reliability early in the life of a system, therefore difficulty with supporting early decisions about whether a system will be acceptably safe to operate [99]. It is also difficult to conduct detailed studies based on empirical software fault and failure data, because such data is not readily available, and is inconsistent, incomplete, or lacking [333].

Achieving software reliability involves four interacting technical approaches [76];

- Fault prevention: avoiding fault occurrences by construction;
- Fault removal: detecting, through verification and validation, the existence of faults and then eliminating them;

- Fault tolerance: providing, usually through redundancy, service complying with the specification even when faults have occurred or are occurring;
- Fault or failure forecasting: estimating the presence of faults and the occurrences and consequence of failures.

Obstacles to software reliability include [272];

- Novelty: software is used to implement functionality never implemented with another technology, and software is often a single-production item in comparison with other manufactured products;
- Non-repeatability: software products differ from each other;
- Difficulty: problems addressed by software require considerable intellectual effort;
- Complexity: growth in complexity of software has exceeded increase in complexity of physical components of a system;
- Human-based: most technologies used in software engineering are human-based, so software is subject to variations in human abilities.

6.3.1 Difficulty, novelty, and complexity

Practically, computing hardware remains more reliable than software, in part because of a philosophy of manufacturers to design hardware products so their reliability concerns remain "in the noise" in comparison with software concerns [265]. Systems designers are therefore less likely to encounter hardware failure than they are to encounter design flaws that will be triggered in all copies of some software in response to particular external conditions [99]. Software, however, is a complex intellectual product and its complexity and scope have increased significantly in recent decades, while the engineering techniques for producing software have only advanced moderately, at best [238].

The following discussion is summarised from [99], except where noted otherwise.

The design process in all branches of engineering involves a mixture of novelty and legacy. A new system will contain elements of design novelty when compared with earlier systems, but there will also be aspects of design carried across from those earlier systems. The novel aspects introduce the "value added" of a new design, but this is accompanied by an increased risk that new design faults will be introduced. On the other hand, reuse of "tried and tested" legacy components may reduce the risk associated with introducing new design faults, at the price of placing constraints on the system designer affecting provision of required new system functionality.

Significant system functionality and therefore complexity is often allocated to the software, so residual design faults are more likely to be found in software than in digital electronics. There is a tendency for system designers to take on tasks that are intrinsically difficult when building software-based systems. Basing a system on software frees the designer from some constraints associated with a pure hardware system. The likelihood of mistakes increases with difficulty of the task, resulting in the introduction of faults that cause system failure when triggered by appropriate input conditions.

The difficulty of tasks that a software system has to perform is often accompanied by a greater degree of novelty than in other branches of engineering. It is becoming increasingly common that software solutions are sought for previously unresolved engineering problems that were considered impractical using other technology or approaches. This imposes particular difficulty for systems with significant requirements for reliability, as there is little precedent for learning from previous work. While the addition of functionality to an existing software system can be accomplished as a natural and progressive evolution, software engineering tends to be associated with more "step changes" than other engineering disciplines. The increasing tendency to evolve a non-digital electronic control system into a software based system also implicitly involves a step change, despite often being viewed as a simple transition. In contrast, other engineering disciplines view such step changes as carrying significant risk.

These trends towards new and increased functionality in computer-based systems are, almost unavoidably, accompanied by increased complexity in the internal structure and external interfaces – particularly in software. Measurement of this complexity is an active research area but, regardless of measurement approach, complexity impedes understanding and comprehension of a system and therefore increases likelihood of mistakes. One of the most significant dangers with high design complexity is the difficulty of understanding: no single person can claim to understand the system completely, introducing uncertainty about the properties of the program – particularly its reliability.

Measurement of software complexity is also an active area of research. Practically, imperfect measures such as code size (e.g. lines of code) offers some measure of internal complexity of software while the size of a user manual offers some indication of complexity of its external interfaces. The average modern novel is about 10,000 lines long and is written in a naturally understandable language. In comparison, software is written in a constrained language that is less naturally understandable, and projects with hundreds of thousands or millions of lines are reasonably common. Moreover, the impact of complexity on understanding does not increase merely linearly with size.

The relative ease with which sophisticated tasks can be implemented using software has trapped projects into undertaking designs with excessive novelty and complexity, resulting in systems that are not only unreliable in operation, but so complex and poorly understood that their development becomes unmanageable so they are abandoned before becoming operational.

Exhaustive enumeration of all possible behaviours is generally only practical for trivial code software components: it is impractical to test software for all conditions it might meet in operation in order to guarantee it will be failure-free in operation.

These concerns are as important for formal software specifications as they are for software implementations: both formal specifications and implementation are digital specifications, written in a formal language, and both may contain faults. The only difference is that specifications will (ideally) be shorter and simpler than the corresponding implementation. However, most software organisations document neither requirements nor resulting specifications in any formal manner and, if they do write documents, typically don't update them as software evolves because the effort affects schedule [238].

The inherent discreteness of digital system behaviour further increases difficulty with assuring their reliability. In contrast with conventional mechanical and electrical systems, it is often impossible to extrapolate from evidence of failure-free operation in one context to support a claim that it will perform acceptably in another context. Standards offer practitioners a structure and roadmap to identify, specify, and quantify software quality [238,271]. With a notable exception of certifiable safety-critical applications, COTS (Commercial off-the-shelf) software solutions have become commonplace in several domains, including military, because they provide standardised functionality with more responsiveness, a short time-to-market, and (claimed) lower costs than custom-made applications [279].

For these reasons, a significant goal through the design and implementation process must be minimisation and control of complexity, in order to achieve and demonstrate reliability. Some complexities are unavoidable, as they result directly from the requirement, specification, or the design that matches the specification. Apart from clarifying or reducing the requirement set, these complexities cannot be addressed. Decisions early in the requirements capture and design process are therefore critical in ensuring reliability. Some other complexities are avoidable, resulting from inadequate skills, experience, techniques and tools.

6.3.2 Uncertainty in the failure process

There is a sense [99] in which execution of a program is completely deterministic: it is either fault-free, in which case it will never fail, or it contains faults, in which case the circumstances that cause it to fail once will always cause it to fail. In contrast, hardware components will inevitably fail given enough time, and can fail randomly in circumstances where they have previously worked perfectly.

Reliability practitioners traditionally describe software failures as "systemic", in contrast with hardware failures being described as "random". This terminology is considered misleading [99] as it describes the fault mechanism rather than the failure process (i.e. if software fails on particular inputs due to a software fault, it will fail on those inputs until the fault is removed). The software failure process actually arises from random uncovering of faults during the execution of successive inputs so software failures are characterised by inherent uncertainties and random characteristics, as it is not possible to predict all possible future inputs to the software under operating conditions in advance [99]. Knowledge of the software itself is invariably incomplete, so there is also uncertainty about what faults the software contains. It is therefore not possible to know which inputs, of those not yet provided, would trigger a failure.

Another little-understood source of uncertainty is caused by understanding of boundaries of the input space. Engineering judgments are made when building software systems based on engineering knowledge that certain combinations are not possible, so the software need not be designed to handle them. Practical experience [99], however, shows that many system failures categorised as software failures only arose when the software was in a state it was never designed to handle. Even with systems designed to cope with unexpected states, there is always the potential for unexpected input conditions to create a failure.

6.3.3 Measuring reliability

Software reliability measurement (collecting and analysing data about observed reliability) and prediction (using a model to forecast future reliability) are approaches to quantify software quality [238]. With inherent uncertainty in the software failure process, it is necessary to express reliability requirements probabilistically [99]. The specification will depend on the nature of the system, such as whether the system is required to respond to rare demands or it is a control system that must continuously keep a physical system within acceptable operating bounds. Availability of a system may also be a factor in its reliability: there is then a need to ensure availability of the system when needed, and reliability of its actions when available. In general, [99], it is possible for there to be several distinct reliability (or safety) requirements associated with different types of undesirable events.

There are severe practical limitations on the levels of reliability that can be measured statistically through direct observation of the failure behaviour of a program. If a low probability of failure is required, and the time to perform individual tests, or the time for the test harness to produce particular inputs, is significant then direct observations of the software failure will require significant time.

While there is evidence in several industries of software that has exhibited failure-free working for long periods of operational usage, such systems are considered to only provide indirect and weak evidence that future derived systems will exhibit required reliability, as there will typically be unique aspects of new systems and the development process for new systems is likely to differ considerably from earlier systems [99].

6.3.4 Software estimation

There have been a number of reports that survey software effort estimation, in order to identify how many software development projects suffer cost or schedule overruns, or project failures (i.e. cancellation). The Standish Group "Chaos" series of reports, for example [68], are regularly cited in scientific reports, presumably as they offer compelling statistics about the need to improve estimation techniques (e.g. only 9% of projects successful, 31% of project cancelled, average cost overrun 89%). However these reports are not considered to represent a scientific survey and information about how organisations and projects were selected is not disclosed [174]. These reports also do not agree with other surveys [174,224] which suggest that 60-80% of software projects encounter cost or schedule overruns, but the average cost overrun is a (substantially smaller) 30-40% [174]. For example, a recent review of US Defense Acquisitions [308] noted that;

- fourteen of thirty-three assessed programs provided data indicating that estimates of number of lines of code have grown 25% or more since program commencement;
- on average, these high growth programs experienced a 40% increase in research and development costs and a 38 month delay in fielding Initial Operating Capabilities in comparison with, on average, a 12% increase in research and development costs, and an 8-month delay for programs with less software growth.

Little work has been performed to analyse the reasons for project overruns, and a recurring problem in surveys appears to be respondents being biased and/or affected by selective

memory, rather than being uninvolved or impartial reviewers [174]. There are fewer studies concerned with understanding the contributions or perceptions of software developers involved in software project success or failure. It is suggested [109] that most surveys may be too narrowly defined, create negative perceptions about software developers, and that there may be instances when failure statistics are used as fear-based advertisements for consultant services or quick-fix techniques/tools.

Not withstanding this, a common and recurring theme [109,145,203,227] is that stakeholders, project managers, and developers, often have different criteria for success. Criteria for success of a software project include [145]; meeting agreed business objectives; completion on-time and on-budget; degree to which the project achieved its technical goals; reliability; maintainability; meeting user requirements; user satisfaction; effective project teamwork; and professional satisfaction.

Case studies have found the most recurring common ground between developers and project managers is having the best interests of users and, to some extent, customers at heart in terms of user satisfaction [109,227], although this is stronger when the project manager has a background as a developer [109]. Developers are likely to view a project as successful when planning of cost and schedule is comparable to normal industry standards (avoiding excessive unpaid overtime) but will perceive a failure when faced with unrealistic schedule expectations, lack of resources, and poor understanding of scope at the outset [145] – even if management considers the project meets business goals.

Several studies evaluate project successes and failures using misnamed measures, and therefore portray them inappropriately. One paper [109], for example, presented an initial estimate of code size of 50000 LOC versus a final product code size of 65000 LOC as a 130% under estimation, in comparison with the mathematically correct 23% (or a growth of 30%). Such basic errors would increase fear, uncertainty, and doubt among practitioners, managers, and stakeholders through over or under estimating impacts of failures or extent of successes.

6.3.5 Software metrics

"Software metrics" is a misleading collective term that describes a range of activities concerned with measurement in software engineering [110]. The classical definition is of numerical values that characterise software code, but the broader definition includes models that predict software resource requirements and software quality, and includes quantitative aspects of quality control and assurance, such as recording and monitoring of software defects through development and testing [110].

Early work on software engineering noted that significant effort was needed to prove correctness of even small programs [13]. This may explain introduction of lines of code based metrics (LOC or KLOC) in early publications [12,17,20], despite recognised drawbacks and a need for more discriminating measures being emphasised by diversity of programming languages [110]. There is therefore a significant body of research concerned with software complexity measures [e.g. 14,22,30,31,35,57,70,91,92,154,161,270,293] and/or of software size [e.g. 23,43,50,61,126,146].

Not withstanding this body of research, adoption of software metrics in industry is low [110]. Survey results indicate metrics are adopted by 45% of respondents, with tracking and performance metrics used more widely than quality or estimation metrics, and software size data only collected by 21% [105]. The majority of industry adoption is claimed to be based on metrics developed in the 1970s [110], with mismatches between research and industrial application possibly attributed to a number of factors [110];

- *Irrelevance in scope of research.* Most of the body of research can only be applied, or the metrics computed, for small programs, but most industry interest in metrics is concerned with large programs. Some models from research rely on parameters that cannot be practically measured.
- *Irrelevance in content*. Industry is in need of metrics relevant to process improvement, but research has concentrated on detailed code metrics.
- *Industry motivation is low*. Companies most commonly introduce metrics when things are bad, or to satisfy some external assessment body. There is little known about the effectiveness of software, and very few convincing success stories of long-term payback from using metrics. Collection of metrics would typically add an overhead of 4-8% to software projects, and will be one of the first compromises when deadlines are tight. Commitment of technical staff involved in development and testing is needed, but there are no easy ways to motivate such commitment.
- *Industry metrics activity is poorly executed.* There are many examples of industry practice that ignore best-practice guidelines for data collection and analysis, and apply metrics in ways known to be invalid.

The purpose of software metrics is generally considered to be to improve ability to monitor, control, or predict software attributes, and of the commercial software development process, implying that industry adoption is a significant factor in success of any software metric or method [110], although this observation must be treated cautiously as the case study on which it was based [83] drew data from only two organisations. Metrics that appear to meet this measure of success are [110] metrics based on lines of code (despite their known drawbacks); metrics related to code defects obtained through code inspection; and metrics based on cyclomatic complexity [14].

Common attributes of these metrics are that they are not particularly discriminating but are relatively easily computed [110]. However, function point analysis [23,32] is used relatively frequently by industry [110] despite being difficult to apply properly [73,82].

In practice, there is also no universally applicable software reliability growth model [42]. Although it is claimed [52] that stochastic reliability growth models can accurately predict reliability of a software system if sufficient failure data is collected, this is of little utility if predictions are needed before the software is operational [110]. Practically, software metrics are not consistently defined and interpreted, so achieved reliability measures may vary between applications, yielding inconclusive results [243].

Case studies have also found little support for hypotheses or beliefs that are typically used to justify employment of commonly used metrics [110]. While the results are not necessarily general, some evidence suggests [40, 110] that;

- Complexity metrics are closely related to size metrics;
- Complexity metrics and size metrics are both reasonable predictors of the absolute number of faults, but poor predictors of fault density;
- Complexity metrics are poor predictors of which modules will be fault-prone before release (i.e. they are inherently poor at predicting what they are intended to predict);
- The belief that "a small proportion of modules in a system accounts for most of the faults and are likely to be fault-prone both pre-release and post-release" is incorrect.

6.3.6 Challenges of concurrent software

Hardware designers now find it relatively easy to design multicore systems but these systems provide new challenges for programmers [320]. Research into implicit (i.e. hardware-supported) techniques, such as speculative multithreading or automatic parallelisation of loops, is not currently promising so achieving the claimed or promised performance benefits for multicore systems requires concurrent software [278]. This presents a disruptive shift of emphasis towards concurrent (or parallel) software because, although traditional sequential programming is hard, concurrent programming and debugging is significantly more difficult [153,205]. Humans experience more difficulty reasoning about concurrent code than about sequential code [205]. Development of concurrent software has historically been relegated to a niche requiring specialist, even heroic, effort [278]. Concurrent programming is therefore described as revolutionary [205] or as a paradigm shift [276]. Verification of concurrent systems is also considered one of the most challenging areas of software verification, due to the many ways in which concurrently executing processes may be interleaved [231].

The "parallel programming problem" has been addressed, in high-performance computing, for at least 25 years but only a small number of specialised developers actually write parallel code [278]. Even for numerically intensive applications, where parallel algorithms are relatively well understood, professional software engineers almost never write parallel software [278]. Few engineers currently know how to program multicore processors and state-of-the-art techniques are not user-friendly due to the effort needed to explicitly design and debug multicore programs [331]. Achieving performance and scalability of parallel code is currently labour intensive, and the code is usually not portable between hardware platforms or even to later implementations of the same instruction set architecture [234].

Parallel programming research has historically been dominated by an engineering approach: build "it", show "it" works, and move on [278]. A deliberate scientific approach, based on hypothesis development and hypothesis testing, development of predictive theory and models, and peer review continues to be absent [278]. Parallel programming has therefore developed along informal and empirical lines, and lacks any body of theory to guide research or engineering practice [278]. Programming is inherently a human endeavour, so the core of any body of theory must be informed by how programmers think and be based on some human-centred model [234,278].

There are several unresolved performance-related issues with multicore systems [319]. System designers are experiencing performance drops when moving single-core multiprocessor designs to a multi-core processor [319]. Porting software between different parallel architectures has always been difficult, and the wide range of multicore architectures currently available increases this difficulty, although some recent software technologies do support multiple architectures [345]. Programming languages and modelling methods need to evolve to support safe concurrency in a way that allows "ordinary programmers" to write efficient and trustworthy concurrent programs, while providing scalability and evolvability [276]. Multicore processors also increase demands on memory and cache performance, performance of techniques for maintaining cache coherency, and memory bandwidth [337]. Although capacity of memory chips continues to grow, the number of processor cycles required to access main memory is also growing, and is considered a likely limiting factor for the performance that can be achieved by multicore processors [234].

Any parallel or concurrent system may experience synchronisation errors (interleaving errors, deadlocks, livelocks), race conditions, violations of order of precedence of operations, and timing errors [153]. Concurrent systems are also particularly subject to the "probe effect" [26], a software equivalent of Heisenberg's uncertainty principle, as any code added to or removed from a system for monitoring purposes can affect execution times, resource usage, and other behaviours [153].

From a software perspective, multicore processors might be viewed as an expansion of symmetric multiprocessor systems (SMP) that have existed for some time [337]. However, promised benefits of user-observed responsiveness, increased task-level throughput, and higher performance of multithreaded applications can only be achieved if the system software stack and tools are in place to support these improvements [337]. Modern operating systems, for example, will need to use finer-grained locks to avoid contention for key data structures used to better manage scheduling, process migration, I/O requests, and other resource allocations [337]. Operating systems schedulers require enhancement so they do not become bottlenecks in the juggling of active processes across significant numbers of cores or processors [337].

6.4 Assurance and certification

Any safety related industry tends to have a conservative approach to innovation [172,314]. The systems are custom-made for a relatively small market, compared with the overall market, so there is a comparatively low investment and corresponding low rate of innovation [172]. The pace at which technology used in safety related applications evolves tends to exceed the rate of evolution of regulations, policy, and advisory material. This is particularly true in the airworthiness domain [296] and continues to be evident with advances in the area of Integrated Modular Avionics (IMA) [235,296].

The concept of system safety relies on a risk management strategy based on identification, analysis of hazards, and application of remedial controls using a systems-based approach [44]. A systems-based approach to safety requires the application of scientific, technical and managerial skills to hazard identification, hazard analysis, and elimination, control, or management of hazards throughout the life-cycle of a system, program, project or an activity or a product [44].

Safety-related systems may cause or allow accidents only through physical systems they are designed to control or protect [169]. Computer-based safety systems are generally complex, comprising sensors, actuators, processors, and program logic [169]. Undesired behaviour of a computer-based safety system may contribute to an accident, but the same behaviour in a different environment may be neutral or beneficial so safety systems have implicit or explicit requirements, captured as specifications, that determine what is considered safe behaviour in the specific environment for which they have been developed [169].

Every system with identified safety implications inherently requires a reasoning that provides rationale, expert opinion, and justification that allows it to be certified and put into production [334]. Although COTS and reuse of components can provide significant benefits from a system design viewpoint, they pose challenges for safety-assurance of the system as general purpose components are generally not developed with the safety-context of a particular system in mind [334].

6.4.1 Process based and evidence based standards

Safety assurance standards applied to software systems have often been process-based (for example, IEC 61508, DEFSTAN 00-55 (Superseded) [87], and RTCA/DO-178B [53]). These standards list pre-determined activities that, when followed by developers, are considered to result in an acceptably safe system, with a default position that higher levels of risk require provision of more evidence and greater scrutiny [334]. Software failures typically result from systemic (design) faults introduced during development so software safety standards have often sought to define requirements and constraints for the software development and assurance processes with the intent of reducing the number of faults specifically introduced by the process (e.g. increased rigour in specification) and increasing the number of faults revealed by the process (e.g. increased rigour in verification) in order to ensure faults may be removed [230].

Process-based standards do not, contrary to some claims, necessarily prescribe the nature of evidence that must be provided. For example, RTCA/DO-178B [53] identifies important steps in a development process [343] but is more concerned with specifying objectives that are closely related to the software lifecycle [285,343]. All but three of these objectives are described in a manner permitting flexibility in how they are satisfied [285].

Recent evidence-based standards, like DEFSTAN 00-56 Issue 4 [249] require the developer to assure the safety of the delivered system through structured reasoning, with provision of a safety case, and do not specify a list of activities to be followed [334]. A goal-based process is considered to allow greater flexibility, as developers are not instructed to use any specific set of techniques, but can include unpredictability of planning as developers do not have any pre-defined set of activities to follow [334].

Avionics companies and designers, facing rigours of DO-178B guidance, began moving system functionality from software to hardware, effectively side-stepping the need to comply with RTCA/DO-178B requirements [356]. This is considered [356] to have been a significant driver for the development of RTCA/DO-254 [115].

RTCA/DO-178B [53] is the preferred standard to be applied to software systems in the Australian military context [285]. It is also widely accepted by National Airworthiness Authorities such as FAA and EASA, where there is evidence that it is effective [229].

6.4.2 Usage of formal methods

For some time there has been desire [e.g. 169] for formal methods in relation to safety-critical systems, even when there was a lack of documented factual evidence about their efficacy [71], leading to criticism of lack of support (or mandating) of formal methods in certification standards. A significant proportion of this desire has been from academics, researchers, or consultants in the domains of high integrity systems or formal methods [285]. Practically, however, formal methods are not yet universally applicable to all functions, systems, and their failure modes [285], although practical industry experience is increasing over time [e.g. 343].

Within Australian Defence, the Director General Technical Airworthiness (DGTA) encourages the use of formal methods where appropriate, and requires that its application should be proposed and negotiated through the Plan for Software Aspects of Certification (PSAC) [285]. However, DGTA also recognises that safety-related software errors arise most often from discrepancies between documented requirements specifications and the requirements needed for correct functioning of the system, and also from misunderstandings about the software's interface with the rest of the system [285]. Accordingly, the extent of application of formal methods needs to be carefully balanced with software safety analysis, and is certainly not a substitute for such analysis [285].

6.4.3 COTS

Commercial off-the-shelf (COTS) products are officially allowed by RTCA/DO-178B [53], but no requirements are waived, increasing difficulty with exploiting COTS products that have not been developed with DO-178B requirements in mind [172]. RTCA/DO-278B, a standard for Air Traffic Management based on RTCA/DO-178B, specifically defines processes for planning, acquisition, verification, configuration management, and quality assurance of independently developed or pre-existing COTS products, including the usage of COTS service experience [172].

The electronic flight bag is a COTS-based hardware platform that supports several independent software applications, possibly simultaneously. The FAA has produced specific guidance, Advisory Circular 120-76A [160]. "Type A" applications include pre-composed presentations of aviation data. A specifically listed set of applications is referred to as "Type B", and some of these interactively manipulate and present aviation data. "Type A" and "Type B" do not require a DO-178B approval process but "Type C", consisting of all other applications, do [160].

6.4.4 Complex Electronics

Complex electronics are categorised as neither hardware nor software, but as "soft hardware" [259]. Some concerns affecting assurance of complex electronics are [259];

- ASICs and FPGAs are used to avoid rigours of the software assurance process, in particular bypassing fundamental verifications;
- Complex Electronic devices are designed and programmed by electronic engineers (designers), often without quality assurance oversight or configuration management control of the designs. Additionally, the development process may not be well defined or followed;
- ASICs, FPGAs, and SoC may contain embedded microprocessor cores with user-supplied software, effectively combining electronics and firmware onto one chip. The presence of the firmware (i.e. software) is not always obvious to assurance personnel;
- Hardware designers increasingly utilise high-level software languages to define complex electronic designs, either in whole or in part;
- Hardware quality assurance personnel may not be fully cognisant of the functions, potential problems, and issues with these devices;
- Meaningful verification efforts require knowledge about the complex electronic device and about the tool suite used to create and implement the design.

6.5 Non-operating reliability

Design of equipment often considers that reliability concerns result from being in service [94]. Modern electronic systems, including safety critical embedded systems, spend a significant majority of their life in a non-operating state [69,94]. The two main non-operating conditions are dormancy and storage [69].

Non-operating conditions are typically viewed as benign [18,94], but may actually be quite stressful on electronic equipment [69,94] as the equipment comes in contact with numerous environmental stresses, which may be either natural (e.g. adverse weather) or manmade (e.g. mishandling or abuse), particularly for equipment that must be inactive in its intended field environment [69]. Systems designed for high operating reliability do not necessarily function well (or at all) after long periods of exposure to non-operating conditions, particularly if potential non-operating failures are not considered in the design of the equipment [69].

6.5.1 Dormancy and storage

Dormancy is defined [69] as the state in which equipment is in its normal operating configuration and connected, but not operating. Equipment in the dormant state is generally characterised by connection to a functioning system so it is immediately ready to operate on demand and by its non-operating condition where there is reduction or elimination of most of the physical, electrical, or environmental stresses associated with the operating condition [18]. Equipment in the dormant state may be periodically cycled on and off but, during dormancy, the electrical stresses associated with operational conditions are usually eliminated or reduced [69]. The dormant state does not include equipment operating at very low levels of its function

(e.g. power output) or equipment that has been disconnected or in storage [18]. Built-in Test Equipment in the military domain is estimated to spend over 99% of calendar time in the dormant condition [18].

Storage is defined [69] as the state in which the system, subsystem, or component is totally inactive, and it resides in a storage area. A product in storage may have to be unpacked and connected to a power source in order to be tested [69].

6.5.2 Non-operating environments

A system may be situated in numerous non-operating environments during its lifetime. Some of these may cause harm to a system, and others are of negligible importance. Systems may lie inactive in the field (subject to possible harsh environmental factors) or elsewhere (e.g. in route for maintenance).

During these times, systems may experience environmental stresses which may be natural (such as adverse weather) or man made (such as mishandling or abuse). Possible environments, other than the field, include [69];

- **Storage.** While in storage, parts or systems may or may not be in a controlled environment. Factors such as moisture from condensation and diurnal temperatures, which can range from –50C to 75C, can become a concern. Temperature variations on components can be increased in poor ventilation conditions. Thermal expansion coefficients vary greatly for different materials so surface mounted ICs can literally pop off their circuit board due to extreme temperatures. The two primary causes of mechanical stress in storage environments are inertial and thermal-mechanical interactions [15].
- **Receipt Screening**. Before being placed in storage, parts are subjected to receipt screening that may involve removal of protective coverings and exposing affected sites to environmental stresses. Human or mechanical handling may also cause shock or particulate contamination.
- **Repair/Modification**. While systems are undergoing repair, they experience stresses normally associated with manufacturing as well as stresses from transportation, storage, and packaging. This can include mechanical shock, physical deformation, and electromagnetic radiation. Replacement parts may also be introduced with reliabilities different from those being replaced.
- **Test.** Systems and parts that need to be tested or re-certified are subject to similar environments as those cited above for Repair/Modification.
- **Movement/Transportation**. Parts and systems being transported can experience a broad range of adverse stresses such as thermal and biological exposure, acceleration, vibration, mechanical shock, radiation, pressure, and physical impact. Facilities at intermediate stops are more likely to have personnel inexperienced with the handling and care of certain parts.

6.6 Product ratings

Electronic equipment designers use part manufacturers' data sheets to help select parts [120]. The data sheet is a snapshot of information that the manufacturer chooses to divulge. Not all data sheets are public. The part manufacturer uses the data sheet as marketing literature, a technical fact sheet, and as a business document used to provide disclaimers and limitations on the usage of a part. The content of a data sheet is not standardised, and there are significant variations in content and format of part data sheets, both among manufacturers and between parts from one manufacturer. Published data typically includes [120];

- Part type and category;
- Information on outlines, terminal identification and connections, case material, and lead finish;
- Electrical, thermal, and mechanical ratings;
- Electrical and thermal characteristics;
- Mechanical data;
- Environmental and/or reliability data;
- Graphical representation of characteristics.

Data sheets may be issued and updated at various stages of product development [120].

Part data sheets typically provide absolute maximum ratings (AMRs), which represent a limit for "reliable" use of a part, and recommended operating conditions in which electrical functionality and specifications of a part are guaranteed [120,151]. AMRs typically specify operational, environmental, and other parameters such as power, power derating, supply and input voltages, operating temperature, junction temperature, or storage temperature [120].

Recommended operating conditions (ROC) typically include voltage, temperature ranges, input rise and fall time, and similar parameters [120]. These can differ substantially from nominal operating conditions in military aircraft [131].

Part manufacturers have differing views on the use of a part between AMR and ROC, and these differences are not reflected in product data sheets [200]. Manufacturers often, in practice, state that part performance is not guaranteed below the AMR, but useful life of the part will not be affected [120,151]. Other manufacturers state that part performance is not guaranteed above ROC but useful life is unaffected [200]. Others state that parameters within the ROC are not guaranteed at or near AMR and that, if the part is used in such conditions over a long period, there are reliability concerns affecting useful life [200]. These observations suggest that part temperature ratings are set for electrical performance reasons rather than for package or device reliability [151,200].

Derating

Derating is the operation of a part outside its rated operating limits, and may involve intentional reduction of applied stress on a component to assure reliability [100,211], increasing strength of a part for the application [100], or operating at reduced levels of functionality [106].

Uprating

Part manufacturers typically guarantee electrical parameters (usually as typical, minimum, and maximum) of parts only when used within recommended operating conditions and standard circuit conditions [120,294]. Manufacturers usually rate parts for operation in the "commercial": 0 to 70°C and, to a lesser extent, in the "industrial" –40 to 85°C operating temperature range [218,294]. These ratings may satisfy demands of computer, telecommunications, and consumer electronics industries, but there is demand for parts rated beyond the "industrial" range from aerospace, military, oil and gas exploration, and automotive industries [294]. However, this demand is insufficient to attract and retain the interest of major semiconductor manufacturers to make those parts [218,294]. Wide temperature range parts are therefore becoming obsolete, and not being replaced by functionally equivalent parts in the same temperature range [218,294].

Uprating is a process of assessing the capability of a part to meet functional and performance requirements for use outside the manufacturer-specified temperature range [86]. Uprating requires following of documented, controlled, and repeatable processes that are integrated with the parts selection and management plans [218]. Uprating is often applied when a MIL-SPEC part no longer has viable customer demand while commercial or industrial versions of the part continue to be available in high volume [294].

The main areas of risk addressed by uprating are [86] capability of the die to operate in the desired environment without physical degradation, capability of the packaged component to withstand exposure to the desired environment without failing, and capability of the component to perform its required electronic function in the desired environment.

Risks affecting die reliability may be managed or mitigated by applying due diligence during selection of a manufacturer, to ensure appropriate design rules were applied at the transistor level. Risks affecting package reliability can be managed or mitigated by evaluation of the qualification tests employed by the manufacturer. If risks to die reliability and package reliability have been addressed, then uprating can often be viewed as a concern of electrical performance rather than a reliability concern [86].

The three main methods of uprating are parameter conformance, parameter recharacterisation, and stress balancing [121,218,294].

Parameter conformance is an uprating process that tests the part to assess if its functionality and electrical parameters meet manufacturers' specifications over the target temperature range. Electrical testing verifies compliance with manufacturer-specific parameter limits. "Go/no-go" tests are performed at upper and lower target temperature limits, using the manufacturer-specified test setups, possibly with a margin of safety by testing in a range wider than the target temperature range.

Parameter re-characterisation mimics the part manufacturer's characterisation process, and statistically characterises part functionality and electrical parameters over the target temperature range, possibly leading to re-specification of the parameter limits (i.e. an update of the manufacturer part datasheet).

Stress balancing is a process that maintains least one electrical parameter for a part below its maximum allowable limit, in order to reduce heat generation and therefore allow operation at higher temperature. The stress balancing process exploits the possibility that the application may not require the full range of device capabilities, allowing trade-off between power and operating temperature. Testing is therefore performed to determine the relevant trade-offs for each specific application. The performance of active electronic parts is determined by;

$$T_j = T_A + P\theta_{JA}$$

Where T_j is the junction temperature, T_A is the ambient temperature, P is the power dissipation, and θ_{JA} is the junction to ambient thermal resistance [137]. If junction temperature is kept constant, the (temperature-dependant) performance of the part should not change. A part may therefore be used in a higher ambient temperature if power dissipation is increased, introducing a trade-off of electrical characteristics as power dissipation often depends on parameters such as operating voltage or frequency.

There are trade-offs between uprating cost and the risk of subsequent failure when choosing between the three uprating methods above for avionics applications [121]. Costs for uprating are often split into engineering cost and testing costs, with testing costs usually dominant. Parameter characterisation is likely to carry highest cost of the three methods, but offers lowest risk because of more comprehensive testing than the other methods. Stress balancing only involves sufficient testing to check applicability of theoretical results, so has slightly higher risk than parameter recharacterisation but lowest cost of the three options. However, stress balancing is only applicable for uprating to higher temperature limits and requires the part to have significant power dissipation which may be traded off against some performance parameter. Parameter conformance is associated with highest risk, as testing is limited, with both high cost and high cost variability because of potential need to repeat testing on new lots of parts.

Some organisations use assembly-level testing as a means of uprating parts, rather than uprating individual parts [294]. Such approaches are unique to the specific assembly, so there is no guarantee that a particular part can be used in another assembly. If the uprating process fails it is then necessary to perform part-level testing, in order to isolate the cause of observed problems. If an uprated part is replaced during maintenance, it is often necessary to re-test the whole assembly. Retesting may also become necessary if a non-uprated part is replaced, due to tolerance build-up in the circuit and erosion of design margins within the assembly.

Uprating is not always possible, and any design or manufacture process change may render a part unable to be uprated. Manufacturers reserve the right to make such changes without changing part designation [120]. Supplier restructuring or sub-contract manufacturing changes (outsourcing) can make a previously uprated part no longer able, or only partially able, to be uprated [294]. The semiconductor industry employs die shrinks in order to put more devices on a single wafer and maximise profitability and, on average, active and developing electronic

parts undergo die shrinkage every six months [218]. Manufacturers typically do not issue change notices unless there is a noteworthy change of form, fit, or function within the manufacturer's recommended environmental operating limits [218]. These concerns yield configuration risks for products with uprated parts, making an identification system for uprated parts (e.g. part codes, identification numbers) necessary [294].

An incomplete or inaccurate operating profile that does not include representative times spent at different temperatures may yield unacceptable reliability of a product or component parts, and these effects can be amplified by extended operating times at high or low temperatures [294].

6.7 Transient and intermittent faults

Faults experienced by semiconductor devices may be categorised as permanent, transient, or intermittent. Permanent faults reflect irreversible physical changes, while transients are induced by temporary environmental conditions, and intermittent faults occur due to unstable or marginal hardware [269]. Although transient and intermittent faults appear similar, they have different observable characteristics of activation and deactivation [269];

- Intermittent faults occur repeatedly at the same location, while transients randomly affect different locations in the device;
- Replacement of an offending circuit can eliminate an intermittent fault, but transients cannot be addressed by repair;
- Errors associated with transients and intermittent have random characteristics, but errors associated with intermittent sources often occur in bursts.

Causes of transient and intermittent failures include alpha particles [97], power supply fluctuations [97], loose connections [97,206], corroded or dirty connector contacts [206], partially defective or deteriorating components [97], and poor hardware design [97]. Such failures can be very difficult to isolate or identify [97]. Increasing scaling, or miniaturisation, of semiconductors and higher circuit complexity are expected to increase likelihood of intermittent faults, despite extensive use of fault avoidance techniques [269].

Any temporary deviation from nominal operating conditions of a circuit or device and subsequent recovery of the function is often referred to as an "intermittent" [148,274]. A circuit or device exhibiting such a deviation is also often labelled as an intermittent. Three basic types of intermittent occurrence in electronic systems or circuits are engineering, test void, and connection [64].

Engineering, or design, intermittents occur when a normal operating event causes a circuit to temporarily deliver a wrong output. Such events occur because of complex interactions between system components, and are often related to specific timing events or requirements. Contributors include switching transients, induced EMF, load changes, ground loops, leakage through circuit boards and conformal coatings, software, and poor initial design. These usually occur as a syndrome that is difficult to isolate and correct.

When a unit under test (UUT) continuously fails to perform it's function in an operating system, or fails a high-level test, and the malfunction is not detectable at a lower level of testing, it is designated a Test Void intermittent. Since the failure can be repeated in a high-level testing, test void intermittents may often be fixed by addressing deficiencies or lack of coverage in lower level test programs. This category of intermittent therefore receives the most engineering attention, because the problems are isolatable, fixable, and the results are quantifiable [64].

Connection intermittents are caused by a temporary break in a circuit's continuity, and can result from loose (cold) solder joints, oversized or worn connector pins, heat sensitive components, broken or frayed wires, damaged circuit board traces, noisy components, corroded connections, or loose screws [64]. These types of defects increase over life of a product, based on amount of wear encountered, and are often triggered by stress factors in the system's operating environment [64]. Connection intermittents grow over time until they become a major source of failure in older systems. Initially they are characterised by small, short duration fluctuations, voltage drops, or electrical noise that do not affect overall function of a system. As amplitude and duration of the fluctuations increase, random system failures occur. Traditional ATE (Automated Test Equipment) often cannot detect these fluctuations until the strength or duration increases substantially [64].

6.8 "No Fault Found" phenomena

A failure observed in the field may not be duplicated in subsequent fault-finding activities [97,206]. Failures observed in depot testing may also not be observed in subsequent engineering tests [135,206]. Some such failures occur while the component is under stress conditions, but seldom occur under more benign conditions on a test bench [206]. Others are associated with transient effects during operation that are not recreated in subsequent testing [97,135,246].

These failures are described in field failure databases using terms such as "cannot duplicate" (CND), "retest OK" (RTOK), "no fault indicated" (NFI), "no fault found" (NFF), "erroneous removal" (ER), "cannot verify" (CNV), "no evidence of failure" (NEOF), "no problem found" (NPF), and "no trouble found" (NTF) [54,64,97,135]. Notably, such descriptions and interpretations often imply that a problem never existed, suggesting that a unit under test was erroneously removed, the technician or operator who reported the problem may have made a mistake, or the problem has amazingly disappeared [64].

Such failures have been associated with between 25% and 75% of all reported avionics removals in different studies [54] or as high as 85% [97] or even 90% [246]. The average rate of such occurrences reported in literature is about 50% [246]. Occurrences are traditionally treated as an error in previous fault reports, leading to return of the affected component into the supply system without further investigation [246]. Some components may exhibit a continuous loop of rejection from an aircraft followed by NFF classification in subsequent workshop testing and are referred to as "rogues" [167] or "bad actors" [206].

The impact of NFF occurrences may be measured as the proportion of repair budget wasted by not finding the root cause of faults [135,246]. NFF increases the burden on the supply and maintenance system, which may be measured in terms of volumes of spare parts inventories, increased pipeline time, and increased cost of work and manpower [97,246]. The NFF

phenomenon can account for up to 90% of the total maintenance costs related to aircraft electronics [97,246]. Rogue units contribute to a further increase in repair expenditures [135,246].

NFF consequences associated with mission aborts, flight delays, and cancellations affect ability to achieve availability and dependability [167,246]. Guaranteeing the failure rate of a product may be insufficient to meet evolving dependability requirements, particularly those related to "power by the hour" or "maintenance free operating periods" if products or systems are affected by high NFF rates [246].

In the aerospace industry, NFF occurrences are sometimes associated with poor maintenance practices, in which a potentially faulty unit is returned to operation where it might result in a safety hazard [97,246]. In the automotive industry, such considerations have led to claims [148] that every complaint or return of a product in a safety or emissions regulated product should be viewed as a field failure.

Although loose connections [97,206] and corroded or dirty connector contacts [206] are associated with intermittent faults, there is little correlation between number of connectors on a board and the number of NFF events [135,246]. Hardware redundancy is often used to increase reliability, but such system designs increase system complexity and therefore risks of incorrect system design, which can increase incidence of false alarms and result in NFF events [246].

Interactions with other units can cause secondary or cascade faults, in which a fault in one unit can cause others to exhibit fault behaviours and result in later NFF occurrences with those affected (non-faulty) units [246]. The risk of such secondary faults increases in highly integrated systems [246]. When a system is used in a wide range of unpredictable operational conditions, unit returns without any clear description of an observed failure are common [168] and efforts to duplicate conditions in a support environment can yield significant NFF rates in properly functioning units [246]. Conversely, such fault reports can also, naturally, occur in units which really are faulty, but shop testing is unable to discover the fault [97,167].

Causes of NFF concerns during the utilisation and support stages include inefficient diagnostic methodologies, incorrect fault localisation procedures, inadequate diagnostic tools, and inefficient service strategies or policies [97,167,246,277]. A common approach within the aerospace, automotive, and trucking industries that leads to NFF concerns is module swapping or part-changing, sometimes called the "shotgun" approach, in which technicians replace several LRUs to ensure that the right one is replaced [64,148,246]. Such approaches quickly return a system to an operational state, but introduce the cost of handling multiple removed LRUs that are not faulty and therefore subsequently introduce NFF occurrences [64,246]. NFF events can also occur due to compatibility, reliability, calibration, and health concerns with test stations, and due to variations of configuration between identical test stations [246].

High system complexity is considered a major cause of NFF events [148,149,167,246], probably due to the relationship between system complexity and system reliability and maintainability that are established at design time [246]. Some analyses [135] have, however, shown little relationship between the number of complex components on a circuit board and number of NFF events, but it needs to be noted that an analysis at different indenture levels may contribute to different conclusions [246]. Another cause of NFF events is ineffective or ambiguous test

requirements stemming from lack of any distinction between physical faults and functional anomalies by which those faults might be recognised or localised [246].

The NFF phenomenon is considered difficult to solve, because it is inherently complex and the same symptom has multiple potential causes. These causes may be grouped broadly into three categories of factors: intermittent failures, diagnostic methods, and service strategies [149]. Current guidelines for management of NFF phenomena advocate a complete system methodology that crosses all domains including design and production, flight operations, line operations, and shop operations [277].

6.9 Subverted hardware

With the global trend towards outsourcing manufacture of hardware components, electronic systems are increasingly vulnerable to maliciously modified hardware components [292,299,322,332,344]. Because of these trends, the cost of ensuring the entire fabrication process (the trusted foundry) is trustworthy is increasingly prohibitive [349].

Increasing hardware complexity increases the investment needed to deliberately subvert a hardware design [292]. However, this hardware complexity also increase the difficulty of detecting subverted hardware [292] and therefore of achieving an appropriate level of system assurance [299]. Encapsulation of modern integrated circuits (coating within layers of resin) protect the circuit from natural damage and from tampering, and protect the intellectual property invested in the design, but also increase the difficulty of detecting subverted hardware, often driving a need for destructive techniques [292].

Trojans may be implemented as hardware modifications to ASICs, COTS parts, microprocessors, microcontrollers, network processors, digital signal processors (DSPs), or as firmware modifications [349].

Hardware trojans may be characterised as functional or parametric [292]. A functional trojan is implemented by introducing or removing transistors or gates, in order to systemically change the function of the circuit [292]. Such changes may include redirection of information to alternate storage channels or subjecting information to some transformation [292]. A parametric trojan modifies structure, physical specification, or arrangement of a circuit in order to affect its operating parameters [292]. Recent examples of parametric trojans introduce hidden reliability defects that accelerate time-based wearing mechanisms such as HCI (§5.2.3), NBTI (§5.2.5), or TDDB (§5.2.4) and/or condition-based triggers such as electrostatic discharge (§5.4.1), latchup (§5.4.4), or soft errors (§5.2.6) [321,322].

Hardware trojans also vary in size [292]. Small trojans may modify, add, or delete only a few circuit components, while a large trojan may include many such changes [292]. Smaller trojans are more likely to be activated than large ones [292].

Other characteristics of hardware trojans include their distribution across a circuit (localised or not), activation methods, and their intended effect of payload [292].

The main concerns associated with hardware trojans, particularly for high-assurance systems, are difficulty with their detection or removal. Standard testing methods are ineffective because

structural pattern testing generally does not cover unanticipated behaviours, and routine functional testing is unlikely to reveal harmful functions unless the presence and nature of the trojan is known [332]. Trojans will typically be designed to only activate under very specific conditions, which reduces likelihood of being uncovered by random or functional stimuli [349]. Exhaustive input pattern testing is prohibitive with complex circuits [258,332] as is negative functional verification (exhaustive testing to prove that a chip contains no extra functions) [292].

Destructive testing of a chip is typically very expensive (analysis of a single chip can take months), becomes more expensive with transistor density (i.e. miniaturisation), and results cannot be extrapolated across a manufacturing batch as an adversary may compromise only a small population within a manufactured batch [258,332].

For reasons such as these, most current techniques for detecting the presence of a hardware trojan rely on the existence of a golden gate-level netlist (i.e. a trusted specification or model of the circuit design) [344]. However, in practice, a golden model may not exist [313,344] due to either technology or commercial concerns.

7. Obsolescence of electronic systems

In the normal course of product development, the design of products and systems change in a manner consistent with shifts in demand and with changes in availability of materials and components from which they are manufactured [215]. For most high-volume, consumer oriented products and systems rapid rate of technology change translates into a need to stay on the leading edge of technology in order to prevent loss of market share to competitors [215]. However, some product sectors such as aerospace, ships, industrial equipment, and medical equipment lag behind the leading edge because of high cost or long times associated with technology insertion or refresh [215]. Several of these product sectors are "sustainment-dominated", in the sense that long-term lifecycle costs exceed procurement costs of the system [215]. In such applications, limited lifetime of electronic parts results in printed circuit boards being redesigned for no other reason than to substitute parts that are no longer available [241]. This often requires modification of software, with the need for additional system testing [241].

Obsolescence is a primary risk driver for the Low Volume Complex Electronic Systems (LVCES) industry, because most such systems are intended for use over an extended time, so they are vulnerable to obsolescence of parts, subsystems, and technologies [90]. Reliability is difficult to achieve in LVCES because of application environments being harsher than those for which system components are designed [90]. This introduces maintainability and supportability concerns, particularly when the system life cycle significantly exceeds design lives of components or sub-systems [90]. Suppliers of systems with long system life cycles are in a difficult position, because it is almost inevitable that manufacture of any semiconductor device will be discontinued, typically with six to twelve months notice, and the system suppliers then no longer have a reliable source of components to meet ongoing production, maintenance, and repair requirements [350].

In contrast, obsolescence is not generally a risk for Volume-Driven Complex Electronic Products, where the main concerns include affordability and profitability, development cycle time, functionality and performance, manufacturability and quality [90].

Embedded systems, as used in military and aerospace industry, are typically LVCES used for control, monitoring, or communication that would be in use for more than ten years [241]. However, the high volume consumer market relies on getting new devices quickly to market, resulting in closing down low-volume production of older parts [241]. Problems of obsolescence of such systems can therefore be expected to exist for as long as Moore's Law (§4.1) remains in effect [241].

A survey of Norwegian electronic companies indicated that that 78% of companies would need to make a partial redesign within six months to include new parts and that 15% of suppliers would need to make a complete redesign within three years [241].

Within the current Australian Defence Capability [353] the main stated purpose of some projects is addressing obsolescence or performing some form of technology refresh. There are also several projects that are directly or indirectly considering concerns such as addressing obsolescence, technology refresh, future sustainment, improving sustainability, or update/modernisation of systems or telecommunications equipment.

7.1 Planned obsolescence

Value engineering (§4.5) supports or enables the economic process of planned obsolescence [29,59,75], also known as built-in obsolescence. This is the process, first introduced in the 1920s and 1930s with the advent of mass production, of a product becoming obsolete and/or non-functional after a certain period or amount of use in a manner planned or designed by the manufacturer. Planned obsolescence offers benefits for a producer because the product fails, or its utility reduces, and the customer is under pressure to purchase again [2]. Planned obsolescence also hides the real cost per use from the customer, allowing a supplier to charge a higher price than the customer would otherwise be willing to pay, or would be unwilling to spend all at once [2].

Planned obsolescence stimulates demand by encouraging purchasers to buy again sooner if they still need a functioning product [29]. Estimates of planned obsolescence can influence a company's decisions about product engineering: the company can use the least expensive components that satisfy product lifetime projections or goals [29,59]. Planned obsolescence may also be implemented by making the cost of repairs comparable to replacement cost, or by simply withdrawing service, maintenance, or parts needed to sustain an old product [29]. Creating new product lines that do not interoperate with older products can also make an older model quickly obsolete, forcing replacement [29].

Planned obsolescence tends to benefit a producer with at least an oligopoly [187]: before introducing a planned obsolescence the producer has to know the customer is sufficiently likely to buy a replacement from them. This is often facilitated by information asymmetry between the producer, who knows how long the product was designed to last, and the customer, who does not [187]. The practice of planned obsolescence is often difficult to pinpoint by customers, as it is

complicated or obscured by related concerns, such as presence of competing technologies or feature creep which expands functionality in newer product versions [29].

Social criticism [2] identified two categories of planned obsolescence, called "obsolescence of desirability" and "obsolescence of function". "Obsolescence of desirability", also described as "psychological obsolescence", "an illusion of change", or "styling" [2], refers to marketers' efforts to wear out a product in the owner's mind.

If marketers expect a product to become obsolete then it can be designed to last for a specific lifetime [29,59,75], through application of value engineering to reduce the cost of making the product and therefore lower the price [252]. Products could be built with higher-grade components but they are not because, it is argued, this imposes an unnecessary cost on the purchaser so a company will typically use the least expensive components that satisfy lifetime projections [187]. In practice, the commercial electronics market focuses on satisfying warranty period, and ensuring a product is obsolete before wear-out effects become dominant [150]. It is considered [38] that obsolescence occurs due to the equipment design of the manufacturer, or worn-out equipment, and that it is designed into a product to encourage sales sooner than the customer would expect. This means that obsolescence may be either intentional or the result of poor planning [38]. The relationship of obsolescence to company business plans is exhibited by a quote from Bill Gates "*The only big companies that succeed will be those that obsolete their own products before someone else does*" [171,266].

Planned obsolescence remains an active consideration in industry and economics research [e.g. 254,281,326,354].

7.2 Views of obsolescence

Obsolescence as a process

Obsolescence is a process in which value or utility of a product reduces over time due to introduction of new products or changes in demand. It affects all manufactured products to some degree. Component obsolescence has always been a fact of life, but occurred infrequently enough that equipment suppliers managed it on an ad hoc basis [134]. However, the frequency of concerns has increased due to reducing component lives, particularly of semi-conductor components, causing challenges for suppliers of long-life mission or safety related systems, due to a shift of emphasis within the electronics sector to high volume, highly integrated, low cost, mass-manufactured components. Avionics and military systems may encounter obsolescence before being fielded, and always experience obsolescence concerns during their field life [266].

Obsolescence as a state or condition

Obsolescence is also viewed as the state when a part is no longer available from the original manufacturing source [178]. This may occur due to the manufacturer no longer being in business or having sufficient commercial incentive to continue supply. Part obsolescence may also occur because of non-availability of a base material.

Obsolescence as DMSMS

Obsolescence may also be described as Diminishing Manufacturing Sources and Material Shortages (DMSMS), defined as the loss or impending loss of manufacturers or suppliers of critical items and raw materials due to discontinuance of production [119,286,340].

In practice, the government customer perspective on DMSMS management is usually "How do I protect myself?", with initial acquisition cost and total operating cost often significant considerations [340]. In contrast, the supplier perspective on DMSMS reflects a dichotomy of "How do I do the right thing and maintain a competitive edge?", as one aspect requires adding overhead and the other requires reducing overhead [340]. Suppliers are rarely concerned with total operating costs, as they traditionally do not deal with long-term storage and warehousing costs associated with post-deployment sustainment [340].

7.3 Impacts of obsolescence

Obsolescence affects all equipment through its intended or extended life cycle and also affects software support and test equipment, standards, processes and other logistical products [286].

If a component becomes obsolete, it may lead to obsolescence of the next-higher assembly or application [207]. Regardless of the cause of component obsolescence, the magnitude of any impact depends upon the application of the equipment in which it is used and the manner in which that equipment configuration is managed. While the design and service period for modern commercial electronic systems rarely exceed five years, the design, production, and service periods for aircraft can vary in the range of 20 to 40 years [286], in some cases exceeding 50 years [129]. In cases where a solution is found to treat obsolescence of a particular aircraft system component, the usage of the planned modification typically requires extensive regulatory and therefore technical justification for qualification or certification [215].

Suppliers who assemble components that they acquire from manufacturers into assemblies also experience the effects of component obsolescence [129]. Systems integrators generally seek to maintain technology continuity through ATD (Advanced Technology Development), EMD (Engineering Manufacturing Design), LRIP (Low Rate Initial Production), and Production phases [130]. COTS products have contributed to reduced length of these development cycles, particularly for non-mission critical or benign environment programs where the jump has been made from ATD directly to production and deployment but typical life of individual components is sometimes insufficient to support two phases of program development [130].

Papers studying optimisation of preventative or corrective maintenance rarely consider obsolescence and assume that failed or used pieces of equipment are replaced by identical equipment [186,317]. In reality, new equipments are often readily available in the market to achieve the same missions but with desirable attributes (e.g. lower failure rates, lower energy consumption, lower cost, etc) and it is increasingly difficult or costly to find old-generation spares [186,317].

Another common study assumption is that new components are compatible with the system in which they are to be installed [186]. However, even if newer components display higher performance, their inclusion in a system could weaken it, particularly at the beginning because

of changes of installation procedure or unexpected adaptations of the system [186]. The likelihood of this may be expected to decrease with time and experience, due to vendors providing better technical information and technical staff getting used to the new technology [186].

7.4 Types of obsolescence

Commercial hardware and software vendors have developed a symbiotic supply chain relationship that is not readily influenced by the military or aerospace sectors [266]. Hardware improvements cause software suppliers to produce new software and thereby make older software versions obsolete. New software, in turn, renders older hardware obsolete. Functioning of long service life systems therefore increasingly rely on technologies that have become inaccessible.

Functional obsolescence

Functional obsolescence refers to impairment of usefulness of a device or equipment due to a design defect, or due to its inability to be upgraded or modified to meet new functional requirements. Products which naturally wear out or break down may become obsolete if replacement parts are no longer available, or when the cost of repairs or replacement parts exceeds the cost of a new item.

Technological obsolescence

Technological obsolescence results from evolution of technology and associated business decisions: as newer technologies appear, older ones cease to be used. New components may support different interfaces than the original, so changing one component makes it necessary to change others. Strategies for dealing with technological obsolescence include; migration of digital information to technologies from which it is accessible; hardware or software emulation of obsolete systems, and; preservation of obsolete technologies through maintenance or remanufacturing activities.

Logistical obsolescence

Logistical obsolescence is the result when supply or support arrangements terminate because of business decisions by suppliers such as; ceasing to produce or sell the product (end of life); refusing to expand or renew licensing agreements (legally unprocurable); or terminating maintenance agreements.

Addressing logistical obsolescence is less technically challenging than addressing functional or technological obsolescence [212], as it may be resolved by simple means such as license downgrades of software and replication. However, it is commercially more challenging in circumstances of a constrained licensing agreement on a legacy system, if an inflexible vendor refuses to supply sufficient licenses to allow continuing use.

7.5 Software obsolescence

COTS software supportability characteristics differ somewhat from those of COTS hardware, although the progressive obsolescence of both hardware and software can affect system sustainment [166].

Little attention has been paid in literature to software obsolescence, other than to the topics of "information or digital preservation" or to termination of sales and support [266]. Most organisations employ only a reactive approach to manage software obsolescence, such as factoring in unspecified additional integration efforts or vendor communication [212].

The definition of software obsolescence depends both on the system that uses the software and on how that system is used [266]. The "end of support" date is a significant criterion for many commercial applications and operating systems as that is when security patches cease so the software becomes a security risk [266]. For embedded or isolated applications, software obsolescence is often determined by either inability to obtain necessary licenses or functional changes to the system in which the software is embedded [266].

Few strategies exist for managing software obsolescence [212], and few military programs track and mitigate software obsolescence as a distinct risk. Options available to manage software obsolescence include [212] upgrading of software licenses to recent versions, downgrading to older versions, employing open-source software products, employing standard Application Programming Interfaces (APIs) or software wrappers, using middleware to maintain boundaries between disparate commercial platforms, and performing regular market analysis to maintain market awareness. Few mainstream approaches used for mitigating hardware obsolescence risks are applicable, or even meaningful, for managing software obsolescence [266].

The FAA [166] considers the projected End-of-Service (EOS) date, at which the vendor no longer provides support, to be the primary point in time at which impacts and treatment options for software obsolescence should be evaluated. Although there is some variation, COTS software vendors typically provide support for two previous software generations before declaring EOS [166]. Vendor support may take the form of technical support to integrate the product during development and provision of updates to incorporate fixes [166]. Technical support may be available after EOS on an hourly basis [166]. Factors to be considered during continuous evaluation of COTS software products include [166];

- Reducing software support skills (integrator, third-party, or integrator) over time;
- New software product compatibility with underlying hardware platform;
- Complexity of COTS software interfaces (e.g. operating system) with other software products, applications, middleware, glue code, and custom/legacy interfaces;
- The ability to modify a system function without unknowingly exceeding a software product tolerance;
- Potential for introduction of "unknown unknowns" with untested products (unused code, timing differences, firmware changes, etc);
- Sole source dependency for critical software components and data rights availability;

- Information security;
- Licensing options and costs.

7.6 Ageing and legacy aircraft systems

Aircraft have a long operational life, typically exceeding 20 years [136,138] and often greater than original planned life [136]. Contributors to this, for military aircraft, include a reduced threat, trends to high total cost for new weapon systems, resulting in reluctance to undertake new developments to replace ageing aircraft [123]. The obsolescence problem for avionics system integrators may be characterised [334] as coping with a component life of 7 years compared with aircraft left exceeding 30 years, whilst maintaining high-capability and lost-cost upgrades. A range of generic problems arise with maintaining and repair of aged and "legacy" aircraft, including [342];

- They may have been designed and built to standards that are no longer acceptable;
- They may not have been designed with ease of aircraft access and maintenance in mind;
- There may be a diminishing pool of engineers with requisite "old-fashioned" engineering skills;
- Maintaining relevant corporate knowledge and records becomes more difficult as experienced design and maintenance personnel at the Design Authority and in Depth and Front Line maintenance retire;
- The availability of spare parts becomes more difficult as the number and interest of manufacturers dwindles towards end of service life, and "robbing" spares from other aircraft become less easy;
- Adding modifications and integrating new systems with old can become more difficult as the aircraft ages;
- Different systems and components age at different rates;
- Determining a "baseline" of safety for such aircraft becomes more difficult as the aircraft ages.

Particular problems with ageing avionics include [136] identifying the systems that are cost drivers in order to prioritise attention, determining requirements for replacements, identifying alternative technologies that affordably satisfy the requirements, and determining then obtaining needed funding. Addressing these problems introduces both management and technical challenges, including the need to select an upgradeable architecture as parts may become obsolete before completion of production [136]. In practice, demands for net-centric capability mean that older aircraft systems require upgrade [209].

In practice, few avionics systems can be sustained for ten or twenty years without difficulty [138].

7.7 Addressing obsolescence

It is considered [315] that, to realise benefits of using COTS, it is necessary to effectively manage and plan component substitution strategies before components become obsolete in order to keep sustainment costs within projected budget and to provide continued system availability. A successful obsolescence management strategy requires a balanced judgement to be made between probability of obsolescence occurring on a platform and the expected impact, including the long term effects. This may often imply a mix of strategies.

A reactive obsolescence management approach directs that action be taken *only* once obsolescence affects supply of an item [315]. No planning is undertaken to predict or mitigate occurrences [315]. Funding and management activities are only focused on rectification once an issue is observed and will cause an operational impact. A reactive strategy generally exposes the support agency to significant levels of financial or supportability risks, in comparison with proactive or adaptive strategies. Such a strategy is generally acceptable when the risk affecting operational support or availability of the delivered capability is very low, or more active approaches are considered inappropriate or not cost effective [315].

A proactive strategy accepts there will be an inevitable impact of obsolescence, and aims for timely implementation of options to ensure continuing supportability and availability of equipment. Such a strategy requires early commitment and funding to resolve obsolescence concerns in advance, with a mindset of "spend to save". A proactive strategy aims to take initiative through various methods, including monitoring component sources and availability, identifying and purchasing alternatives, introducing Open System Architectures to facilitate easier insertion of technology updates as threats and capability requirements evolve. A proactive strategy incurs cost in order to mitigate significant future financial, supportability, and operational uncertainties and risks.

Adaptive strategies include both reactive and proactive strategy elements, based on a premise that no obsolescence management strategy can be completely proactive and some reactive management is needed. An adaptive strategy is partially passive, involving a "watching brief" to monitor items for possible obsolescence and then initiating groundwork to identify and prepare appropriate responses. If potential obsolescence impacts are identified early, an adaptive strategy can reduce financial uncertainty associated with long-term support and, if the predicted obsolescence impact is significant, can initiate timely and cost-effective mitigation of supportability and operational risks.

Traditionally, most responses to obsolescence have been reactive [286] and bottom-up [129], but some recent legacy programs have started to implement DMSMS risk management strategies [286]. There are several methodologies, databases, and tools that address status, forecasting, risk, mitigation, and management of electronic part obsolescence [240]. Most methodologies (whether reactive, proactive, or strategic) focus exclusively on hardware life cycle cost despite software life cycle costs potentially contributing more to the total life cycle cost in complex systems [266]. Key elements of existing methodologies include [179] part life cycle characterisation, part obsolescence forecasting, product deletion, and life cycle planning.

7.7.1 Government approaches

The Engineering Change Proposal (ECP) process has traditionally been one of the primary tools used in engineering support contracts to mitigate obsolescence. Program managers use ECPs to respond to product safety, manufacturing concerns, and reliability concerns of products in service. The ECP process, however, is sometimes slow and not specifically concerned with managing obsolescence. While OEMs can recommend improvements, the ECP process shifts responsibility for identifying and mitigating any risk of obsolescence to the government. The principle strategy supported by the ECP is product redesign.

It is considered that traditional government approaches, of buying parts to address failures and intensively managing supplies, contribute to concerns such as a large outdated infrastructure, ageing fleets, failing reliability, increasing obsolescence, and rising ownership costs [214]. Usage of COTS is not considered to mitigate obsolescence concerns, as COTS equipment itself is subject to obsolescence [129]. There is a general lack of standard procedures in the defence industry for cost estimation of obsolescence, with the rough estimates made at early project stages often being found to be an inadequate basis for contract negotiation [311].

Lifetime buys are often used to guard against obsolescence but, with components often obsolete before production is complete, this approach is complicated by the need for a supplier to estimate requirements for full-scale production and lifetime support very early in the product life cycle, requiring preplanning for a minimum lifespan, a minimum production volume, and the introduction of a replacement product as early as possible [130]. In 2000, five years was considered a reasonable time span for a product to transition from design and development through to maturity [130]. The average life cycle of a typical semiconductor device (including introduction, design-in, production, low-volume, and end-of-life phases) is currently about three years [350].

Business practices in the UK Defence sector have been evolving from traditional acquisition approaches and include a range of initiatives including spares inclusive, availability based contracting and, ultimately, contracting for capability [311]. This gives challenges with costing availability, particularly with costing of obsolescence, rather than the traditional costing of a solution [311]. In traditional approaches, the customer has been responsible for the cost of resolving obsolescence with the contractor responsible for managing it, with suppliers having no incentive to resolve obsolescence in a cost effective manner, so alternate contracting strategies are emerging that distribute responsibility in different manners between suppliers and customers [311].

7.7.2 Maintenance policies and models

There is a considerable body of work concerned with maintenance models for the control and surveillance of systems subject to deterioration in service [147]. A survey of scheduling policies [5] and a subsequent survey of models [232], concerned with optimising decisions to procure, inspect, and repair/replace a unit upon its deterioration in service, identify and review the majority of this work. Available models support many maintenance policies [147], including;

- age replacement policy, in which a unit is replaced when it reaches a specified fixed age, or upon failure;

- random age replacement policy, in which a unit is replaced when it reaches a specified age, but time is represented as a random variable due to impracticalities of strictly periodic maintenance actions;
- periodic preventative maintenance policy, in which a unit is repaired upon reaching some specified age or upon failure;
- failure limit policy, in which preventative maintenance is only performed if failure rate (or some other reliability measure) exceeds some threshold, and any intervening failures are repaired;
- sequential preventative replacement policy, in which units are replaced at pre-selected times that are not necessarily at equal time intervals;
- repair cost limit policy, in which repair costs are estimated and repairs only performed if they are less expensive than some pre-determined limit;
- repair time limit policy, in which a repair is attempted but if some pre-determined limit is exceeded then the unit is replaced;
- repair number counting policy, in which failed units are subjected to minimal repairs and replaced once they have failed a specified number of times;
- preparedness maintenance policy, in which number of failures are only detected by specific inspection, after which a decision may be made to repair;
- group maintenance policies, in which a complete population of units is replaced or, alternatively, inspected after a single failure occurs;
- opportunistic maintenance policy, in which working units are replaced or repaired at a reduced additional cost while addressing failure of another unit.

There is relatively little work concerned with proactive life cycle planning [179], despite such techniques being considered necessary to contain sustainment costs within projected budget while ensuring availability of a system during its evolution [315].

7.7.3 Total Product Life Cycle Management

Total Product Lifecycle Management, which starts with inception of a new idea and doesn't end until the last customer has withdrawn a product, is suggested as being necessary to bridge the widening gap between customer and end-user needs and industry's ability to deliver effective and maintainable solutions [130]. An ideal solution would be to deal only with suppliers that make a reasonable promise of longevity, but this is not always practical with leading-edge technologies that evolve rapidly [130].

Performance Based Logistics is further suggested as a contracting methodology that offers an integrated approach to modernise systems and address system obsolescence [214].

DMSMS management by US Department of Defense

US Department of Defense [340] considers DMSMS (Diminishing Manufacturing Sources and Material Shortages) to be a risk to the life-cycle support and operational availability of weapon systems, and states [340] that effective DMSMS management requires proactive solution of

obsolescence problems before they adversely affect system availability or total ownership cost. Managing DMSMS risks follows a standard sequence [340];

- *Identify*. Identify "problem" parts in line replaceable units (LRUs) that are, or will be in foreseeable future, obsolete.
- *Assess*. Considering the population of problem parts, determine and prioritise the LRUs most at risk for current and future DMSMS impacts;
- *Analyze*. Research problems parts in high-priority LRUs and, for each LRU, develop an optimum set of DMSMS solutions;
- *Implement*. Budget, fund, contract for, schedule, and execute the solutions for the high-priority LRUs and then for lower-priority LRUs.

Recognising that DMSMS management practice cannot be the same for every weapon system, the US DoD [340] defines four DMSMS levels of intensity.

Level 1 involves largely reactive approaches sufficient to resolve known obsolescence problems. Level 1 processes generally include establishing a DMSMS management team (DMT), basic training for that team, development of a formal DMSMS management plan, and implementing solutions to near-term obsolescence concerns. For new acquisitions, DMSMS tasking and data by-products are specifically considered in development, production, or support contracts.

Level 2 involves more proactive approaches sufficient to mitigate risks of future obsolete items. Level 2 processes include all Level 1 approaches plus predictive analysis, establishing and using a DMSMS solution database, budgeting for future obsolescence solutions, and defining a method to prioritise LRUs/WRAs for DMSMS risk.

Level 3 involves proactive practices sufficient to mitigate risk of obsolescence when there is a high-probability opportunity to enhance supportability or reduce total operating costs. Level 3 processes include all Level 2 approaches plus active maintenance of life-cycle costing and cost-avoidance estimates, advanced training of the DMT, communication of the impact of any funding shortfall to decision makers, inclusion of DMSMS tasking and considerations of data requirements included in applicable contracts concerning legacy systems and an ongoing technology assessment and insertion program.

Level 4 implement proactive approaches during conceptual design of a new system and continued through its production and fielding. Level 4 processes include all Level 3 processes plus use of technology road-mapping, planning of system upgrades, attainment of technology transparency, and achieving accessibility for alternate source development of key components. Higher levels of intensity are generally preferred for new developments, systems with long remaining lives, and for systems with significant or chronic DMSMS concerns [340].

7.7.4 Open System approaches

A number of integrators are adopting an approach based on Open Architecture [189,242,263], Functional Partitioning, and Technology Insertion. Technology Refresh is a derivative of Technology Insertion, in which each new technology step is made 100% backward compatible

with the previous, so that old technology may be refreshed by swapping out old for new whenever maintenance actions allow [130].

Reconfigurability in system design and process is a key approach for coping with obsolescence and some low-volume manufacturers and system integrators are investing in modularised technologies to reduce burdens of system redesign and qualification [90].

Within the electronics industry, lack of standards is affecting progress with implementing some technologies and growing markets, and there is a need for agreement on a mechanism to provide an open architecture for best-in-class test integration [302].

The Generic Open Architecture (GOA) is the subject a Society of Automotive Engineers standard, SAE AS4893, developed as a framework for discussing open systems architecture and for identifying critical systems and interfaces [81]. It is considered useful as a reference model for weapons systems, being comparable to the US DoD Technical Reference Model cited in the Joint Technical Architecture, with main benefits being related to software portability and systems interoperability [113].

The Automotive industry has produced a number of releases of AUTOSAR (AUTomotive Open System Architecture) since 2003 [341]. The objective of AUTOSAR is to establish an open industry standard for automotive software architecture between suppliers and manufacturers, with a principle aim of mastering the growing complexity of automotive electronic architectures [341]. The standard specifies a set of software architecture components and defines their interfaces [341]. Key objectives are defining a common understanding of how electronic control units (ECU) cooperate and separating software from hardware in order to allow software reuse and enable evolutions with limited redevelopment and validation [341].

USAF have endorsed a modular open-system approach (MOSA) as a way of developing scalable, more easily upgradeable avionics systems and reducing total ownership costs in both legacy and new aircraft [127]. It was also found [127] that a comprehensive MOSA solution to aging avionics problems could save money in the long term, but would generally cost more than customised point solutions in the short term, particularly for avionics upgrade in the legacy (in-service) fleet. Key requirements of a system engineering process to apply open system standards to (US DoD) Avionics are that the process is requirements driven, makes use of a catalogue of preferred interface standards, and uses iteration as a key approach to address affordability and mission needs [310]. Designers of avionics equipment for U.S. Navy aircraft consider obsolescence as their biggest obstacle in meeting demand for upgrades and retrofits of existing aircraft, and designing each system with an open architecture as a key means of overcoming this obstacle [348]. MOSA is considered to challenge military procurement models in several ways [127];

- Theoretically, supplier competition may be solicited at various architectural levels (components, circuit-board, module, or subsystem), but it is necessary to provide incentives for qualified suppliers to take advantage of openness and to invest in improvements and innovation of avionics systems;
- The traditional mindset of acquiring software and hardware will need to be changed to one of acquiring functionality;

- Protection and value pricing of a supplier's intellectual property will be a key to success and will require workable business models.

Currently, there is limited uptake of open system approaches and standards in the Integrated Modular Avionics (IMA) industry [262]. Factors in this include the significant likelihood that IMA systems must be specifically designed or modified to fit a unique aircraft installation [296] and the industry-wide impacts that open standards would have on all IMA industry stakeholders [262].

Aviation systems have traditionally used a federated architecture, in which many distinct computer systems are assigned to distinct control functions in the aircraft, and communicate with each other only using directed or broadcast data buses [157]. The systems are largely decoupled and only communicate as needed to perform their designated functions [157]. This provides inherent fault-containment and isolation, as faults cannot easily propagate from functions that are located in separate physical units and, at system level, a federated system that provides a limited set of functions can often be more easily verified and validated than a complex, highly integrated system with many functions [157]. However, federated system approaches have disadvantages in terms of number of systems and components which can increase costs of production, certification, and maintenance while producing obstacles for improvements in functional or safety procedures if these affect several subsystems [157].

Integrated Modular Avionics (IMA), in contrast to a federated architecture, incorporates multiple functions, possibly at different levels of criticality, on a single physical platform [157]. From a certification perspective, IMA systems are not considered fundamentally different from more traditional federated designs, but increasing system complexity means it is becoming increasingly difficult to ensure coverage of all aspects of an IMA system during approval for use on an aircraft as no single entity "owns" or completely understands the various functions contained in an IMA system [296]. When executing multiple functions on the same computer hardware it is necessary to protect functions from adversely affecting each other [157]. This is often achieved through partitioning, which controls any additional hazards or failure conditions that might be introduced when multiple functions share computer processors, memory, input, output, and other system resources [157].

Real-time operating systems (RTOS) have become central computing resources that provide protection schemes in both space (memory) and time (CPU throughput) domains for both IMA and non-IMA systems [157]. The challenge in RTOS design is the design of a partitioning solution that enables exchange of information between partitioned functions and shared resources while keeping the partitioned functions largely autonomous and unaffected by other functions [157].

7.7.5 FAA guidance

The US Federal Aviation Administration (FAA) [166] considers COTS obsolescence one of the more difficult aspects of COTS risk management [166]. This is rooted in the rapid evolution of COTS products and product obsolescence is the fundamental problem because new versions or releases of COTS products are brought to market frequently and the level of maintenance support and availability of spares for a given version or release diminishes over time and can rapidly become more costly [166].

Key trigger points are associated with a product transitioning between obsolescence phases, and represent points at which the FAA considers impact of changes on a system or program should be re-evaluated [166];

- End-of-Life (EOL) occurs when the OEM ceases manufacture of the product. OEMs are typically willing and able to provide repair/replacement services;
- End-of-Service (EOS) occurs when the OEM no longer services the product but third-party sources are available to provide repair/replacement services. If no such source is cost-effectively available, the product reaches the EOR trigger;
- End-of-Repair (EOR) occurs when hardware product support is unavailable by any means or is cost-prohibitive. After EOR, the system usage or demand depletes remaining depot spares over time, increasing uncertainty (therefore supportability risk) for the program about remaining spares quantities and item failure rates;
- End-of-Maintenance (EOM) occurs when a site requisition cannot be replenished. After EOM, depot stores and spares quantities are depleted, leading to service degradation (loss of redundancy) and eventual loss of system operations.

Management of COTS product obsolescence entails initial use of a system-level strategy, which must be formulated early in a COTS-based system's acquisition cycle, and subsequent use of product level support options [166]. The system-level strategy integrates activities such as pre-planned product improvements (P^3Is) and new requirements changes with projected obsolescence induced system upgrades, and provides the basis for budget projections and risk management [166]. Because of the extent of variability encountered when using COTS products, the system-level strategy must be periodically reviewed and adjusted as needed [166].

During early product planning, a notional architecture is used to begin a high level cost estimating process [166]. As the system architecture is defined and the COTS product composition becomes known, system-level assumptions and resultant planning can be refined to reflect EOL and EOS data gathered through market research activities and used to evaluate support options when a manufacturer projects an EOL or EOS date, [166]. The impact to a system or program can range from none to major redesign, depending on vendor notification lead time, failure rate, spares availability, alternate product compatibility, interface dependencies, new requirements, technology trends, costs, and risks [166]. Customers typically have six to twelve months, once an EOL announcement is made, to decide whether to place a last-time buy or find an alternative solution [350]. Support options can include [166];

- No action required, appropriate when product reliability or availability of replacements allows continued product support, regardless of obsolescence phase;
- Lifetime Buy, involving acquisition (purchase, cannibalisation, trade) of replacement products, components, or items sufficient to meet a projected failure or demand rate until a defined point in time;
- Extended Maintenance or Warranty, involving purchase of technical or repair support from the OEM to extend product support beyond the original timeframe;
- Third Party Maintenance, involving establishment of technical and/or repair support by a qualified vendor other than the OEM;

- Technology Refresh, involving periodic replacement of COTS products (e.g. processors, displays, operating systems, commercial software) with equivalents within a larger system to assure continued system supportability. Technology refresh does not change the system performance baseline. Refresh period is based on when the product can no longer be supported;
- Redesign or Integrated Change, involving addressing obsolescence through a system redesign (new products, rearchitecture) or integrating the replacement of obsolete products within a larger system upgrade or pre-planned product improvement (P^3I);
- Purchase of data rights, in which a product user makes an arrangement with the OEM to secure proprietary data rights to enable organic or third-party product support, and;
- Reclamation or salvage, also referred to as cannibalisation, is typically a last-resort option to reclaim discarded product and reassemble to create a functional product.

FAA strategies for managing and mitigating COTS risks include [166];

- Employ COTS-knowledgeable individuals in all of the analytical processes [166]. Required knowledge includes an understanding of the risks and mitigations strategies unique to COTS products and an understanding of COTS product obsolescence stages and how to limit their effects on potential system performance;
- Involve users early and throughout the program life cycle to identify and resolve COTS-related issues;
- Perform continuous COTS product market research of technology trends, product applicability, and obsolescence status;
- Integrate market research results with field data and new requirements;
- Develop and maintain flexible performance requirements suited to the use of COTS products;
- Institute and maintain an ongoing COTS product testing capability;
- Develop and maintain non-technical COTS selection factors;
- Use COTS-sensitive analytical and budget processes;
- Integrate COTS-based technology evolution planning with the overall Integrated Program Plan;
- Emphasise strong and COTS-relevant configuration management processes;
- Use a COTS-experienced systems integration agent;
- Leverage commercial infrastructure wherever feasible;
- Avoid modification of COTS products when possible.

7.7.6 Australian Defence policy on obsolescence management

Australian Defence [193] describes obsolescence as an increasingly difficult problem as life cycles of components are decreasing while life cycles of mission systems are increasing, and describes obsolescence management as a risk management process which addresses; risk of

obsolescence and its most likely nature and point of occurrence throughout the life cycle; consequences of those problems in terms of cost (particularly Life Cycle Cost), availability, performance, supportability, and safety; and options for treating the obsolescence problems, including the costs, benefits, and risks with the treatment options.

Specifically, the Defence policy [193] requires that obsolescence management activities ensure that; systems and equipment will be fully supportable at a minimised life cycle cost; in-service planning during requirements and acquisition addresses obsolescence risks over the programmed life of type; in-service management of both new and legacy systems and equipment addresses obsolescence risks over the life of type, to ensure systems and equipment meet performance and preparedness requirements at a minimise life-cycle cost; and disposal processes that involve reuse within Defence consider the obsolescence problems and risks.

8. Discussion and Conclusions

Defence acquisition programs traditionally relied on assurances associated with Military Standards to more easily meet requirements for extended operating range, quality and reliability, and could achieve economic advantages of reduced parts inventories and better quantity discounts if military grade components were used across multiple programs. With a shift towards use of commercial products, specifications, and standards, the military has less control over reliability and obsolescence characteristics of its equipment. This is particularly evident with electronic systems, which rarely have design and service periods that exceed five years, but are used in military capabilities with service periods of several decades. Manufacturers routinely employ "Physics of Failure" techniques to ensure there are no dominant failure modes during intended warranty periods but also minimise design margins that might contribute to reliability after warranty. Electronic devices or components are becoming smaller, faster, and consuming less energy. These benefits are offset by significant impacts on device reliability because of increased electric fields, increased power densities, increased transistor counts, and increased sensitivity of nanoscale technology to manufacturing and design variability. In this manner, the progress associated with Moore's Law is associated with increasing impacts of electronics obsolescence coupled with reducing reliability of electronic components that reduces effectiveness of traditional approaches for managing obsolescence.

Traditional approaches for assessing, estimating, or improving reliability are largely ineffective for modern electronic equipment. Not withstanding this, traditional reliability methods continue to be called for contractually. Traditional reactive approaches to obsolescence are also increasingly ineffective, because of these reliability concerns, and the frequency and impacts of obsolescence are increasing. Because a number of failure mechanisms are only evident at small scales (e.g. modern microprocessors and memory are formally nanotechnology), several of observable failure modes appear counter-intuitive, occur intermittently, or contribute to "no fault found" phenomena that can account for a significant percentage of the cost of sustaining an electronic system. Modern components often have some form of error detection and recovery schemes, so can appear to function normally even after an internal failure occurs. Several failure modes of components occur in the non-installed or non-operating states. In addition to all of these concerns, there is the possibility that hardware designs can be deliberately subverted:

while modern commercial trends increase difficulty of subverting a hardware design, they also significantly increase the difficulty of detecting a subverted design.

Not withstanding these concerns, software remains a dominant cause of unreliability of electronic systems. Development and verification of software remains difficult, and software is often used to implement functionality that cannot be effectively achieved in any other manner. This is compounded by recent industry trends towards multicore processors. The hardware for these can be easily designed and implemented but these processors place significant demand on other hardware system components. Effective exploitation of multicore processors also requires a shift away from traditional sequential software development techniques towards much more difficult, rarely practiced, and relatively immature techniques for developing concurrent or parallel software.

Although Defence and Defence Industry are both at a strategic disadvantage in dealing with these concerns, due to the small and declining share of the electronic systems market, emerging practices overseas seek to better manage systems and capabilities through long service lives. In contrast with traditional reactive approaches, these emerging practices require up-front investment to capture requirements, architect systems, and to establish ongoing activities that seek to anticipate concerns and address them as early as possible.

Current Australian Defence policy requires implementation of obsolescence management activities, because of impacts in terms of life-cycle cost, availability, performance, supportability, and safety. However, current policy does not provide specific guidance about what those activities might entail.

In order to reduce its strategic disadvantage in dealing with the interrelated concerns of electronic system reliability and obsolescence, the Australian Defence Organisation needs to provide specific guidance for programs and non-specialist practitioners. Some elements of a way forward that might be considered include;

- Formulate guidance for inclusion in Defence Capability Plan (DCP) processes concerned with formulating an Obsolescence Management Plan that will be maintained until planned withdrawal of the capability.
- Mandate early development or update of an Obsolescence Management Plan for all minor and major programs that seek to acquire, modify, replace, or update electronic equipment.
- Avoid increasingly ineffective traditional and reactive approaches such as “Last Time Buy”, unless there is specific evidence they are sufficient to sustain the affected equipment through its planned life.
- Perform continuous product market research of technology trends, product applicability, and obsolescence status in order to predict future supportability concerns of electronic equipment and to formulate business and technical strategies for managing these concerns.
- Require approaches that emphasise reconfigurability and modularity of system designs and processes, in order to minimise the burden of system redesign and requalification.

- Require that software be architected to minimise impacts of computing hardware refresh.
- Support research concerned with better architecting software and refining software assurance policy.
- Support research concerned with evolving technical and business (particularly logistical) approaches concerned with better managing electronic system supportability both in new capabilities and in ageing in-service capabilities.
- Record the usage and failures of in-service equipment, and use this data to forecast reliability trends and proactively plan technology refresh and update activities.
- Minimise modification of COTS products where practical.
- Formulate and survey acquisition strategies to address the significant likelihood that commercially sourced electronic equipment will often be obsolete or physically unreliable before its introduction into service under current acquisition methods.
- Develop contractual models that reward suppliers who support implementation of strategies for proactively addressing concerns related to electronic system obsolescence and reliability.
- Seek to foster strategic business alliances with partners outside Defence and Defence industry that experience similar concerns related to reliability and obsolescence of high-criticality electronics, such as the automotive industry.

9. References

1. J. Blakemore, A. DeBarr, and J. Gunn (1953). "Semiconductor Circuit Elements", *Reports on Progress in Physics*, Vol. 16, pp. 160-215.
2. V. Packard (1963). "*The Waste Makers*", Penguin books, ISBN 0-1402-0589-1963.
3. T. Shilliday and J. Vaccaro, Editors (1962-1967) "*Physics of Failure in Electronics*", RADC Series in Reliability, Volumes 1-5.
4. M. Goldberg, A. Horberg, and H. Lauffenburger (Sep. 1964) "*Study of Comprehensive Failure Theory*", Technical Documentary Report No. RADC –TDR-64-309.
5. J. McCall (Mar. 1965) "Maintenance Policies for Stochastically Failing Equipment: A Survey", *Management Science*, Vol. 11, No. 5, Series A, Sciences, pp. 493-524.
6. G.E. Moore (19 Apr. 1965) "Cramming more components onto integrated circuits", *Electronics*, Vol 38, No. 8.
7. J. Black (Sep. 1969) "Electromigration Failure Modes in Aluminium Metallization for Semiconductor Devices", *Proceedings of the IEEE*, Vol. 57, Issue 9, pp. 1587-1594.
8. L. Chua (Sep. 1971) "Memristor – The Missing Circuit Element", *IEEE Transactions on Circuit Theory*, Vol. 18, Issue 5, pp. 507-519.
9. C. Horsting (Apr. 1972) "Purple Plague and Gold Purity", *10th Annual Reliability Physics Symposium*, pp. 155-158.

10. B. Mandakis (Apr. 1973) "The Solid Tantalum Capacitor – A "Solid" Contributor to Reliability", *11th Annual Reliability Physics Symposium*, pp. 45-53.

11. J. Black (Apr. 1974) "Physics of Electromigration", *12th Annual Reliability Physics Symposium*, pp. 142-149.

12. T. Gilb (11 Sep. 1975) "Software Metrics: State of the art", *Computer Weekly*, p. 6.

13. B. Boehm (Dec. 1976) "Software Engineering", *IEEE Transactions on Computers*, Vol. C-25, Issue 12, pp. 1226-1241.

14. T. McCabe (Dec. 1976) "A Complexity Measure", *IEEE Transactions on Software Engineering*, Vol. SE-2, Issue 4, pp. 308-320.

15. B. Livesay and E.Schreibner (Sep. 1977) "*Reliability Factors for Electronic Components in a Storage Environment*", Report No. DD-14-23, U.S. Army Missile Research and Development Command.

16. D. Yaney, J. Nelson, and L. Vanskike (Jan 1979) "Alpha-Particle Tracks in Silicon and their Effect on Dynamic MOS RAM Reliability", *IEEE Transactions on Electronic Devices*, Vol. ED-26, No. 1.

17. S. Mohanty (Sep. 1979) "Models and Measurements for Quality Assessment of Software", *Computing Surveys*, Vol. 11, No. 3.

18. A. Harris (1980) "Reliability in the Dormant Condition", *Microelectronics and Reliability*, Vol. 20, pp. 33-44.

19. H. Blanks (1980) "Electronics Reliability: A State–of-The-Art Survey", *Microelectronics and Reliability*, Vol. 20, pp. 219-245.

20. B. Boehm (1981) "*Software Engineering Economics*", Prentice Hall, ISBN 0138221227, 769pp.

21. J. Ziegler and W.Lanford (Jun. 1981) "The effect of sea level cosmic rays on electronic devices", *Journal of Applied Physics*, Vol. 52, No. 6, pp.4305-4312.

22. W. Harrison, K. Magel, R. Kluczny, A. DeKock (Sep. 1982) "Applying Software Complexity Metrics to Program Maintenance", *IEEE Computer*, Vol. 15, Issue 9, pp. 65-79.

23. A. Albrecht and J. Gaffney Jr (Nov. 1983) "Software Function, Source Lines of Code, and Development Effort Prediction: A Software Science Validation". *IEEE Transactions on Software Engineering*, Vol. SE-9, Issue 6, pp. 639-648.

24. M. Moss (1985) "*Designing for minimal maintenance expense: the practical application of reliability and maintainability*", CRC Press, ISBN 0824773144.

25. Nippon Telegraph and Telephone Corporation (1985) "*Standard Reliability Table for Semiconductor Devices*".

26. J. Gait (Mar. 1986) "A Probe Effect in Concurrent Programs", *Software – Practice and Experience*, Vol. 16, No. 3, pp. 225-233.

27. A. Avižienis and J. Laprie (May 1986) "Dependable Computing: From Concepts to Design Diversity", *Proceedings of the IEEE*, Vol. 74, Issue 5, pp. 629-638.

28. S. Balakrishnan and B. Wernerfelt (Jul.-Aug. 1986) "Technical Change, Competition, and Vertical Integration", *Strategic Management Journal*, Vol. 7, No. 4, pp. 347-359.

29. J. Bulow (Nov. 1986) "An Economic Theory of Planned Obsolescence", *The Quarterly Journal of Economics*, Vol. 104, No. 2, pp. 729-750.

30. J. Kearney, R. Sedlmeyer, W. Thompson, W. Gray, and A. Adler (Nov. 1986) "Software Complexity Measurement", *Communications of the ACM*, Vol. 29, Issue 11, pp. 1044-1050.

31. D. Kafura and G.Reddy (Mar. 1987) "The Use of Software Complexity Metrics in Software Maintenance", *IEEE Transactions on Software Engineering*, Vol. SE-13, Issue 3, pp. 335-343.

32. C. Symons (Jan. 1988) "Function Point Analysis: Difficulties and Improvements", *IEEE Transactions on Software Engineering*, Vol. 14, No. 1, pp. 2-11.

33. K. Wong (1988) "The Bathtub does not hold water any more", *Quality and Reliability Engineering International*, Vol. 4, 279-282.

34. K. Wong and D. Lindstrom (1988) "Off The Bathtub onto the Roller-Coaster Curve", *Proceedings Annual Reliability and Maintainability Symposium*.

35. E. Weyuker (Sep. 1988) "Evaluating Software Compexity Measures", *IEEE Transactions on Software Engineering*, Vol. 14, Issue 9, pp. 1357-1365.

36. T. Rajeevakumar, N. Lu, W. Henkels, W. Hwang, and R. Franch (Dec. 1988) "A New Failure Mode of Radiation-Induced Soft Errors in Dynamic Memories", *IEEE Electronic Device Letters*, Vol. 9, No. 12.

37. F.Jensen (24-26 Jan. 1989) "Component failures based on flaw distributions", *Proceedings IEEE and Maintainability Symposium*, pp. 91-95.

38. W. Finkelstein and J. Guertin (Feb. 1989) "*Integrated Logistics Support: The Design Engineering Link*", First Edition, ISBN-10: 0387506497, Springer.

39. K. wong (1989) "The Roller-Coaster Curve is in", *Quality and Reliability Engineering International*, Vol. 5, pp. 29-36.

40. J. Musa, A. Iannino, and K. Okumoto (1989) "*Software Reliability: Measurement*, *Prediction*, *Application*", ISBN 0-07-044119-7, McGraw Hill, 291pp.

41. S. Freiman and R. Pohanka (1989) "Review of Mechanically Related Failures of Ceramic Capacitors and Capacitor Materials", *Journal of the American Ceramic Society*, Vol. 72, No. 12, pp. 2258-2263.

42. S. Brocklehurst, P. Chan, B. Littlewood, and J. Snell (Apr. 1990) "Recalibrating Software Reliability Models", *IEEE Transactions on Software Engineering*, Vol. 16, No. 4, pp. 458-470.

43. L. Laranjeira (May 1990) "Software Size Estimation of Object-Oriented Systems", *IEEE Transactions on Software Engineering*, Vol. 16, No. 5.

44. H. Roland and B. Moriarity (1990) "*System Safety Engineering and Management*", 2nd Edition, Wiley-Interscience, 367pp, ISBN 0-471-61816-0.

45. S. Koford (11-16 May 1991 "Environmental Effects on Connector Reliability", *Proceedings of 41st Electronic Components and Technology Conference*, pp. 215-217.

46. American National Standards Institute (1991) "*Standard Glossary of Software Engineering Terminology*" ANSI/IEEE STD-729-1991.

47. K. Wong (1991) "The Physical Basis for the Roller-coaster Hazard Rate Curve for Electronics", *Quality and Reliability Engineering International*, Vol. 7, pp. 489-495.

48. Department of Defense (Dec. 1991) "*Reliability prediction of electronic equipment*", MIL-HDBK-217F.

49. R. Mroczkowski and J. Maynard (Dec. 1991) "Estimating the Reliability of Electrical Connectors", *IEEE Transactions on Reliability*, Vol. 40, Issue 5, pp. 507-512.

50. J. Verner and G. Tate (Apr. 1992) "A Software Size Model", *IEEE Transactions on Software Engineering*, Vol. 18, No. 4.

51. D. Campbell, J. Hayes, J. Jones, and A. Schwarzenberger (1992) "Reliability Behaviour of Electronic Components as A Function of Time", *Quality and Reliability Engineering International*, Vol. 8, pp. 161-166.

52. S. Brocklehurst and B. Littlewood (Jul. 1992) "New Ways to Get Accurate Reliability Measures", *IEEE Software*, Vol. 9, Issue 4, pp. 34-42.

53. Radio Technical Commission for Aeronautics (1 Dec. 1992) "*Software Considerations in Airborne Systems and Equipment Certification*", RTCA/DO-178B.

54. M. Pecht and V. Ramappan (Dec. 1992) "Are Components Still the Major Problem: A Review of Electronic System and Device Field Failure Returns", *IEEE Transactions on Components, Hybrids, and Manufacturing Technology*, Vol. 15, No. 6, pp. 1160-1164.

55. Centre National d'Etude des Télécommunications (1993) "*Handbook of Reliability Data for Electronic Components*", RDF-93, English Issue.

56. R. Mroszkowski (1993) "*Connector Design/Materials and Connector Reliability*", AMP Incorporated, Technical Paper P351-93.

57. R. Banker, S. Datar, C. Kemerer, and D. Zweig (Nov. 1993) "Software Complexity and Maintenance Costs". *Communications of the ACM*, Vol. 36, Issue 11, pp. 81-94.

58. M. Cushing, D. Morton, T. Stadterman, and A. Malhotra (Dec. 1993) "Comparison of Electronics-Reliability Assessment Approaches", *IEEE Transactions on Reliability*, Vol. 42, No. 4, pp. 542-546.

59. A. Fishman, N. Gandal, and O. Shy (Dec. 1993) "Planned Obsolescence as an Engine of Technological Process", *The Journal of Industrial Economics*, Vol. 41, No. 4, pp. 361-370.

60. C. Gossett, B. Hughlock, M. Katoozi, and G. LaRue (Dec. 1993) "Single Event Phenomena in Atmospheric Neutron Environments", *IEEE Transactions on Nuclear Science*, Vol. 40, No. 6.

61. J. Matson, B. Barrett, and J. Mellichamp (Apr. 1994) "Software Development Cost Estimation Using Function Points", *IEEE Transactions on Software Engineering*, Vol. 20, No. 4.

62. M. Pecht and F. Nash (1994) "Predicting the reliability of electronic equipment", *IEEE Proceedings*, Vol. 82, pp. 992-1004.

63. W.J. Perry (29 Jun. 1994) "Specifications & Standards – A New Way of Doing Business", Memorandum for Secretaries of the Military Departments.

64. B. Sorenson, G. Kelly, A. Sajecki, and P. Sorenson (20-22 Sep. 1994) "An analyser for Detecting Intermittent Faults in Electronic Devices", *IEEE Systems Readiness Technology Conference Proceedings*, pp. 417-421.

65. A. Taber and E. Normand (Feb. 1995) "*Investigation and Characterization of SEU Effects and Hardening Strategies in Avionics*", DNA-TR-94-123, Defense Nuclear Agency.

66. C.Yang and A. Takeda (1995) "*Hot Carrier Effects in MOS Devices*", Elsevier, ISBN 9780126822403.

67. British Telecom (1995) "*Handbook of Reliability Data*", HRD-5.

68. The Standish Group (1995) "*The Standish Group Report: Chaos*".

69. J. Pecht and M. Pecht (1995) "*Long-term non-operating reliability of electronic products*", CRC Press (Boca Raton), ISBN 08049396212, 119pp.

70. D. Tegarden, S. Sheetz, D. Monarchi (1995) "A software complexity model of object-oriented systems", *Decision Support Systems*. Vol. 13, pp. 241-262.

71. S. Liu, V. Stavridou, and B. Duterte (1995) "The Practice of Formal Methods in Safety-Critical Systems", *Journal of Systems Software*, Vol. 28, pp. 77-87.

72. J. Laprie (24-27 Oct. 1995) "Dependability of Computer Systems: Concepts, Limits, Improvements", *Proceedings of Sixth International Symposium on Software Reliability Engineering*, pp. 2-11.

73. R. Jeffery and J. Stathis (Jan. 1996) "Function Point Sizing: Structure, Validity, and Applicability", *Empirical Software Engineering*, Vol. 1, No. 1, pp. 11-30.

74. J.D. Moteff (1996) "*The Role of DoD's Investment in Electronics on the Decline of the Consumer Electronics Industry*", The Industrial College of the Armed Forces, National Defense University, Fort McNair, Washington D.C., 20319-5062.

75. M. Waldman (1996) "Planned Obsolescence and the R&D Decision", *The Rand Journal of Economics*, Vol. 27, No. 3, pp. 583-595.

76. M. Lyu (1996) Editor in Chief "*Handbook of Software Reliability Engineering*", McGraw Hill (IEEE Computer Society Press), ISBN 0-07-039400-8.

77. 104th Congress (1996) "Text of Law Requiring Value Engineering in Executive Agencies", National Defense Authorization Act for Fiscal Year 1996, Page 110 Stat. 186, Public Law 104-106.

78. A. Acovic, G. La Rosa, and Y. Sun (1996) "A Review of Hot-Carrier Degradation Mechanisms in MOSFETS", *Microelectronics Reliability*, Vol. 36, No. 7/8, pp. 845-869.

79. A. Nishino (1996) "Capacitors: operating principles, current market, and technical trends", *Journal of Power Sources*, Vol. 60, pp. 137-147.

80. International Organisation for Standardisation (15 Jun. 1996) "*Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*", ISO/IEC 7498-1, Second Edition (Corrected).

81. C. Roark (27-31 Oct. 1996) "SAE AS 4893 Generic Open Architecture (GOA) Framework", *15th AIAA/IEEE Digital Avionics Systems Conference*, pp. 217-222.

82. B. Kitchenham (Mar./Apr. 1997) "The Problem with Function Points", *IEEE Software*, Vol. 14, Issue 2, pp. 29-31.

83. T. Hall and N. Fenton (Mar./Apr. 1997) "Implementing Effective Software Metrics Programs", *IEEE Software*, Vol. 14, Issue 2, pp. 55-65.

84. G. Harman (1997) "*Wire Bonding in Microelectronics: materials, processes, reliability and yield*", 2nd Edition, ISBN 0-07-032619-3, McGraw Hill.

85. N. Lynn and N. Singpurwalla (1997) "Comment: Burn-in makes us feel good", *Statistical Science*, Vol. 12, pp. 13-19.

86. D. Humphrey, and F. McCluskey (Jun. 1997) "Uprating Electronic Components for Use Outside Their Temperature Specification Limits", M. Wright, *IEEE Transactions on Components, Packaging, and Manufacturing Technology, Part A*. Vol. 20, Issue 2, pp. 252-256.

87. UK Ministry of Defence (1 Aug. 1997) "*Requirements for Safety Related Software in Defence Systems*", Defence Standard 00-55, Issue 2, (Part No. 1: Requirements and Part No. 2 Guidance) [Obsolescent as of 29 Apr. 2005].

88. R. Gonzalez, B. Gordon, and M. Horowitz (Aug 1997) "Supply and Threshold Voltage Scaling for Low Power CMOS", *IEEE Journal of Solid-State Circuits*, Vol. 32, No. 8.

89. American National Standards Institute (Sep. 1997) "*Product Life Cycle Data Model*", ANSI/EIA-724-97.

90. A. Dasgupta, E. Magrab, D. Anand, K. Eisinger, J. McLeish, M. Torres, P. Lall, and T. Dishongh (Dec. 1997) "Perspectives to Understand Risks in the Electronic Industry", *IEEE Transactions on Components, Packaging, and Manufacturing Technology, Part A*, Vol. 20, No. 4.

91. R. Banker, G. Davis and S. Slaughter (Apr. 1998) "Software Development Practices, Software Complexity, and Software Maintenance Performance: A Field Study", *Management Science*, Vol. 44, No. 4, pp. 433-450.

92. F.G. Wilkie and B. Hylands (25 Apr. 1998) "Measuring Complexity in C++ Application software", *Software – Practice and Experience*, Vol. 28, No. 5, pp. 513-546.

93. C. Disco, B. van der Meulen (editors) (1998) "*Getting new technologies together*", Walter de Gruyter GmbH &Co., New York, ISBN 311015630X.

94. M. Ohring (1998) "*Reliability and Failure of Electronic Materials and Devices*", Academic Press, ISBN 0125249853, 692pp.

95. IEEE (1998) "*Standard for Application and Management of the Systems Engineering Process – Description*", IEEE STD 1220-1998.

96. W. Meeker and L. Escobar (1998) "*Statistical Methods for Reliability Data*", Wiley, ISBN 9780471143284.

97. R. Williams, J. Banner, I. Knowles, M. Dube, M. Natishan, and M. Pecht (1998) "An Investigation of "Cannot Duplicate" Failures" *Quality and Reliability Engineering International*, Vol. 14, pp. 331-337.

98. D. Dramjanovic (1998) "Ferroelectric, dielectric, and piezoelectric properties of ferroelectric thin films and ceramics", *Reports on Progress in Physics*, Vol. 61, pp. 1267-1324.

99. Health and Safety Commission (1998) "*The use of computers in safety-critical applications*", Final report of the study group on the safety of operational computer systems, Her Majesty's stationery Office, ISBN 0-7176-1620-7.

100. M. Pecht and R. Cogan (24-26 Aug. 1988) "Intelligent Derating for Reliability", *IEEE International Symposium on Intelligent Control*, pp. 98-102.

101. G. Ebel (Sep. 1998) "Reliability Physics in electronics: A Historical View", *IEEE Transactions on Reliability*, Vol. 47, Issue 3.

102. Department of Defense (1 Oct 1998) "*Electronic Reliability Design Handbook*", MIL-HDBK-338B.

103. J. Stathis and D. DiMaria (6-9 Dec. 1998) "Reliability Projection for Ultra-Thin Oxides at Low Voltage", *International Electron Devices Meeting (IEDM)*, pp. 167-170.

104. IEEE (Dec. 1998) "*IEEE Standard for a Software Quality Metrics Methodology*", IEEE Std 1061-1998.

105. S. Dutta, M. Lee, and L. Van Wassenhove (May/Jun. 1999) "Software Engineering in Europe": A Study of Best Practices", *IEEE Software*, Vol. 16, Issue 3, pp. 82-90.

106. W. Wondrak (1999) "Physical Limits and Lifetime Limitations of Semiconductor Devices at High Temperator", *Microelectronics Reliability*, Vol. 39, pp. 1113-1120.

107. American National Standards Institute (1999) "*Processes for Engineering a System*", ANSI/EIA-632-1999.

108. Harper-Collins (1999) "*Collins compact Australian Dictionary*", ISBN-0-00-470688-9, Harper-Collins, Sydney.

109. K. Linberg (1999) "Software developer perceptions about software project failure: a case study", *The Journal of Systems and Software* Vol. 49, pp. 177-192.

110. N. Fenton and M. Neil (1999) "Software metrics: successes, failures, and new directions", *The Journal of Systems and Software*, Vol. 47, pp. 149-157.

111. S. Borkar (Jul.-Aug. 1999) "Design challenges of technology scaling", *IEEE Micro*, Vol. 19, Issue 4, pp. 23-29.

112. N.-C. Lee (Sep.-Oct. 1999) "Lead-Free Soldering – Where The World is Going", *Advancing Microelectronics*, pp. 29-34.

113. D. Parrish and J. James (24-29 Oct. 1999) "Evaluation of the Generic Open Architecture Framework", *18th Digital Avionics Systems Conference*, Vol. 2, pp. 9.B.1-1 to 6.B.1.7.

114. Siemens (9 Nov. 1999) "*Failure Rates of Electronic Components*", Siemens Company Standard SN29500, Version 6.0.

115. Radio Technical Commission for Aeronautics (19 Apr. 2000) "*Design Assurance Guidance for Airborne Electronic Hardware*", RTCA/DO-254.

116. R. Degraeve and G. Groeseneken (May 2000) "Reliability: a possible showstopper for oxide thickness scaling?", *Semiconductor Science and Technology*, Vol. 15, No. 5, pp. 436-444.

117. R. Ramakumar (2000) "Reliability Engineering", Chapter 10 of "*The Electrical Engineering Handbook*", R.C. Dorf, Boca Raton: CRC Press LLC.

118. D. Steinberg (2000) "*Vibration Analysis for Electronic Equipment*", Third Edition, Wiley, ISBN 0-471-37685-X.

119. W. Tomszykowski, A. Fritz, and R. Scalia (2000) "*Program Managers Handbook: Common Practices to Mitigate the Risk of Obsolescence*", Maryland: Defense Microelectronics Activity.

120. D. Das, N. Pendse, M. Pecht, L. Condra, and C. Wilkinson (Sep. 2000) "Deciphering the Deluge of Data", *IEEE Circuits and Devices Magazine*, Vol. 16, Issue 5, pp. 26-34.

121. D. Humphrey, L. Condra, N. Pendsé, D. Das, C. Wilkinson, and M. Pecht (Sep. 2000) "An Avionics guide to Uprating of Electronic Parts", *IEEE Transactions on Components and Packaging Technologies*, Vol. 23, Issue 3.

122. Sony Semiconductor (Oct. 2000) "*Quality and Reliability Handbook*".

123. E. Hitt (7-13 Oct. 2000) "Rebuilding the Requirements Process for Aging Avionics", *Proceedings of the 19th Digital Avionics Systems Conference*, Vol. 1, pp. 4A3/1-4A3/6.

124. R. Solomon, P. Sandborn, and M. Pecht (Dec. 2000) "Electronic Part Life Cycle Concepts and Obsolescence Forecasting", *IEEE Transactions on Components and Packaging Technologies*, pp.707-717.

125. J. Stathis (Mar. 2001) "Physical and Predictive Models of Ultrathin Oxide Reliability in CMOS Devices and Circuits", *IEEE Transactions on Device and Materials Reliability*, Vol. 1, Issue 1, pp. 43-59.

126. T. Hastings and A. Sajeev (Apr. 2001) "A Vector-Based Approach to Software Size Measurement and Effort Estimation", *IEEE Transactions on Software Engineering*, Vol. 27, No. 4.

127. National Research Council (11 May 2001) "*Aging Avionics in Military Aircraft*", National Research Council on the Review of Aging Avionics in Military Aircraft, AFSB-J99-03-A.

128. Telecordia (May 2001) "*Reliability Prediction Procedure for Electronic Equipment*", Telcordia Technologies Special Report SR-332, Issue 1, Telcordia Customer Service, Piscataway, N.J.

129. L. Petersen (Jun. 2001) "The Use of Commercial Components in Defense Equipment to Mitigate Obsolescence. A Contradiction in Itself?", pp. 1-1 to 1-8 in NATO RTO-MP-072, AC/323(SCI-084)TP/31, "*Strategies to Mitigate Obsolescence in Defence Systems using Commercial Components*".

130. D. Young (Jun. 2001) "New Approaches to Processor Lifecycle Management", pp. 2-1 to 2-4 in NATO RTO-MP-072, AC/323(SCI-084)TP/31, "*Strategies to Mitigate Obsolescence in Defence Systems using Commercial Components*".

131. R. Cernko, D. Jäger, and R. Manser (Jun. 2001) "Consequences for the design of military aircraft systems due to integration of commercial electronic components in avionics", pp. 5-1 to 5-10 in NATO RTO-MP-072, AC/323(SCI-084)TP/31, "*Strategies to Mitigate Obsolescence in Defence Systems using Commercial Components*"

132. S. Ratikin (Jul. 2001) "*Software Verification and Validation for Practitioners and Managers*", Artech House, ISBN 1580532969, 420pp.

133. Federal Aviation Administration (Aug. 2001) "*Review of Pending Guidance and Industry Findings on Commercial Off-The-Shelf (COTS) Electronics in Airborne Systems*", DOT/FAA/AR-01/41, Final Report.

134. E. Ing, K. Hunt, and S. Haug (6 Sep. 2001) "Challenges of Component Obsolescence", Presentation to *Royal Aeronautical Society (RAeS) Conference*.

135. J. Jones and J. Hayes (Sep. 2001) "Investigation of the Occurrence of: No-Faults-Found in Electronic Equipment", *IEEE Transactions on Reliability*, Vol. 50, Issue 3, pp. 289-292.

136. E. Hitt and B. Zwitch (14-18 Oct. 2001) "Aging Avionics: The Problems and the Challenges", *The 20th Conference on Digital Avionics Systems*, Vol. 1, pp. 3A2/1-3A2/8.

137. L. Condra, D. Das, C. Wilkinson, N. Pendse, and M. Pecht (Dec. 2001) "Junction Temperature Considerations in Evaluating Parts for Use Outside Manufacturer-Specified Temperature Ranges", *IEEE Transactions on Components and Packaging Technologies*, Vol. 24, Issue 4, pp. 721-728.

138. W. Shawlee II and D. Humphrey (Dec. 2001) "Aging Avionics – What Causes It and How to Respond", *IEEE Transactions on Components and Packaging Technologies*, Vol. 24, Issue 4, pp. 739-740.

139. E. Wu, J. Suñé, W. Lai, E. Nowak, J. McKenna, A. Vashenker, and D. Harmon (2002) "Interplay of voltage and temperature acceleration of oxide breakdown for ulta-thin gate oxides", *Solid State Electronics*, Vol 46, pp. 1787-1798.

140. D. Kececioglu and F. Sun (2002) "*Environmental Stress Screening: It's Quantification, Optimization, and Management*", DESTech Publications Inc., ISBN 1932078045.

141. JEDEC Solid State Technology Association (2002) "*Failure mechanisms and models for semiconductor devices*", JEP-122A.

142. B. Foucher, J. Boullie, B. Meslet, and D. Das (2002) "A review of reliability prediction methods for electronic devices", *Microelectronics Reliability* Vol. 42, pp. 1155-1162.

143. C. Harris and A. Piersol (Editors) (2002) "*Harris' Shock and Vibration Handbook*", McGraw-Hill, ISBN 0-07-137081-1.

144. A. Dziedzic (2002) "Electrical and structural investigations in reliability characteristics of modern passives and passive integrated components", *Microelectronics Reliability*, Vol. 42, pp. 709-719.

145. J. Procaccino, J. Verner, S. Overmeyer, and M. Darter (2002) "Case study: factors for early prediction of software development success", Information and *Software Technology*, Vol. 44, pp. 53-62.

146. D. Tran-Cao, G. Levesque, and A. Abran (2002) "Measuring Software Functional Size: Towards an Effective Measurement of Complexity", Proceedings of the *IEEE International Conference on Software Maintenance*, pp. 370-376.

147. H. Wang (2002) "A survey of maintenance policies of deteriorating systems", *European Journal of Operational Research*, Vol. 139, pp. 469-489.

148. D. Thomas, K. Ayers, and M. Pecht (2002) "The "trouble not identified" phenomenon in automotive electronics", *Microelectronics Reliability*, Vol. 42, pp. 641-651.

149. I. Beniaminy and D. Joseph (2002) "Reducing the "No Fault Found" Problem: Contributions from Expert-System Methods", *IEEE Aerospace Conference Proceedings*, Vol. 6, pp. 6-2971 to 6-2973.

150. D. Humphrey, W.Shawlee II, P. Sandborn, and D. Lorenson (2002) "Utilization Life of Electronic Systems – Aging Avionics Usable Life and Wear-Out Issues", Presented at *World Avionics Congress*.

151. N. Pendsé, D. Thomas, D. Das, and M. Pecht (Jun. 2002) "Uprating of a Single Inline Memory Module", *IEEE Transactions on Components and Packaging Technologies*, Vol. 25, Issue 2, pp. 266-269.

152. M. Caffey, P. Graham, E. Johnson, M. Wirthlin, and C. Carmichael (Sep. 2002) "*Single-Event Upsets in SRAM FPGAs*", LA-UR-02-6138, Los Alamos National Laboratory.

153. J. Huselius (23 Sep. 2002) "*Debugging Parallel Systems: A State of the Art Report*", MRTC Report No. 63, Department of Computer Engineering, Mälardalens University, Västerås, Sweden.

154. R. Glass (Nov. 2002) "Sorting out software complexity", *Communications of the ACM*, Vol. 45, Issue 11, pp.19-21.

155. B. Linder, S. Lombardo, J. Stathis, A. Vayshenker, and D. Frank (Nov. 2002) "Voltage Dependence of Hard Breakdown Growth and the Reliability Implication in Thin Dielectrics", *IEEE Electron Device Letters*, Vol. 23, Issue 11, pp. 661-663.

156. J. Macher, D. Mowery, and T. Simcoe (Dec. 2002) "e-Business and Disintegration of the Semiconductor Industry Value Chain", *Industry and Innovation*, Vol. 9, No. 3, pp. 155-181.

157. Federal Aviation Administration (Dec. 2002) "*Study of Commercial Off-The-Shelf (COTS) Real-Time Operating Systems (RTOS) in Aviation Applications*", Final Report, DOT/FAA/AR-02/118.

158. The European Parliament and of The Council of The European Union (27 Jan. 2003) "Directive 2002/95/EC on the restriction of the use of certain hazardous substances in electrical and electronic equipment", *Official Journal of the European Union*, L 37/19.

159. G. Klutke, P. Kiessler, and M. Wortman (Mar. 2003) "A Critical Look at the Bathtub Curve", *IEEE Transactions on Reliability*, Vol. 52, No. 1.

160. Federal Aviation Administration (17 Mar. 2003) "*Guidelines for the Certification, Airworthiness and Operational Approval of Electronic Flight Bag Computing Devices*", FAA Advisory Circular, AC 120-76A.

161. J. Shao and Y. Wang (Apr. 2003) "A new measure of software complexity based on cognitive weights", *Canadian Journal of Electrical and Computer Engineering*, Vol. 28, Issue 2, pp. 69-74.

162. A. Teverovsky (Apr. 2003) "*Introducing a New Member to the Family: Gold Whiskers*", Internal Memorandum-NASA Goddard Space Flight Center.

163. J. Ganssle and M. Barr (2003) "*Embedded Systems Dictionary*", CMP Books, ISBN 1-57820-120-9, 291pp.

164. P. Marwedel (2003) "*Embedded Systems Design*", Kluwer Academic Publishers, ISBN 1-4020-7690-8, 241pp.

165. J. Licari (2003) "*Coating Materials for Electronic Applications: Polymers*, *Processes*, *Reliability*, *Testing*", Noyes Publications, NY, ISBN 0-8155-1492-1.

166. G. Shaffer and G. McPherson (2003) "*FAA COTS Risk Mitigation Guide: Practical Methods for Effective COTS Acquisition and Life Cycle Suppor*t", Revision 3.1, Federal Aviation Administration.

167. I. James, D. Lumbard, I. Willis, and J. Goble (2003) "Investigating No Fault Found in the Aerospace Industry", *Annual Reliability and Maintainability Symposium*, pp. 441-446.

168. G. Baskoro, J. Rouvroye, W. Bacher, A. Brombacher (2003) "Developing MESA: An Accelerated Reliability Test", *Proceedings of Annual Reliability and Maintainability Symposium* (RAMS), pp. 303-308.

169. M. Thomas (2003) "Issues in Safety Assurance" in *Lecture Notes in Computer Science*, G.Goos, J. Hartmanis, and J. van Leeuwen (Editors), Volume 2788/2003, pp. 1-7, Springer, ISBN 978-3-540-20126-7.

170. P. Dodd and L. Massengill (Jun. 2003) "Basic Mechanisms and Modelling of Single-Event Upset in Digital Microelectronics", *IEEE Transactions on Nuclear Science*, Vol. 50, No. 3, pp. 583-602.

171. APT News (21 Jul 2003) "*The Bill Gates Method*".

172. E. Kesseler (Aug. 2003) "*Consistent safety objectives and COTS versus fragmented certification practices and good safety records: Air transport dilemma in need of innovation*", NLR-TP-2003-378, National Aerospace Laboratory NLR (Nationaal Lucht- en Ruimtevaartlaboratorium), Netherlands.

173. Actel Corporation (Sep. 2003) "*Reliability Considerations for Automotive FPGAs*", White Paper.

174. K. Moløkken-Østvold and M. Jørgensen (30 Sep-1 Oct. 2003) "A Review of surveys on Software Effort Estimation", *International Symposium on Empirical Software Engineering*, pp. 223-230, ISBN 0-7695-2002-2.

175. Department of the Army (31 Dec. 2003) "*Army Acquisition Policy*", Army Regulation 70-1, Headquarters, Department of the Army, Washington DC.

176. M. Economou (26-29 Jan. 2004) "The Merits and Limitations of Reliability Predictions", *Reliability and Maintainability Annual Symposium*, pp. 352-357, ISBN 0-7803-82115-3.

177. A. Avižienis, J. Laprie, B. Randell, and C. Landwehr (Jan.-Mar. 2004) "Basic Concepts and Taxonomy of Dependable and Secure Computing", *IEEE Transactions on Dependable and Secure Computing*, Vol. 1, Issue. 1, pp. 11-33.

178. QinetiQ and ARINC (Mar. 2004) "*Ministry of Defence Component Obsolescence Resolution Cost Metrics Study*".

179. P. Singh, P. Sandborn, T. Geiser, and D. Lorenson (2004) "Electronic Part Obsolescence Driven Product Redesign Planning", *International Journal of Agile Manufacturing*, Vol. 7, No. 1, pp. 23-32.

180. F. Wang (1-4 Jun. 2004) "Airborne Electronics Equipment Fastener Joint Failure Study", *9th Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*, Vol. 1, pp. 607-611.

181. J. Srinivasan, S. Adve, P. Bose, J. Rivers (Jun. 2004) "The Impact of Technology Scaling on Lifetime Reliability", *International Conference on Dependable Systems and Networks*.

182. D. Pinsky, M. Osterman, and S. Ganesan (Jun 2004) "Tin Whiskering Risk Factors" *IEEE Transactions on Components and Packaging Technologies*, Vol. 27, Issue 2, pp. 427-431.

183. FIDES Group (2004) "*FIDES Guide 2004*, *Issue A*, *Reliability Methodology for Electronic Systems*", DGA-DM/STTC/CO/477-A.

184. International Organisation for Standardisation (2004) "*Software Engineering – Guidelines for the application of ISO 9001:2000 to computer software*", ISO/IEC 90003:2004.

185. C. Kramer and W. Bond (2004) "Software Engineering Metrics: What Do They Measure and How Do We Know?", *10th International Software Metrics Symposium*.

186. S. Mercier and P. Labeau (2004) "Optimal replacement policy for a series system with obsolescence", *Applied Stochastic Models in Business and Industry*", Vol. 20, pp. 73-91.

187. B. Orbach (2004) "The Durapolist Puzzle: Monopoly Power in Durable-Goods Markets", *Yale Journal on Regulation*, Vol. 21, pp. 67-118.

188. Department of Defense (18 Jun 2004) "*Test method standard: Microcircuits*", MIL-STD-883F.

189. K. Flowers and C. Azani (25-28 Oct. 2004) "Open Systems Policies and Enforcement Challenges", *Proceedings of the National Defence Industrial Association Systems Engineering Conference*, Dallas Texas.

190. G. Galyon and R. Gedney (Aug. 2004) "Avoiding Tin Whisker Reliability Problems", *Circuits Assembly*, pp. 26-31.

191. M. Pecht, Y. Fukuda, and S. Rajagopal (Oct. 2004) "The Impact of Lead-Free Legislation Exemptions on the Electronics Industry". *IEEE Transactions on Electronics Packaging Manufacturing*, Vol. 24, Issue 4, pp. 221-232.

192. G. Galyon (Jan. 2005) "Annotated Tin Whisker Bibliography and Anthology", *IEEE Transactions on Electronics Packaging Manufacturing*, Vol. 28, No. 1, pp. 94-122.

193. Department of Defence (9 Mar. 2005) "*Defence Policy on obsolescence management*", Australian Government, Defence Instruction (General) LOG-07-19.

194. G. Groeseneken, R. Degraeve, B. Kaczer, and P. Roussel (4-7 Apr. 2005) "Recent trends in reliability assessment of advanced CMOS technologies", *Proceedings of the IEEE Conference on Microelectronic Test Structures*, pp. 81-88.

195. J. Srinivasan, S.V. Adve, P. Bose, J.A. Rivers (May-Jun 2005) "Lifetime Reliability: Toward an Architectural Solution", *IEEE Micro*, Vol. 25, Issue 3, pp 70-80.

196. J. Stathis (9-11 May 2005) "Impact of ultra thin oxide breakdown on circuits", *International Conference on Integrated Circuit Design and Technology*, pp. 123-127.

197. Z. Mei, M. Ahmad, M. Hu, and G. Ramakrishna "Kirkendall (31 May-3 Jun. 2005) "Voids at Cu/Solder Interface and Their Effects on Solder Joint Reliability", *Proceedings 55th Electronic Components and Technology Conference*, Vol. 1, pp. 415-420.

198. S. Sheu and Y. Chen (2005) "Optimal burn-in time to minimize the cost for general repairable products sold under warranty", *European Journal of Operational Research*, Vol. 163, pp. 445-461.

199. M. Alam and S. Mahapatra (2005) "A comprehensive model of PMOS NBTI degradation", *Microelectronics Reliability*, 45, pp. 71-81.

200. R. Mishra (2005) "*An Uprateability Risk Assessment Methodology*", M.Sc. Thesis, Graduate School of the University of Maryland.

201. J. Licari and D. Swanson (2005) "*Adhesives Technology for Electronic Applications*", William Andrew Publishing, NY, ISBN 0-8115-1513-8.

202. J. Musa (2005) "*Software Reliability Engineering: More Reliable Software Faster and Cheaper*", 2nd Edition, Authorhouse, ISBN 1-4184-9387-2.

203. J. Procaccino, J. Verner, K. Shelfer, D. Gefen (2005) "What do software practitioners really think about project success: an exploratory study", *Journal of Systems and Software*, Vol. 78, pp. 194-203.

204. B. Paul, K. Kang, H. Kufluoglu, A. Alam, and K. Roy (Aug 2005) "Impact of NBTI on the Temporal Performance Degradation of Digital Circuits", *IEEE Electron Device Letters*, Vol. 26, No. 8.

205. H. Sutter and J. Larus (Sep. 2005) "Software and the Concurrency Revolution", *Queue*, Vol. 3, Issue 7, pp. 54-62.

206. B. Steadman, S. Sievert, B. Sorenson, and F. Berghout (26-29 Sep. 2005) "Attacking "Bad Actor" and "No Fault Found" Electronic Boxes", *IEEE Systems Readiness Technology Conference* (AUTOTESTCON), pp. 821-824.

207. J. Boyle (Oct. 2005) "A novel approach to obsolescence management", *Defense Electronics*, pp. 20-25.

208. S. Borkar (Nov.-Dec. 2005) "Designing Reliable Systems from Unreliable Components: The Challenges of Transistor Variability and Degradation", *IEEE Micro*, Vol. 25, Issue 6, pp. 10-16.

209. E. Hitt (Dec. 2005) "Aging Avionics and Net-Centric Operations", *IEEE Aerospace and Electronic Systems Magazine*, Vol. 20, Issue 12.

210. D. Radaelli, H. Puchner, S. Wong, and S. Daniel (Dec. 2005) "Investigation of Multi-bit Upsets in a 150nm technology SRAM Device", *IEEE Transactions on Nuclear Science*, Vol. 52, No. 6, pp.2433-2437.

211. J. McLinn (23-26 Jan. 2006) "Creating a Sensible Derating System", *Annual Reliability and Maintainability Symposium*, pp. 362-367.

212. L. Merola (Feb. 2006) "The COTS Software Obsolescence Threat", *Fifth Conference on Commercial-off-the-shelf (COTS)-Based Software Systems*, ISBN 0-7695-2515-6.

213. N. Vichare and M. Pecht (Mar. 2006) "Prognostics and Health Management of Electronics", *IEEE Transactions on Components and Packaging Technologies*, Vol 29, No. 1, pp. 222-229.

214. S. Openshaw (31 Mar. 2006) "*Performance Based Logistics: A Path to Reduced Reliance on Contractor Technical Support for Weapon Systems in the Field?*", Civilian Research Report, US Army War College.

215. P. Singh and P. Sandborn (Apr-Jun 2006) "Obsolescence Driven Design Refresh Planning for Sustainment-Dominated Systems", *The Engineering Economist*, Vol. 51, No. 2, pp. 115-139.

216. J.B. Bernstein, M. Gurfinkel, X Li, J. Walters, Y. Shapira, and M. Talmor (2006) "Electronic circuit reliability modelling", *Microelectronics Reliability* Vol. 46, pp. 1957-1979.

217. J. Stathis and S. Zafar (2006) "The negative bias temperature instability in MOS devices: A review", *Microelectronics and Reliability*, Vol. 46, Issues 2-4, pp. 270-286.

218. M. Pecht and D. Humphrey (2006) "Uprating of electronic parts to address obsolescence", *Microelectronics International*, Vol. 23, No. 2, pp. 32-36.

219. A. Goel and R. Graves (Jun. 2006) "Electronic System Reliability: Collating Prediction Models", *IEEE Transactions on Device and Materials Reliability*, Vol. 6, Issue 2, pp. 258-265.

220. NEC Electronics Corporation (Jun. 2006) "*Optical/Microwave Semiconductor Devices, Review of Quality and Reliability Handbook*", Document No. PQ10478EJ02V0TN (2nd edition).

221. S. Thompson and S. Parthasarathy (Jun. 2006) "Moore's Law: the future of Si microelectronics", *Materials Today*, Vol. 9, No. 6.

222. S. Borkar (Jun. 2006) "Tackling variability and reliability challenges". *IEEE Design and Test of Computers*, Vol. 23, Issue 6, p. 520.

223. L. Xu and J. Pang (2006) "Effect of Intermetallic and Kirkendall Voids Growth on Board Level Drop Reliability for SnAgCu Lead-free BGA Solder Joint", *Electronics Components and Technology Components*.

224. M. Jørgensen and K. Moløkken-Østvold (2006) "How large are software cost overruns? A review of the 1994 CHAOS report", *Information and Software Technology* Vol. 48, pp. 297-301.

225. H. Pham (2006) "*System Software Reliability*", Springer-Verlag, ISBN-10: 1-85233-950-0.

226. W. Wang, D. Pan, and M. Chen (2006) "Architecture-based software reliability modelling", *The Journal of Systems and Software*, Vol. 79, pp. 132-146.

227. J. Procaccino and J. Verner (2006) "Software project managers and project success: An exploratory study", *The Journal of Systems and Software*, Vol. 79, pp. 1541-1551.

228. B. Meyer (2006) "*Dependable Software*",pp. 1-33, in "*Dependable Systems*", Lecture Notes in Computer Science Volume 4028, Part I, ISBN 978-3-540-36821-2, Springer-Verlag Berlin.

229. National Transportation Safety Board (2006) "*Safety Report on the Treatment of Safety-Critical Systems in Transport Airplanes*", Safety Report NTSB/SR-06/02, National Transportation Safety Board, Washington DC.

230. J. McDermid and T. Kelly (2006) "Software in Safety Critical Systems: Achievement and Prediction", *Nuclear Future*, Vol. 2, No. 03.

231. H. Chockler, E. Farchi, Z. Glazberg, B. Godlin, Y. Nir-Buchbinder, and I. Rabinovitz (17 Jul. 2006) "Formal Verification of Concurrent Software: Two Case Studies", *International Symposium on Software Testing and Analysis*, pp. 11-22.

232. W. Pierskalla and J. Voelker (21 Nov. 2006) "A Survey of Maintenance Models: The Control and Surveillance of Deteriorating Systems", *Naval Research Logistics Quarterly*, Vol. 23, Issue 3, pp. 353-388.

233. A. Tipton, J. Pellish, R. Reed, R. Schrimpf, R. Weller, M. Mendenhall, B. Sierawski, A. Sutton, R. Diestelhorst, G. Espinel, J. Cressler, P. Marshall, and G. Vizkelethy (Dec. 2006) "Multiple-Bit Upset in 130nm CMOS Technology", *IEEE Transactions on Nuclear Science*, Vol. 53, Pt. 1, No. 6, pp.3259-3264.

234. K. Asanovic, R. Bodik, B. Catanzaro,J. Gebis, P. Husbands, K. Keutzer, D. Patterson, W. Plishker, J. Shalf, S. Williams, and K. Yelick (18 Dec. 2006) "*The Landscape of Parallel Computing Research: A View from Berkeley*", Technical Report No. UCB/EECS-2006-183, Electrical Engineering and Computer Sciences, University of California at Berkeley.

235. P. Parkinson and L. Kinnan (Dec. 2006-Jan. 2007) "Putting COTS back in the box". *Electronics Systems and Software*, Vol. 4, Issue 6, pp. 26-29.

236. J. Smetana (Jan. 2007) "Theory of Tin Whisker Growth: "The End Game"", *IEEE Transactions on Electronics Packaging Manufacturing*, Vol. 30, No. 1.

237. S. Dick (Jan. 2007) "Software-Reliability Modeling: The Case for Deterministic Behavior", *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, Vol. 37, No. 1.

238. N. Schneidewind (Jan-Feb 2007) "A Quantitative Approach to Software Development Using IEEE 982.1", *IEEE Software*, Vol. 24, Issue 1, pp. 65-72.

239. S. Gokhale (Jan.-Mar. 2007) "Architecture-Based Software Reliability Analysis: Overview and Limitations", *IEEE Transactions on Dependable and Secure Computing*, Vol. 4, No. 1.

240. P. Sandborn, R. Jung, R. Wong, and J. Becker (Apr. 2007) "A Taxonomy and Evaluation Criteria for DMSMS Tools, Databases, and Services", *Proceedings of the 2007 Ageing Aircraft Conference*, CA.

241. J. Torresen and T. Lovland (11-13 Apr. 2007) "Parts Obsolescence Challenges for the Electronics Industry". *IEEE Conference on Design and Diagnostics of Electronic Circuits and Systems*, pp. 1-4.

242. R. Rendon (16-18 Apr. 2007) "Using a Modular Open Systems Approach in Defense Acquisitions: Implications for the Contracting Process", *IEEE International Conference on System of Systems Engineering (SoSE)*, pp. 1-6.

243. M. Lyu (May 2007) "*Software Reliability Engineering: A Roadmap*", in "*Future of Software Engineering*, FOSE '07, ISBN 0-7695-2829-5, pp. 153-170.

244. S. Wu and D. Clements-Croome (2007) "Burn-in policies for products having dormant states", *Reliability Engineering and System Safety*, Vol. 92, pp. 278-285.

245. C. Wu, C. Chou, C. Huang (2007) "Optimal burn-in time and warranty length under fully renewing combination free replacement and pro-rata warranty", *Reliability Engineering and System Safety*, Vol. 92, pp. 914-920.

246. P. Söderholm (2007) "A system view of the No Fault Found (NFF) phenomenon", *Reliability Engineering and System Safety*, Vol. 92, pp. 1-14.

247. D. Schroder (2007) "Negative bias temperature instability: What do we understand?", *Microelectronics Reliability*, No. 47, pp. 841-852.

248. W. Robinson, M. Alles, T. Bapty, B. Bhuva, J. Black, A. Bonds, L. Massengill, S. Neema, R. Schrimpf, and J. Scott (May 30-1 Jun. 2007) "Soft Error Considerations for Multicore Microprocessor Design", *IEEE International Conference on Integrated Circuit Design and Technology*, pp. 1-4.

249. UK Ministry of Defence (1 Jun. 2007) "*Safety Management Requirements for Defence Systems*", Defence Standard 00-56, Issue 4.

250. W. Wang, Z. Wei, S. Yang, and Y. Cao (2007) "An Efficient Method to Identify Critical Gates under Circuit Aging", *Proceedings of the 2007 IEEE/ACM International Conference on Computer-aided Design*, pp. 735-740.

251. I. Baylakoğlu (14-16 Jun. 2007) "Reliability Concerns of Lead-free Solder Use in Aerospace Applications", *3rd International Conference on Recent Advanced in Space Technologies*, pp. 158-164.

252. The Value Society (Jun. 2007) "*Value Standard and Body of Knowledge*".

253. G. Ribes, M. Rafik, and D. Roy (2007) "Reliability issues for nano-scale CMOS dielectrics", *Microelectronic Engineering*, Vol. 84, pp. 1910-1916.

254. T. Gylfason and G. Zoega (2007) "A golden rule of depreciation", *Economics Letters*, Vol. 96, pp. 357-362.

255. International Council on Systems Engineering (2007) "*INCOSE Systems Engineering Handbook v3.1*".

256. International Roadmap Committee (2007) "*International Technology Roadmap for Semiconductors – 2007 Edition*".

257. S. Borkar, N. Jouppi, and P. Stenstrom (2007) "Microprocessors in the Era of Terascale Integration", *Design, Automation, and Test in Europe*, pp. 237-242.

258. M. Franz (Jul.-Aug. 2007) "Containing the Ultimate Trojan Horse", *IEEE Security and Privacy*, Vol. 5, Issue 4, pp. 52-56.

259. R. Plastow (Aug. 2007) "*Filling the Assurance Gap on Complex Electronics*", NASA/CR-2007-214939.

260. T. Henzinger and J. Sifakis (Oct. 2007) "The Discipline of Embedded Systems Design", *IEEE Computer*, Vol. 40, Issue 10, pp. 32-40.

261. M. Glinz (15-19 Oct. 2007) "On Non-Functional Requirements", *15th IEEE International Requirements Engineering Conference*, pp. 21-26.

262. J. Littlefield-Lawwill and R. Viswanathan (21-25 Oct. 2007)"Advancing Open Standards in Integrated Modular Avionics: An Industry Analysis", *26th IEEE/AIAA Digital Avionics Systems Conference*, pp. 2.B.1-1 to 2.B.1-14.

263. PEO-IWS 7 (25 Oct. 2007) "*Naval Open Architecture Contract Guidebook*", Department of the Navy, Version 1.1.

264. Y. Freeman, R. Hahn, P. Lessner, and J. Prymak (Oct-Nov. 2007) "Reliability and Critical Applications of Tantalum Capacitors". *Proceedings Capacitor and Resistor Technology Symposium (CARTS)*, Electronics Components, Assemblies, and Materials Association, pp. 193-204.

265. A. González, S. Mahlke, S. Mukherjee, R. Sendag, D. Chiou, and J. Yi (Nov-Dec. 2007) "Reliability: Fallacy or Reality?". *IEEE Micro*, Vol. 27, Issue 6, pp. 36-45.

266. P. Sandborn (Dec 2007)"Software Obsolescence – Complicating the Part and Technology Obsolescence Management Problem", Editorial in *IEEE Transactions on Components and Packaging Technologies*, Vol. 30, No. 4.

267. P. Reviriego, J. Maestro, and C.Cervantes (Dec. 2007) "Reliability Analysis of Memories Suffering Multiple Bit Upsets", *IEEE Transactions on Device and Materials Reliability*, Vol. 7, No. 4.

268. National Aeronautics and Space Administration (Dec. 2007) "*NASA System Engineering Handbook*", NASA/SP-2007-6105 Rev 1.

269. C. Constantinescu (28-31 Jan. 2008) "Intermittent Faults and Effects on Reliability of Integrated Circuits", *Reliability and Maintainability Symposium*, pp. 370-374.

270. L. Kharb and R. Singh (Mar. 2008) "Complexity Metrics for Component-Oriented Software Systems", *ACM SIGSOFT Software Engineering Notes*, Vol. 33, Issue 2, Article 4.

271. J. Bøegh (Mar.-Apr. 2008) "A New Standard for Quality Requirements", *IEEE Software*, Vol. 25, Issue 2, pp. 57-63.

272. K. Uzunov and T. Nguyen (Apr. 2008) "*Dependability of Software in Airborne Mission Systems*", DSTO-TR-2111.

273. H. Ma, Y. Zhang, Z. Cai, J. Suhling, P. Lall, and M. Bozack (20-23 Apr. 2008) "Aging Induced Evolution of Free Solder Material Behavior", *International Conference on Thermal, Mechanical, and Multi-Physics Simulation and Experiments in Microelectronics and Micro-Systems*, pp. 1-12.

274. H. Qi, S. Ganesan, and M. Pecht (29 Apr. 2008) "No-fault-found and intermittent failures in electronic products", *Microelectronics Reliability* Vol. 48, pp. 663-674.

275. K. Liao and C. Chang (29 Apr. 2008) "Applications of damage models to durability investigations for electronic connectors", *Materials and Design*, Vol. 30, pp. 194-199.

276. P. Grogono and B. Shearing (12-13 May 2008) "Concurrent Software Engineering: Preparing for Paradigm Shift", *Proceedings of the 2008 C3S2E Conference* Vol. 290, pp. 99-108.

277. Aeronautical Radio Incorporated (9 Jun. 2008) "*Guidelines for the Reduction of No Fault Found (NFF)*", ARINC Report 672.

278. T. Mattson and M. Wrinn (8-13 Jun. 2008) "Parallel Programming: Can we PLEASE get it right this time?", *45th ACM/IEEE Design Automation Conference*, pp. 7-11.

279. E. Kesseler (2008) "Assessing COTS software in a certifiable safety-critical domain", *Journal of Information Systems*, Vol. 18, pp. 299-324.

280. International Organisation for Standardisation (2008) "*Systems and Software Engineering – System Life Cycle Processes*", ISO/IEC 15288:2008.

281. A. Utaka (2008) "Pricing strategy, quality signalling, and entry deterrence", *International Journal of the Industrial Organisation*, Vol. 26, pp. 878-888.

282. W. Wolf (2008) "*Computers as Components: Principles of Embedded Computing System Design*", 2nd Edition, Elsevier, ISBN 978-0-12-374397-8, 544pp.

283. B. Tekinerdogan, H. Sozer, and M. Aksit (2008) "Software architecture reliability analysis using failure scenarios", *The Journal of Systems and Software*, Vol. 81, pp. 558-575.

284. M. White and Y. Chen (2008) "*Scaled CMOS Technology Reliability User Guide*", Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California. JPL Publication 08-14 3/08.

285. D. Reinhardt (2008) "Considerations in the Preference for and Application of RTCA/DO-178B in the Australian Military Avionics Context", *13th Australian Conference on Safety-Related Programmable Systems* (SCS 2008), pp. 49-68, T. Cant (Ed), Australian Computer Society.

286. M. Marshall and K. Lambert (Jul. 2008) "Insights into Supporting Complex Systems under Conditions of Obsolescence", *Portland International Conference on Management of Engineering and Technology*, ISBN 978-1-890842-17-5, pp. 1918-1923.

287. L. Hoffman (Jul 2008) "In Search of dependable Design", *Communications of the Association for Computing Machinery*, Vol. 51, Issue 7, pp. 14-16.

288. Lockheed Martin Corporation (Jul. 2008) "*Optimal Configuration and Deployment of software on Multi-core Processing Architectures*", Air Force Research Laboratory. AFRL-RI-RS-TR-2008-195.

289. A. Behbahani, N. Gibson, M. Rangarajan, and D. Benson (Jul. 2008) "*Multi-core Processors: An Enabling Technology for Embedded Distributed Model-based Control (Postprint)*", Air Force Research Laboratory, AFRL-RZ-WP-TP-2008-2183.

290. J. McElroy, Sr., R. Pfahl, Jr. (28-31 Jul. 2008) "Environmentally Friendly Electronics for High Reliability", *International Conference on Electronic Packaging Technology and High Density Packaging*, pp. 1-3.

291. T. Henzinger (31 Jul. 2008) "Two challenges in embedded systems design: predictability and robustness", *Philosophical Transactions of the Royal Society A*, Vol. 366, pp. 3727-3736.

292. A. Jokerst, J. Martin, K. Rodgers, K. Roland, and E. Tesla (Aug. 2008) "*US Reliance on Foreign IT: Mitigating Risks Associated with Foreign Sources of Hardware Components*", USSTRATCOM Global Innovation and Strategy Center, Project 08-03.

293. S. Misra and I. Akman (Sep. 2008) "*A Model for Measuring Cognitive Complexity of Software*", pp. 879-886 of "*Knowledge-Based Intelligent Information and Engineering Systems*", ISBN 978-3-540-85564-4.

294. M. Pecht and D. Humphrey (Sep. 2008) "Addressing Obsolescence – The Uprating Option", *IEEE Transactions on Components and Packaging Technologies*, Vol. 31, Issue 3, pp. 741-745.

295. W. Wang, V. Reddy, B. Yang, V. Balakrishnan, S. Krishnan, and Y. Cao (21-24 Sep. 2008) "Statistical Prediction of Circuit Aging under Process Variations", *IEEE Custom Integrated Circuits Conference*, pp. 13-16.

296. G. Bartley and B. Lingberg (26-30 Oct. 2008) "Certification Concerns of Integrated Modular Avionics (IMA) Systems", *27th IEEE/AIAA Digital Avionics Systems Conference*, pp. 1.E.1-1 to 1.E.1-12.

297. Renesas Technology (28 Nov. 2008) "*Semiconductor Reliability Handbook*", REJ27L0001-0101, Rev. 1.01.

298. R. Williams (Dec. 2008) "How We Found the Missing Memristor", *IEEE Spectrum*, Vol. 45, Issue 12, pp. 28-35.

299. M. Anderson, C. North, and K. Yiu (Dec. 2008) "*Towards Countering the Rise of the Silicon Trojan*", DSTO-TR-2220.

300. International Roadmap Committee (5 Jan. 2009) "*International Technology Roadmap for Semiconductors, 2008 Update: Overview*".

301. Toshiba Corporation Semiconductor Company (Jan. 2009) "*Semiconductor Reliability Handbook (2009-01)*".

302. International Electronics Manufacturing Initiative (iNEMI) (Jan. 2009) "*INEMI 2009 Roadmap*", Virginia USA.

303. M. Krasich (26-29 Jan. 2009) "How to Estimate and Use MTTF/MTBF. Would the Real MTBF Please Stand Up", *Annual Reliability and Maintainability Symposium*, pp. 353-359.

304. M. Kumar (Jan.-Feb. 2009)"*Memristor – Why Do We Have to Know About It?*", Editorial, *IETE Technical Review*, Vol. 26, Issue 1, pp. 3-6.

305. Semiconductor Industry Association (2 Feb. 2009) "*Global Semiconductor Sales Fell by 2.8 Percent in 2008*", Press Release.

306. I. Bate and P. Conmy (3-5 Feb. 2009) "Certification of FPGAs – Current Issues and Possible Solutions", *Proceedings of the Seventeenth Safety-Critical Systems Symposium*, Springer London, ISBN 978-1-84882-348-8, pp. 149-165.

307. The Industrial College of the Armed Forces (Spring 2009) "*Final Report: Electronics Industry*", National Defense University, Fort McNair, Washington, DC.

308. United States Government Accountability Office (Mar. 2009) "*Defense Acquisitions: Assessments of Selected Weapons Programs*", GAO-09-326SP.

309. T. Katayama, T. Kishi, S.Hosoai, T. Nakajima, T. Yuasa, M. Sugaya, and T. Ugawa (17-20 Mar. 2009) "Project Report: Toward the Realization of Highly Reliable Embedded Systems", *IEEE International symposium on Object/Component/Service-Oriented Real-Time Distributed Computing*, pp. 105-111.

310. C. Roark and B. Kiczuk (18 Mar. 2009) "Application of Open System Standards to DoD Avionics", *Naval Engineers Journal* Vol. 106, Issue 6, pp. 65-69.

311. F. Romero Rojo, R. Roy, E. Shehab, and P.J. Wardle (1-2 Apr. 2009) "Obsolescence Challenges for Product-Service Systems in Aerospace and Defence Industry", *Proceedings of the 1st CIRP Industrial Product-Service Systems Conference*, Cranfield University, pp. 255-260.

312. R. Waser (5 Apr. 2009) "Resistive non-volatile memory devices", *Microelectronic Engineering*, No. 86, pp. 1925-1928.

313. M. Abramovici and P. Bradley (13-15 Apr. 2009) "Integrated Circuit Security – New Threats and Solutions", *Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence*, Article No. 55.

314. D. Jackson (Apr. 2009) "A Direct Path to Dependable Software", *Communications of the Association for Computing Machinery*, Vol. 52, Issue 4, pp.78-88.

315. A. Concho, J. Ramirez-Marquez, and T. Herald (20-23 Apr. 2009) "Functionally Equivalent COTS for Optimal Component Substitution within System Evolution Planning", *7th Annual Conference on Systems Engineering Research*.

316. A. Bansal, R. Rao, J. Kim, S. Zafar, and J. Stathis (26-30 Apr. 2009) "Impact of NBTI and PBTI in SRAM Bit-cells: Relative Sensitivities and Guidelines for Application-Specific Target Stability/Performance", *IEEE International Reliability Physics Symposium*, pp. 745-749.

317. J. Clavareau and P.-E. Labeau (2009) "Maintenance and replacement policies under technological obsolescence", *Reliability Engineering and System Safety*, Vol. 94, pp. 370-381.

318. J. Jirsa and K. Dušek (13-17 May 2009) "Risk Management of Technology of Lead Free Soldering", *32nd International Spring Seminar on Electronics Technology*, pp. 1-4.

319. M. Levy and T. Conte (May-Jun. 2009). "Embedded Multicore Processors and Systems", *IEEE Micro*, Vol. 29, Issue 3, pp. 7-9,

320. J. Holt, A. Agarwal, S. Brehmer, M. Domeika, P. Griffin, and F. Schirrmeister (19 Jun. 2009) "Software Standards for the Multicore Era", *IEEE Micro*, Vol. 29, Issue 3, pp. 40-51.

321. Y. Shiyanovskii, F. Wolff, C. Papachristou, D. Weyer, and W. Clay (20 Jun. 2009) "Hardware Trojan by Hot Carrier Injection", *Computing Research Repository* abs/0906.3832.

322. Y. Shiyanovskii, F. Wolff, C. Papachristou, D. Weyer, and W. Clay (20 Jun. 2009) "Exploiting Semiconductor Properties for Hardware Trojans", *Computing Research Repository* abs/0906.3834.

323. T. Tsiakis and P. Katsaros (18-23 Jun 2009) "Hands on Dependability Economics", *Second International conference on Dependability*, pp. 117-121,

324. S. Wu, X. Li, X. Xing, P. Hu, Y. Yu and S. Li (24 Jun 2009) "Resistive dependence of magnetic properties in non-volatile Ti/Mn:TiO_2/ $SrTi_{0.993}Nb_{0.007}O_3$/Ti memory device", *Applied Physics Letters* Vol. 94, 253504 (3pp).

325. R. Kuehl (2009) "Stability of thin film resistors – Prediction and differences base on time-dependent Arrhenius law", *Microelectronics Reliability*, Vol. 49, pp. 51-58.

326. M. Pangburn and S. Sundaresan (2009) "Capacity decisions for high-tech products with obsolescence", *European Journal of Operational Research*, Vol. 197, pp. 102-111.

327. E. Nogueira, M. Vázquez, and N. Núñez (2009) "Evaluation of AlGaInP LEDs reliability based on accelerated tests", *Microelectronics Reliability*, Vol. 49, pp. 1240-1243.

328. H. Charles (2009) "*Advanced Wire Bonding Technology: Materials*, *Methods and Testing*", Chapter 4, pp. 113-179, of "*Materials for Advanced Packaging*", D. Lu and C. Wong, Editors, 724pp, ISBN 978-0-387-78218-8, Springer Science and Business Media.

329. N. Lee (2009) "*Lead-Free Soldering*", Chapter 5, pp. 181-218, of "*Materials for Advanced Packaging*", D. Lu and C. Wong, Editors, 724pp, ISBN 978-0-387-78218-8, Springer Science and Business Media.

330. L. Chung and J. do Prado Leite (6 Jul. 2009) "*On Non-Functional Requirements in Software Engineering*", in Lecture Notes in Computer Science, Vol. 5600/2009, pp. 363-379.

331. C. Shih, C. Wu., C. Lin, P. Hsiung, N. Hsueh, C. Chang, C. Koong, and W. Chu (20-24 Jul. 2009) "A Model-Driven Multicore Software Development Environment for Embedded System", *33rd Annual IEEE International Computer Software and Applications Conference*, Vol. 2, pp. 261-268.

332. Y. Jin, N. Kupp, and Y. Makris (27 Jul. 2009) "Experiences in Hardware Trojan Design and Implementation", *IEEE International Workshop on Hardware-Oriented Security and Trust*, pp. 50-57.

333. M. Hamill and K. Goševa-Popstojanova (Jul./Aug. 2009) "Common Trends in Software Fault and Failure Data", *IEEE Transactions on Software Engineering*, Vol. 35, No. 4, pp. 484-496.

334. G. Despotou, M. Bennett, and T. Kelly (3-7 Aug. 2009) "Supporting Through Life Safety Assurance of COTS Based Upgrades", *27th International System Safety Conference*, 10pp.

335. H. Choi, H. Jung, J. Lee, J. Yoon, J. Park, D. Seong, W. Lee, M. Hasan, G. Jung, and H. Hwang (4 Aug 2009) "An electrically modifiable synapse array of resistive switching memory", *Nanotechnology* Vol. 20, 345201 (5pp).

336. M. Chentir, J. Jullien, B. Valtchanov, E. Bouyssou, L. Ventura, and C. Anceau (Aug. 2009) "Percolation theory applied to PZT thin films capacitors break down mechanisms", *Microelectronics Reliability*, Vol. 49, pp. 1074-1078.

337. C. Moore and P. Conway (29 Aug. 2009) "*General-Purpose Multi-core Processors*", Chapter 6 of "*Multicore Processors and Systems*", S. Keckler, K. Olukotun, and P. Hofstee (Editors), ISBN 978-1-4419-026207, Springer, New York.

338. Q. Xia, W. Robinett, M. Cumbie, N. Banerjee, T. Cardinali, J. Yang, X. Li, W. Tong, D. Strukov, G. Snider, G. Medeiros-Ribeiro, and R. Williams (1 Sep. 2009) "Memristor – CMOS Hybrid Integrated Circuits for Reconfigurable Logic", *Nano Letters*, Vol. 9, No. 10.

339. M. Stork, J. Hrusak, and D. Mayer (9-10 Sep. 2009) "Memristor Based Feedback Systems", *Applied Electronics 2009*, pp. 237-240.

340. Defense Standardization Program Office (Sep. 2009) "*Diminishing Manufacturing Sources and Material Shortages – A Guidebook of Best Practices and Tools for Implementing a DMSMS Management Program*", SD-22.

341. S. Fürst, J. Mössinger, S. Bunzel, T. Weber, F. Kirschke-Biller, P. Heitkämper, G. Kinkelin, K. Nishikawa, and K. Lange (7-8 Oct. 2009) "AUTOSAR – A Worldwide Standard is on the Road", *14th International VDI Conference: Electronic Systems for Motor Vehicles*.

342. C. Haddon-Cave QC (28 Oct. 2009) "*The Nimrod Review: An independent review into the broader issues surrounding the loss of the RAF Nimrod MR2 Aircraft XV230 in Afghanistan in 2006*", HC1025. ISBN 978-0-102-96265-9. The Stationary Office, London.

343. J. Souyris, V. Wiels, D. Delmas, and H. Delseny (4 Nov. 2009) "*Formal Verification of Avionics Software Products*", in *Lecture Notes in Computer Science*, Vol. 5850/2009, pp. 532-546, ISBN 978-3-642-05088-6.

344. R. Chakraborty, S. Narasimhan, and S. Bhunia (4-6 Nov. 2009) "Hardware Trojan: Threats and Emerging Solutions", *IEEE International High Level Design Validation and Test Workshop*, pp. 166-171.

345. H. Kim and R. Bond (Nov. 2009) "Multicore Software Technologies: A survey", *IEEE Signal Processing Magazine*, Vol. 26, Issue 6, pp. 80-89.

346. J. Rushby (23-27 Nov. 2009) "Software Verification and System Assurance", *7th IEEE International Conference on Software Engineering and Formal Methods*.

347. Y. Li, Y. Kim, E. Mintarno, D. Gardner, and S. Mitra (Nov.-Dec. 2009) "Overcoming Early-Life Failure and Aging for Robust Systems", *IEEE Design and Test of Computers*, Vol. 26, Issue 6, pp. 28-39.

348. J. McHale (Jan. 2010) "U.S. Navy Avionics Systems embrace open architectures", *Military and Aerospace Electronics*, Special Report, Vol. 21, Issue 1.

349. M. Tehranipoor and F. Koushanfar (Jan.-Feb. 2010) "A Survey of Hardware Trojan Taxonomy and Detection", *IEEE design and Test of Computers*. Vol. 27, Issue 1, pp. 10-25.

350. G. Karalias (Jan/Feb. 2010) "Obsolete semiconductors: A proactive approach to End-of-Life", *Military Embedded Systems*, Vol. 6, No. 1, pp. 34-37.

351. J. Abella and X. Vera (Feb. 2010) "Electromigration for Microarchitects", *Association of Computing Machinery Computing Surveys*, Vol. 42, Issue 2, Article 9, 18pp.

352. O. Khan (Feb. 2010) "*A Hardware/Software Co-Design Architecture for Thermal, Power, and Reliability Management in Chip Multiprocessors*", Ph.D. dissertation, Department of Electrical and Computer Engineering, Graduate School of the University of Massachusetts Amherst.

353. Defence Materiel Organisation and Capability Development Group (26 Feb. 2010) "*Defence Capability Plan 2009*", Commonwealth of Australia, ISBN 978-0-642-29704-4, published Jul. 2009, Updated 26 Feb. 2010.

354. A. Utaka (5 Mar. 2010) "The Timing of Upgrades", *Japanese Economic Review*, 10pp.

355. J. Robles (Mar. 2010) "The Systems Engineering Relationship between Qualification, Environmental Stress Screening and Reliability", *Society of Automotive Engineers (SAE) International Journal of Aerospace*, Vol. 2, No. 1, pp. 268-274.

356. A. Kornecki and J. Zalewski (14 Apr. 2010) "Hardware certification for real-time safety-critical systems: State of the art", *Annual Reviews in Control*, Vol. 34, pp. 163-174.

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION
DOCUMENT CONTROL DATA

1. PRIVACY MARKING / CAVEAT (OF DOCUMENT)

2. TITLE

A Survey of Electronics Obsolescence and Reliability

3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION)

Document	(U)
Title	(U)
Abstract	(U)

4. AUTHOR(S)

R. J. O'Dowd

5. CORPORATE AUTHOR

DSTO Defence Science and Technology Organisation
506 Lorimer St
Fishermans Bend Victoria 3207 Australia

6a. DSTO NUMBER	6b. AR NUMBER	6c. TYPE OF REPORT	7. DOCUMENT DATE
DSTO-TR-2437	AR-014-805	Technical Report	July 2010

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. NO. OF PAGES	12. NO. OF REFERENCES
2010 / 1084887	07 / 245	CAOD	119	356

13. URL on the World Wide Web

http: / / www.dsto.defence.gov.au / corporate / reports / DSTO-TR-2437.pdf

14. RELEASE AUTHORITY

Chief, Air Operations Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE, PO BOX 1500, EDINBURGH, SA 5111

16. DELIBERATE ANNOUNCEMENT

No Limitations

17. CITATION IN OTHER DOCUMENTS Yes

18. DSTO RESEARCH LIBRARY THESAURUS http: / / web-vic.dsto.defence.gov.au / resources / library / resources / Thesaurus / output / index.htm

Electronics; Systems; Obsolescence; Reliability

19. ABSTRACT

The service life of military assets significantly exceeds design life of commercial electronic systems used within them. Electronic obsolescence is increasingly associated with physical characteristics that reduce component and system reliability, both in usage and storage, with few design margins outside commercial warranty periods. Software content, however, remains a dominant limiting factor for reliability of electronic systems, and emerging commercial trends compound this. Traditional approaches to manage and sustain electronic systems are therefore increasingly ineffective and costly. This report surveys the interrelated concerns of obsolescence and reliability of electronic systems, and describes emerging responses to these concerns.