ARIZONA STATE UNIVERSITY

Spacecraft Integration and Test Strategies

CSE Capstone

Ian Cleary

5/1/2017

This paper overviews hardware integration and test paradigms for spacecraft systems, with discussions of possible applications of software verification, validation and testing concepts from previous literature.

Table of Contents

Introduction

Spaceflight, aerospace, and military equipment goes through rigorous screening and testing before the start of their missions. While especially true for spaceflight hardware, all three types have prohibitively high repair costs once fielded. The rigorous screening and testing is meant to detect defects in the systems before their mission. Undetected defects can appear at the beginning of life (often called latent failures) due to workmanship issues. A more damaging type of defect is undetected degradations such as suspect solder joints and partial damage due to Electro-Static Discharge (ESD) events. The later can occur well into a system's mission life and are virtually impossible to prevent with absolute certainty. Great care and thought has been put into the hardware testing and screening process, with the goal of detecting and preventing both types of failure with the maximum probability within the programs financial and temporal budgets.

Software verification, validation and testing concepts and best practices are continually evolving. There is a constant search to document, disseminate, and adopt the best practices in the software industry. The following software concepts will be discussed: Scenario Testing, Regression Testing, System Test Prioritization, and Defect Tracking.

While these software concepts apply to both software and spacecraft hardware systems, the financial and temporal cost for their respective implementation are, on average, subject to vastly different assumptions. If regression testing in a SW system detects a bug, agile development and proper version control could allow the roll back to the last stable build. Information Technology (IT) automation has matured to the point where this time required, reverting to a stable build, could be on the order of seconds, minutes, or hours (depending on scale). Spacecraft integration and testing activities are not subject to that convenience. Often revision of an assembly are available but rework and repair events occur on a time scale on the order of hours, days, and weeks depending on severity and location of the hardware (fielded or in assembly/test). Additionally, the deployment or build of new revisions for hardware is subject to procurement, manufacturing, and shipping lead times. It is often a reasonable assumption to have a 4-8 lead time on Printed Wiring Assemblies (PWAs) or machined housings. While those are often custom designs, they dwarf in comparison to the lead time required for custom integrated circuits from foundries (especially if you do not have priority relative to the total list of customers). Wafer runs for Silicon (Si), Gallium Arsenide (GaAs), and Gallium Nitride (GaN) wafers often range from 6-20 weeks. The manufacturing process may only take 6-12 weeks, but the overhead of their queue, procurement, and shipping activities leads to the longer total time. The above lead times account for just the procurement, incoming inspection and vendor testing of those components. There has been no mention of functional testing at different assembly levels. The concept of unit testing is shared between software and hardware systems, but the assumptions stated above dramatically elongate the time required to fix a bug or defect in a spacecraft hardware system relative to a software system with the similarly well-defined and developed unit tests at every level of assembly.

The relative weight between hardware and software transitions from or to a build revision is not the subject of this paper, but rather how the software methodologies can be applied to the spacecraft design paradigms. Consideration will be taken to discuss the difference of assumptions between design types and how the proposed applications could be modified, from their software roots, to more directly map spacecraft integration and test paradigms.

Spaceflight Hardware Design Paradigms:

Overview of Spacecraft Electronics Systems Design/Development/Test Lifecycle

Spacecraft Electronic Systems are comprised of different types of assemblies that function together to accomplish the system level requirements. Due to the complexity of most spacecraft systems, the top level requirements are flowed down to each level of assembly. Every level of the system goes through several processes before integration into the higher level to ensure a successful mission.

These processes ensure that each assembly meets its functional requirements, is screened to precipitate workmanship defects, and will survive the environments induced while during launch, orbit-raising, and normal mission operations.

Verification

A system's requirement management plan defines how every assembly documents and verifies requirements. Requirements that must be verified contain the word "shall". There are two main parameters that define how and when a requirement is verified: method and phase. A requirement's verification method defines how it is verified.

Verification Methods

A set of methods could be the following: Inspection, Analysis, Demonstration, and Test.

- **Inspection** is used when a physical parameter or observation is verified again the specification. An example is an assemblies mass or number of output connectors being verified against an assembly drawing.
- **Analysis** is used when some form of simulation or data analysis is performed. This generally occurs before the Critical Design Review (CDR). However, Analysis can be coupled with test when there is post process of data across a flight population
- **Demonstration** is used when there is qualitative verification of a functional requirement. An example is a scenario that the system turns on after exposure to certain conditions. There is no quantitative definition for the turn on event, but it is observable and can be used to verify compliance to the requirement.
- **Test** is used when there is a there is quantitative verification of a requirement. There should be bounded test limits that have a single digit greater precision in the decimal place than the measurement precision.

In addition to above set, several more methods can be used to handle verification across levels of assembly. These are review of lower level data and screen of lower level data. Both inherit the verification method and phase of the lower level specification to which they are related.

- **Review of lower level data** handles when a higher level specifies that the population of lower level assemblies it inherits upon integration meet their requirements. This is not always required, but can help to show full traceability across levels of integration.
- **Screening of lower level data** is used when a specific set of lower level assemblies is used for multiple higher level assemblies. An example would be if 10 modules are used in a communications transponder, however two groups are used, while each draw from the same part. You could take the risk of only building one variation and screening parts to meet the other's specification.

Verification Phases

A set of phases could be the following: *DVT, QT, and AT.*

- **DVT** stands for Design Verification Test. This phase is for preliminary designs before the flight build. A requirement that is verified during the DVT phase can be bought off by any preliminary design iteration, as long as there is no change to the portion of the design from that point onward. The judgment of what is a significant enough change to determine re-validation should be made by systems engineering, with input from all relevant stakeholders.
- **QT** stands for Qualification Test. This phase occurs in conjunction with the flight build. In its truest form, QT is a sampling of serial numbers from the flight production build, with testing performed to higher levels to ensure that the flight build will meets its requirements after exposure to higher mechanical and thermal environments. Since the verification to these higher (qualification) levels induces wear and stress to the assembly (that the flight assemblies will not see), qualification assemblies are barred from flying. Depending on the system top level schedule and critical paths, the qualification effort either occurs A) in series in front of the flight build, or B) in parallel with the flight build. From purely the qualification of the flight build perspective, it is risky to perform the qualification effort in parallel with the flight build, but this often occurs in practice as schedule delays often come at a high enough cost that the goal is to have the qualification effort as close to parallel as practical.
- **AT** stands for Acceptance Testing. This is the final phase an assembly goes through before integration into a higher level assembly or delivery to the customer. Unlike DVT and QT, acceptance testing occurs for every serial number. DVT and QT are design verification events and occur per design, as opposed to per individual instance of the design. Acceptance Testing exercises the design to verify the functional requirements while exposing the design to only the environments that the design will see in its lifetime. This differs from the more extreme qualification environments exposed to the qualification unit. At the end of acceptance testing, there is a compilation of all the data (test data, pictures along the process, technician logs, mate/de-mate logs for connectors, etc.) and then a formal review with systems engineering, quality assurance, manufacturing, and test engineering to formally sing off on the unit. This is a rigorous process for spaceflight hardware and requires thorough procedures.

In addition to above set, the phases below are commonly used compromises between the above phases to mitigate risk. The risk can be from schedule pressure, technical concerns from new portions of the design, un-proved hardware, new vendors/manufacturing processes, and others.

- **Pre-Qual** is an informal risk reduction activity that is performed on the engineering prototypes before the flight build. While Qualification Testing and Acceptance Testing are rigid formal test flows for record, Pre-Qual is a sub-set of the qualification test plan with the single purpose of performing as risky test events as soon in the development process as possible. This reduces the cost of change after a failure is discovered, for the following reasons:
    - The failed assembly is not part of the flight build and does not need to be formally dispositioned to get at root cause of the failure

    - There is likely at least one design iteration before the flight build and the root cause can be corrected or its probability of failure can be reduced.
- **Proto-Qual** and **Proto-Flight** are two modifications/compromises between Qualification hardware and Flight Acceptance hardware. Both Proto-Qual and Proto-Flight are intended to be flown. Several reasons why to choose Proto-Qual or Proto-Flight are listed below:
    - Quantity of the build
    - Schedule Concerns
    - Cost of the assembly versus budget

Summary

The above verification methods and phases define the language in which system's engineering defines and maps requirements to an assembly. In the case of a system with many levels of integration, the above methodology applies to all levels of the system. There is an inheritance of tested and approved sub-assemblies inter higher level assemblies until the full system is assembled. The cost of change dramatically increases as the system is integrated. If a failure is detected at the higher level, its root cause may dictate that a significant disassembly, rework and repair, re-test, and reassembly effort. The same is true for failures in the field. If a terrestrial product fails, the cost is still high to replace/repair a fielded unit, but that cost is dwarfed, in relative magnitude, to the cost to launch a replacement spaceflight system. In order to mitigate that significant cost, there is a vast body of work put into workmanship screening (a type of Environmental Stress Screening) for electronic spaceflight system.

Environmental Stress Screening

Introduction/Definition/Background

Environmental Stress Screening (ESS) is a means of minimizing failures due to defective parts and poor workmanship as early in the production process as possible. *"Environmental stress screening of a product is a process which involves the application of one or more specific types of environment stresses for the purpose of precipitating to hard failure, latent, intermittent, or incipient defects or flaws which would cause product failure in the use environment. The stress may be applied in combination or in sequence on an accelerated basis but within product design capabilities"* (Mosemann, Willoughby, Burdt 3). ESS has many benefits to a program; the table below outlines ESS' benefits to several customer groups.

	Management Target	Benefits/Rationale
Engineering & Manufacturing Manager	Program Manager and Engineering Manager	• Ensures Hardware Performance • Contributes to Parts List Development • Improves Reliability Growth Testing • Assures Readiness of Production Screens • ESS Weeds Out Problem Vendors • 60% of Failures are due to Workmanship Defects • 30% of Failures are due to part flaws • ESS Design is by nature iterative
Production	Manufacturing Manager	• Reduces Rework Cost • Minimizes Schedule Delays • Facilitates Achievement of Design Reliability in Production Hardware • Improves Productivity • Reduces or eliminates workmanship and part defects • Entire production process more efficient • Defect surfaced at lowest levels of assembly and root cause corrective action implemented • Acceptance test passed on first pass, less high assembly level rework
General		• Less program cost • Less schedule and variance impact • Higher field reliability • Failures are forced to emerge at convenient production steps • Less latent defects shipped to field
User		• Initially delivered hardware meets reliability and quality requirements • Lowers support cost • Lowers life cycle cost • Spares meet reliability requirements • On-time deliveries • Increases confidence and satisfaction • Removes defect normally present in delivered hardware • Fewer field failures or maintenance actions • ESS influences throughout acquisition cycle

Table: ESS Benefits to Management (Mosemann, Willoughby, Burdt 5).

While ESS is common today, its origins a result of shared lessons learned between the Department of Defense and their contractors during the early to mid-20th century. Most military product failures (in the field) were a result defective parts and improper workmanship. To combat this, functional testing of incoming parts was introduced in the 1950s (with the initial goal of reducing rework due to defective parts). After enough functional test failures were dispositioned as the result of infant mortality of parts, burn-in of higher level assemblies was introduced during the 1960s. However, time proved that burn-in and functional testing were not adequate in precipitating manufacturing defects. In terms of flaw precipitation, the two most successful forms of ESS are random vibration and temperature cycling. Other forms of ESS may prove effective for specific hardware and should be evaluated when an ESS procedure is developed (Mosemann, Willoughby, Burdt 1).

Levels of Assembly

Any large scale system requires several levels of assembly to allow the system design to be broken down into manageable pieces. There is a significant amount of effort and time required to integrate lower level assemblies into higher level assemblies. Some examples of the steps involved are inspections of workmanship and photos, engineering reviews to buy off the designs data packages, inventory and financial transactions to move hardware from one state to another. It is cost effective to perform ESS at the lowest level of assembly. Whether or not to perform ESS at higher levels should be a discussion for, but not limited to, the system engineering and quality assurance departments. A few generic reasons why the cost of finding a failure dramatically increases with the level of assembly are below:

- At higher levels
  - More Assembly work has to be undone and redone when failures occur
  - More material may need to be scrapped
  - More impact on production flow and schedule usually occurs
- At lower levels
  - Corrective action is quicker
  - Repair cost is lower

While ESS is generally performed at the lowest level, each step and level of assembly presents opportunities for the introduction of flaws and defect. “Obviously, ESS at a particular level cannot uncover flaws that are not introduced until the next level. *Generally, this dilemma is usually controlled by performing ESS at each major functioning level in the manufacturing process consistent with an assessment of the potential defect population at each level of assembly*. Resolution of these conflicting consideration usually involves screening at multiple (usually 2 or 3) levels of assembly. ESS at lower levels should focus on surfacing and correcting flaws in piece parts and PWA processing. Thus, most ESS failures at higher levels will reflect flaws introduced later in the manufacturing sequence that are usually correctable without tear-down of the PWA level” (Mosemann, Willoughby, Burdt 10)

Types of flaws to be precipitated

Standard ESS profiles should be tailored to a product’s responses to environmental stimuli in order to maximize the precipitation of flaws. The table below shows typical failure mechanics:

	Screening Environment		
	Thermal Cycling	**Vibration**	**Thermal or Vibration**
Type of Failure	Component Parameter Shift	Particale Contamination	Defective Solder Joints
	PWA Opens/Shorts	Chafed, Pinched Wires	Loose Hardware
	Component Incorrectly installed	Defective Crystals	Defective Components
	Wrong Component	Mixed Assemblies	Fasteners
	Hermetic Seal Failure	Adjacent Boards Rubbing	Broken Component
	Chemical Contamination	Two Components Shorting	Surface Mount Technology Flaws
	Defective Harness Termination	Loose Wire	Improperly Etched PWAs
	Improper Crimp	Poorly Bonded Component	
	Poor Bonding	Inadequately Secured Parts	
	Hairline Cracks in Parts	Mechanical Flaws	
	Out-of-Tolerance Parts	Improperly Seated Connectors	

Table: Screening Environment versus Typical Failure Mechanics

Types and Severity of Stresses

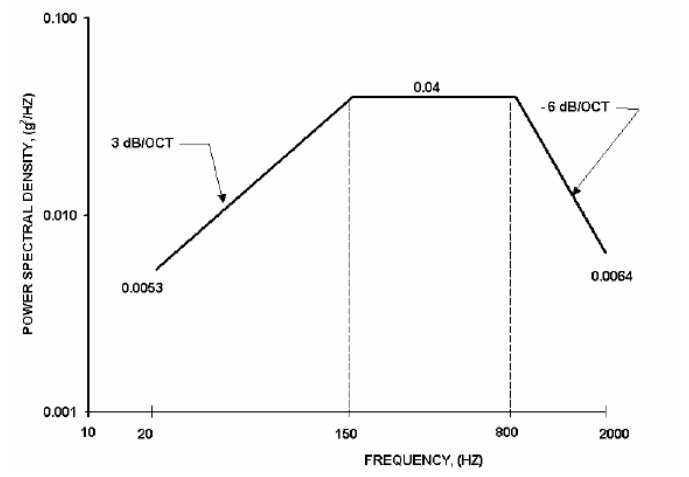
Of all the types of environmental stresses that have been used for ESS, random vibration and thermal cycling are currently considered the most cost effective. Table: Screening Environment versus Typical Failure Mechanics identifies common types of failures and whether random vibration or thermal cycling is the more likely stress to precipitate that particular failure. When designing a test flow, care must be considered since a failure may be surfaced under one stress, but detected under the other.

While traditional ESS consists of temperature cycling and random vibration, the type of hardware used in the system should be considered when deciding if any other environments should be considered. Some other types of stress screening are power cycling, which is effective at detecting certain types of latent defects; pressure cycling for sealed equipment, or equipment sensitive to rapid pressurization and depressurization events; and acoustic noise, which may excite microelectronic structures better than vibration born in the structure itself. In the end, the ESS environments used should be chosen based upon the types of flaws that are known or expected to exist.

Random Vibration

In the past, fixed-frequency or swept-sine vibrations were sometimes used. However, as test equipment improved and a better understanding of the pitfalls of fixed frequency and swept-sine vibration materialized, broadband random vibration became the predominant form used. True random vibration and quasi-random vibration are almost exclusively used today. “True random vibration implies excitation at all frequencies within a certain bandwidth (usually around 20 to 2000 Hz) and is neither cyclic nor repetitive. Quasi-random vibration, on the other hand, is a relatively new technology using pneumatically driven vibrators which generate repetitive pulses. For screening applications, several (usually 4 to 6) of these vibrators are attached to a specially designed shaker table which is allowed to vibrate in multiple axes simultaneously” (Mosemann, Willoughby, Burdt 10-11). This complex motion, (6 degrees of freedom vibration – 3 linear axis and 3 rotational axes, is very effective at finding most type of flaws. As indicated in Table: Screening Environment versus Typical Failure Mechanics, vibration normally precipitates latent assembly flaws caused by the undesired relative motion of parts, wires, solder joints, epoxy, PWAs, etc. as well as mechanical flaws that lead to propagating cracks and other failures.

Consideration should be taken on what frequency range and power spectral densities to be used based upon the type and size of the device. The figures below illustrate a recommended ESS workmanship Random Vibration profile for space vehicles and three scenarios when the device’s predicted environment differs from the previously stated workmanship profile.



Curve Values

Frequency (Hz)	Minimum PSD (g^2/Hz)
20	0.0053
20 to 150	+3 dB per octave slope
150 to 800	0.04
800 to 2000	-6 dB per octave slope
2000	0.00644

The overall acceleration level is 6.9 grms.

Note: This spectrum applies only to units whose mass does not exceed 23 kilograms (50 pounds).

MIL 1540E Figure for ESS Workmanship Random Vibration Profile

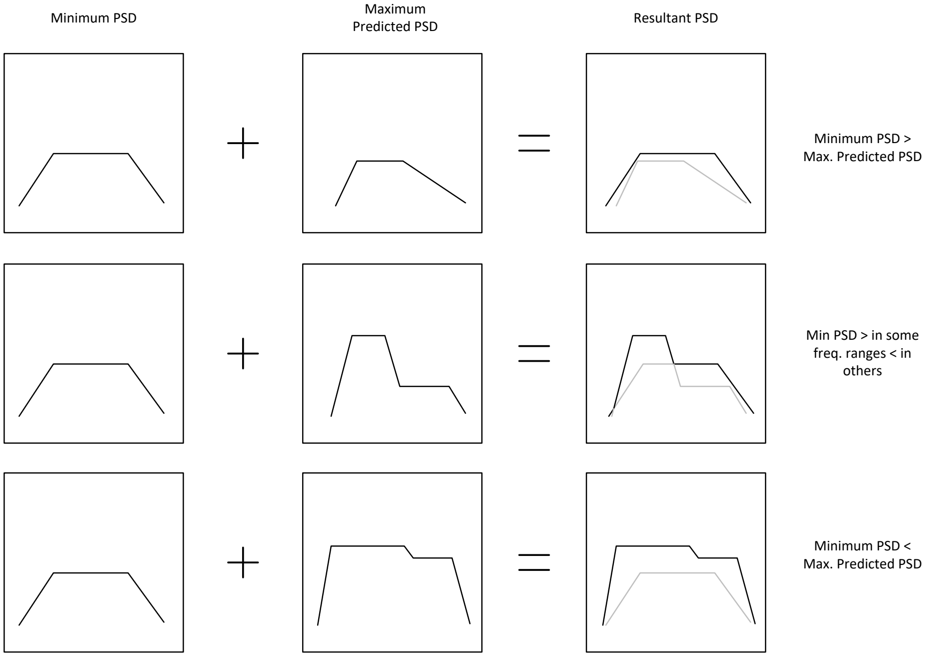

Minimum PSD
Maximum Predicted PSD
Resultant PSD
+
=
Minimum PSD > Max. Predicted PSD
Min PSD > in some freq. ranges < in others
Minimum PSD < Max. Predicted PSD

Figure: Combination of Minimum PSD from ESS Flow and Maximum Predicted PSD

Thermal Cycling

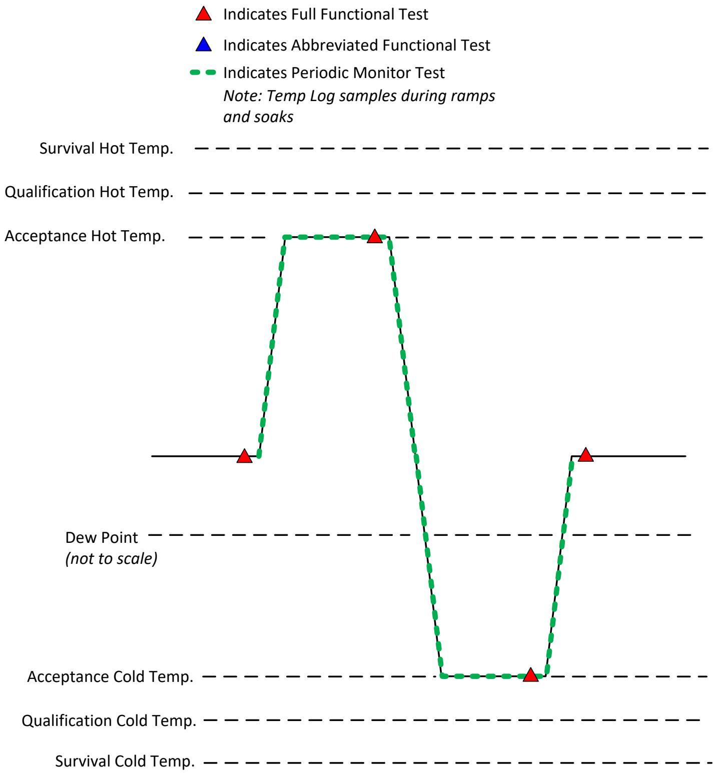
Effective thermal cycling usually requires large, rapid temperature changes. This type of thermal cycling is “used for the detection of assembly flaws that involve installation errors or inadequate chemical or mechanical isolation or bonding. Under rapid thermal cycling (e.g. in solder joints), differential thermal expansion takes place without sufficient time for stress relief, and this is a major mechanism for precipitating latent defects to detectable failures. (Mosemann, Willoughby, Burdt 11).

Burn-In for Performance Stabilization

The definition of Burn-in varies by company or agency. For the sake of this paper, burn-in shall be defined as the exposure of powered equipment to a steady elevated temperature. Burn-in is effectively valuable when parts have high power densities that are maintained for long periods of time. One class of example is power amplifiers in GaN or GaAs technologies. They are integrated circuits with a total area on the order of 25 mm^2 and possible power consumptions of 10-100W or more. The circuits contain Field Effect Transistors (FETs) and passive structures, such as Lange couplers, that have very dielectrically filled gaps between conductive metal, with separations on the order of nanometers to micrometers. Assuming proper design for normal operation, the gaps are still susceptible to damage if the conducting material degrades due to electro-migration effects or changes in the mechanical structure change the electrical interactions between interfaces cause overstresses in the materials. Burn-in is also commonly used to settle the mechanical properties of the GaAs and GaN die, which can materialize in a shift in electrical gain through the circuit. This is especially important if the functional requirements of the device are intolerant. While burn-in is “considered as a possible supplement to the ESS requirement”, it is recommended for assemblies containing the two listed integrated circuit technologies (Mosemann, Willoughby, Burdt 11).

Failure Detection: Thermal Cycling

Failure detection during thermal cycling consists of two approaches, which can be used in tandem. In the first, limited or full performance measurements are made before and after an ESS thermal cycles. The second approach calls for continuous monitoring during both the temperature ramps and the dwell portion of each cycle. Both types are illustrated in the figure below.



Failure Detection: Random Vibration

Failure detection for random vibration follows a similar philosophy to thermal cycling. Pre and post functional tests (full or abbreviated) can be performed on the device under test. Additionally, monitoring equipment can be used during vibration screens to detect a marginal component failure that may only show up during the application of vibration.

Difference between ESS and Environmental Stress Tests

A comment on the difference between ESS and EST (Environmental Stress Test): ESS is a set of processes that are meant to precipitate defects. They are not a set of tests in the common accept/reject sense. “Failures” during an ESS process are encouraged as they will bring flaws to light. EST is meant to indicate any test performed during an ESS procedure. For instance, during and ESS process, a product could “fail” a functional test performed at the product’s hot acceptance temperature. While the test “failed”, the ESS process has successfully identified a defect that can be used to correct the root cause or possibly modify the ESS program.

Qualification and Acceptance Test Planning

In addition to workmanship screening, qualification of spaceflight hardware and software occurs to ensure performance parameters are met, with adequate margin, after the system is exposed to launch and operating environments. The sections below describe a boilerplate for space-flight qualification hardware and flight hardware testing.

Official design qualification and flight acceptance testing generally occurs at the unit level of hardware builds. Testing is also performed at the module level to verify functionality and performance, however, this boilerplate document focuses on the unit-level testing related to qualification and acceptance.

Definitions

Item Levels

The categories of items in hierarchical order are defined in this section.

- **Module**. A module is a functional item that is viewed as a complete and separate entity for purposes of manufacturing, maintenance, or record keeping. Examples: Transmit/Receive Modules, RF Converter Modules, Power Supply modules, Digital Processing and Control modules.
- **Unit**. A unit is an assembly of functionally related modules. It consists of two or more modules and may include interconnection items such as cables or a backplane, and the supporting structure to which they are mounted. Example: Radio Front End Assemblies (Antennas, transmit receive assemblies, waveguide diplexers), main power supply units (combination of modules, fuse boxes, and power supply switching assemblies).
- **Component**. A component is a module, unit, or other assembly. This term is used frequently in this document to generically refer to a module, unit, or other assembly.
- **Payload**. A payload is a lower level assembly of a space vehicle that provides the space-based capability for the overall mission. Examples: optical imaging, science probes, communications.
- **Spacecraft Bus**: A space craft bus is the scaffolding that provides structural support, insulation, solar arrays for power, and generally telemetry and control to the Payload.
- **Space Vehicle**: The combination of the payload and the spacecraft bus. The space vehicle is the assembly that must fit inside the fairing on top of the launch vehicle.

Prototype Hardware

Prototype hardware (also referred to as Engineering Model (EM) hardware or Design Verification Test (DVT) hardware) is hardware that is used for development purposes. This hardware may be built to flight standards, but is not required. This hardware shall not be used for flight.

Qualification Hardware

Qualification (also known as "Qual") hardware is hardware that has been built using the same parts, materials, and processes as flight hardware, but will undergo environmental testing at qualification levels in order to demonstrate design margin. Qualification hardware shall not be used for flight.

Flight Hardware

Flight hardware is hardware that has been built using flight parts, materials, and processes, and tested to flight acceptance levels to screen for workmanship defects. This hardware is intended for use in the space-flight end product.

Protoqualification Hardware

Protoqualification (also known as "protoflight") hardware is hardware that is designed to qualification levels, built using flight parts, materials, and processes, and tested to protoqualification levels. This hardware is intended for flight use.

Service Life

The service life of an item starts at the completion of fabrication and continues through all acceptance testing, handling, storage, transportation, prelaunch testing, all phases of launch, orbital operations, disposal, reentry or recovery from orbit, refurbishment, retesting, and reuse that may be required or specified.

Temperature Stabilization

For thermal cycle and thermal vacuum testing, temperature stabilization for a component is achieved when the component baseplate is within the allowed Test Tolerances on the specified test temperature, and the rate of change of temperature has been less than 3°C per hour for 30 minutes. For steady-state thermal balance testing, temperature stabilization is achieved when the component having the largest thermal time constant is within 3°C of its steady state value, as determined by numerical extrapolation of test temperatures, and the rate of change is less than 1°C per hour.

Applicable Documents

The following documents (listed in table below) are references used in the creation of this section. They are provided as reference and sources for this

Document Name	Document ID
Test Requirements for Launch, Upper-Stage, and Space Vehicles : Baselines	MIL-HDBK-340A, Vol.I
Test Requirements for Launch, Upper-Stage, and Space Vehicles : Applications Guidelines	MIL-HDBK-340A, Vol.II
Electromagnetic Compatibility Requirements for Space Systems	MIL-STD-1541
Requirements for the Control of Electromagnetic Interference, Characteristics of Subsystems and Equipment	MIL-STD-461E
Measurement of Electromagnetic Interference Characteristics	MIL-STD-462
Reliability Modeling and Prediction	MIL-STD-756
Test Method Standard – Microcircuits	MIL-STD-883
Test Method Standard, Electronic and Electrical Component Parts	MIL-STD-202

Test Strategy

General

- The Aerospace Corporation's proposed MIL-STD-1540E, "Test Requirements for Launch, Upper-Stage, and Space Vehicles", TR-2004(8583)-1 Rev. A
- The NASA Goddard Space Flight Center's "General Environmental Verification Standard (GEVS)", GSFC-STD-7000A;
- The Department of Defense Handbook, "Test Requirements for Launch, Upper-Stage, and Space Vehicles", MIL-HDBK-340A (Vol. 1: Baselines and Vol. 2: Applications Guidelines).

Qualification testing

Qualification tests will be conducted to demonstrate that the design and manufacturing process will produce components that meet the specification requirements. In addition, the qualification tests will validate the planned acceptance program including test techniques, procedures, equipment, instrumentation, and software.

The hardware subjected to qualification testing will be produced from the same drawings, using the same materials, tooling, manufacturing process, and level of personnel competency as used for flight hardware.

In general, a single qualification test specimen of a given design will be exposed to all applicable environmental tests. The baseline qualification strategy consists of testing dedicated hardware to qualification levels to verify the design. After qualification testing of a unit has been completed, and any design improvements incorporated, subsequent flight hardware will be acceptance tested to screen workmanship defects.

Some units are intended to be qualified via protoflight qualification testing. These units will not have a qualification unit allocated for dedicated testing. Instead the units serve as both the flight unit and the qualification unit.

Hardware qualification will occur at the unit level only. Qualification is not intended to occur at the module or payload levels.

The environmental tests performed on each of the qualification units is identified in the test matrix shown in Environmental Test Matrix. The sequence in which the environmental tests are performed is shown in the Test Flows section below.

Acceptance Testing

Acceptance tests will be conducted on each flight unit to demonstrate conformance to specification requirements and provide quality-control assurance against workmanship or material deficiencies. Acceptance testing is intended to stress screen items to precipitate incipient failures due to latent defects in parts, materials, and workmanship.

It is common to perform acceptance testing on the flight units. Acceptance testing performed at the module level is intended to provide early workmanship and material defect screening prior to unit-level build-up.

Test Objectives and Descriptions

Inspection Test

Objective

For both Qualification and Acceptance testing, the objective of the Inspection activity is to verify that the following attributes (or subset thereof) of the component are in compliance with the specifications: finish, identification markings, cleanliness, weight, and critical dimensions.

Description

Inspection and physical tests include all or a subset of the following, depending on applicability: inspection of surface finish for defects and cleanliness, verification of identification markings, measuring weight, and measuring dimensions. Inspections of flight hardware should not entail the removal of module or unit covers nor any disassembly, unless specifically called out in the test procedures.

Functional Test

Objective

For both Qualification and Acceptance testing, the objective of Functional Test activity is to verify that the electrical and mechanical performance of the component meets the specified operational requirements of the component. Full functional tests typically test the component against all operational requirements, while abbreviated functional tests verify a subset of the requirements.

Description

Electrical tests will include application of expected voltages, impedance, frequencies, pulses, and waveforms at the electrical interfaces of the component, including all redundant circuits. These parameters will be varied throughout their specification ranges and the sequences expected in flight operation. The component output will be measured to verify that it performs to specification requirements. Functional performance may also include electrical continuity, stability, response time, mechanical alignment, or other special tests that relate to a particular component configuration. Each testable assembly will have functional requirements that are unique to the module or unit. The test procedures shall specify the tests to be run at each functional test occurrence.

Functional testing of a component may consist of executing all operational tests, which is referred to as a "full functional" test. Functional tests are also performed to determine that exposure to a test environment did not induce a hardware failure. These types of functional tests typically do not require execution of all operational tests to detect the failure. For most instances, a subset of the "full functional" test suite is performed. This type of functional test is referred to as an "abbreviated functional" test.

Burn-In (Performance Stabilization)

Objective

For both Qualification and Acceptance testing, the objective of the Burn-in activity is to stabilize active component performance.

Burn-in is generally applicable at the module level or at assembly levels which contain active components. Systems Engineering and Quality Assurance will determine which modules undergo burn-in testing.

Both qualification units and flight units will be assembled from either:

1. Modules subjected to burn-in,
2. modules made of components and sub-assemblies subjected to burn-in (as appropriate), with no burn-in performed at the module level, or
3. modules that have been verified to contain components that have been burned in and waivers for components that have not been subjected to burn-in, with no burn-in performed at the module level.

Description

The test is performed with the module powered and its temperature elevated to the acceptance hot temperature. Perceptive parameters should be monitored throughout the duration of the test.

The duration of this test is such that module operational time is accumulated so that the combined duration of module thermal cycling, thermal vacuum, and burn-in/powered-on testing is at least 200 hours. The last 100 hours of operation should be failure free.

For internally redundant modules, the operating hours should consist of at least 100 hours of primary operation and at least 100 hours of redundant operation. The last 50 hours of each (primary and redundant) should be failure free.

Cumulative Powered-On Time

Objective

For both Qualification and Acceptance testing, the objective of the Accumulating Powered-On Time is to precipitate latent failures.

All units shall meet a cumulative powered-on time requirement of minimum 200 hrs.

Accumulated Power-On time is applicable at all levels, and shall be tracked for the end item data package at each assembly level.

Both qualification units and flight units' accumulated powered-on time shall be tracked.

Description

The test is performed with the assembly powered and its temperature elevated to the either acceptance hot or nominal temperature (temperatures lower than nominal temperature shall be avoided to prevent unnecessary risk of condensation).

Perceptive parameters should be monitored throughout the duration of the test.

The duration of this test is such that units' operational time is accumulated so that the combined duration of module thermal cycling, thermal vacuum, and burn-in/powered-on testing is at least 200 hours. The last 100 hours of operation should be failure free.

For internally redundant modules, the operating hours should consist of at least 100 hours of primary operation and at least 100 hours of redundant operation. The last 50 hours of each (primary and redundant) should be failure free.

Thermal Cycle Test (Ambient Pressure)

Objective

For Qualification testing, the objective of Thermal Cycle Test (Ambient Pressure) is to demonstrate the ability of electrical and electronic components to survive exposure to the qualification temperature range at ambient pressure and to characterize performance during operation in this temperature range.

For Acceptance testing, the objective of Thermal Cycle Test (Ambient Pressure) is to reveal material and workmanship defects which are reactive to thermal cycling.

Description

The minimum and maximum predicted temperatures are determined by adding uncertainty margin to the analytical minimum and maximum temperature range. The acceptance temperature range must encompass the minimum and maximum expected temperatures. The protoqualification temperature range is equal to the acceptance temperature range widened by 5°C at each temperature extreme, while the qualification temperature range is equal to the acceptance temperature range widened by 10°C at each temperature extreme. The Figure below shows the relationships between the temperature extremes defined above and when test limits are applicable.

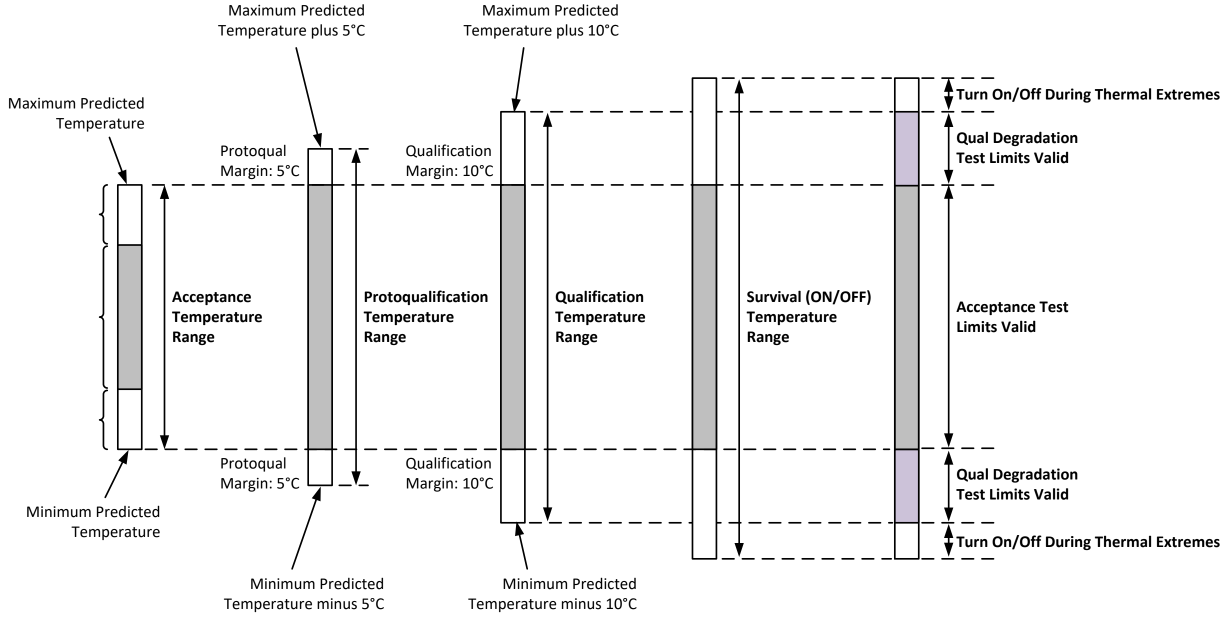

Maximum Predicted Temperature plus 5°C
Maximum Predicted Temperature plus 10°C
Maximum Predicted Temperature
Protoqual Margin: 5°C
Qualification Margin: 10°C
Turn On/Off During Thermal Extremes
Qual Degradation Test Limits Valid
Acceptance Temperature Range
Protoqualification Temperature Range
Qualification Temperature Range
Survival (ON/OFF) Temperature Range
Acceptance Test Limits Valid
Protoqual Margin: 5°C
Qualification Margin: 10°C
Qual Degradation Test Limits Valid
Turn On/Off During Thermal Extremes
Minimum Predicted Temperature
Minimum Predicted Temperature minus 5°C
Minimum Predicted Temperature minus 10°C

Figure: Component Predicted and Test Temperature Ranges; Test Limits

With the unit powered in an operational state and while monitoring critical telemetry, the test will follow the temperature profile in Figure below. The test control temperature will be measured at a representative location on the unit, such as at the mounting point on the baseplate.

A thermal cycle begins with the thermal sources and sinks at ambient temperature. With the component operating and while perceptive parameters are being monitored, the component temperature is raised to the qualification or acceptance hot temperature (based on unit under test) and follows the thermal profile Figure below. Once the component temperature has reached the qualification or acceptance hot temperature, the component will be powered off and the component temperature will be raised to the specified survival hot temperature and maintained. Once the component's temperature is stabilized at the survival hot temperature, a 1 hour soak will be performed. The component temperature will then be decreased to the qualification or acceptance hot temperature and stabilized. After stabilization, the component will be hot started and then functionally tested. Once the functional test is completed, the component temperature will be lowered to the qualification or acceptance cold temperature. The component can be powered off after at or below the unit's acceptance cold temperature to aid in reaching the desired cold temperature. Upon reaching the unit's qualification or acceptance cold temperature, the unit must be powered off (if it had not been powered off already). The component temperature will be reduced to the survival cold temperature, stabilized, and maintained for a 1 hour soak. After the 1 hour soak, the component temperature will be elevated to the qualification or acceptance cold temperature and stabilized. Once the component temperature has stabilized, the component will be cold started and functionally tested. The temperature of the sinks will then be raised to ambient conditions. This constitutes one complete thermal cycle.

The first and last thermal cycles in vacuum will have additional plateaus at the survival temperatures to allow for survival temperature testing and hot-/cold-start functional testing. The intermediate cycles are executed to the qualification or acceptance hot/cold temperature limits with the component powered.

While the component is powered during thermal cycle testing, perceptive parameters will be monitored for failures and intermittents using a practicable sampling rate. Components shall meet their performance requirements within specifications over the acceptance temperature range.

Test Levels and Duration

Pressure and Humidity

The thermal cycle test will be conducted at ambient pressure. Provisions will be taken to preclude condensation on and within the component at low temperature by flooding the chamber with dry air or nitrogen or by performing the last half cycle at the hot temperature.

Temperature

Depending on the unit purpose (qualification or flight), the component temperature will reach either the qualification or acceptance hot temperature during the hot half cycle and the qualification or acceptance cold temperature during the cold half cycle. The transitions between hot and cold should be at an average rate of 3 to 5°C per minute, and will not be slower than 1°C per minute.

Duration

Thermal soak durations will be at least 1-hour at the hot and cold temperatures. Measurement of thermal soak durations will begin after the component temperature has stabilized.

Thermal Vacuum Test

Objective

For Qualification testing, the objective of the Thermal Vacuum Test is to demonstrate the ability of the component to perform in the qualification thermal vacuum environment and to endure the thermal vacuum testing imposed on flight components during acceptance testing (if applicable). It also serves to verify the component thermal design.

For Acceptance testing, the objective of the Thermal Vacuum Test is to reveal material and workmanship defects which are reactive to the thermal vacuum environment.

Description

The component will be mounted in a vacuum chamber on a thermally controlled heat sink or in a manner similar to its actual installation in the vehicle. The component surface finishes, which affect radiative heat transfer or contact conductance, will be thermally equivalent to those on the flight components. For components designed to reject their waste heat through the baseplate, a control temperature sensor will be attached either to the component baseplate or the heat sink. The location will be chosen to correspond as closely as possible to the temperature limits used in the vehicle thermal design analysis or applicable component-to-vehicle interface criteria. For components cooled primarily by radiation, a representative location on the component case will similarly be chosen. The component heat transfer to the thermally controlled heat sink and the radiation heat transfer to the environment will be controlled to the same proportions as calculated for the flight environment.

Assemblies which have voltage present at the connector pins during launch vehicle ascent shall have the power voltages applied to their connectors during evacuation of the vacuum chamber while monitoring for corona effects. All other components are unpowered and do not have power present at their connectors. For these components the unit under test will be powered off during evacuation of the vacuum chamber.

The chamber pressure will be reduced to the required vacuum conditions (for values, see Test Levels and Duration's "Pressure" section below). Once the test pressure level has been reached, the component will have electrical power applied.

A thermal cycle begins with the conductive or radiant sources and sinks at ambient temperature. With the component operating and while perceptive parameters are being monitored, the component temperature is raised to the qualification or acceptance hot temperature (based on unit under test) and follows the thermal profile in "Typical Thermal Vacuum Profile" Figure below. Once the component temperature has reached the qualification or acceptance hot temperature, the component will be powered off and the component temperature will be raised to the specified survival hot temperature and maintained. Once the component's temperature is stabilized at the survival hot temperature, a 1 hour soak will be performed. The component temperature will then be decreased to the qualification or acceptance hot temperature and stabilized. After stabilization, the component will be hot started and then functionally tested. Once the functional test is completed, the component temperature will be lowered to the qualification or acceptance cold temperature. The component can be powered off after at or below the unit's acceptance cold temperature to aid in reaching the desired cold temperature. Upon reaching the unit's qualification or acceptance cold temperature, the unit must be powered off (if it had not been powered off already). The component temperature will be reduced to the survival cold temperature, stabilized, and maintained for a 1 hour soak. After the 1 hour soak, the component temperature will be elevated to the qualification or acceptance cold temperature and stabilized. Once the component temperature has stabilized, the component will be cold started and functionally tested. The temperature of the sinks will then be raised to ambient conditions. This constitutes one complete thermal cycle.

The first and last thermal cycles in vacuum will have additional plateaus at the survival temperatures to allow for survival temperature testing and hot-/cold-start functional testing. The intermediate cycles are executed to the qualification or acceptance hot/cold temperature limits with the component powered.

While the component is powered during thermal vacuum testing, perceptive parameters will be monitored for failures and intermittents using a practicable sampling rate. Components shall meet their performance requirements within specifications over the acceptance temperature range.

Test Levels and Duration

Pressure

For components required to operate during ascent, the time for reduction of chamber pressure from ambient to 20 Pascals (0.15 Torr) will be at least 10 minutes to allow sufficient time in the region of critical pressure. The pressure will be further reduced from 20 pascals for operating equipment, or from atmospheric for equipment which does not operate during ascent, to 13.3 millipascals (10^{-4} Torr) at a rate that simulates the ascent profile to the extent practicable.

Temperature

Hot-/cold-start and functional testing will occur at the qualification or acceptance hot and cold temperatures. The transitions between hot and cold should be at an average rate simulating flight conditions or between 1 and 5°C per minute.

Duration

The number of thermal cycles to be performed should be decided by systems engineering based upon the application. An example of the first TVAC cycle is shown in the figure below. Thermal soak durations will be at least 1-hour at the qualification hot and cold temperatures. Measurement of thermal soak durations will begin after the component temperature has stabilized.

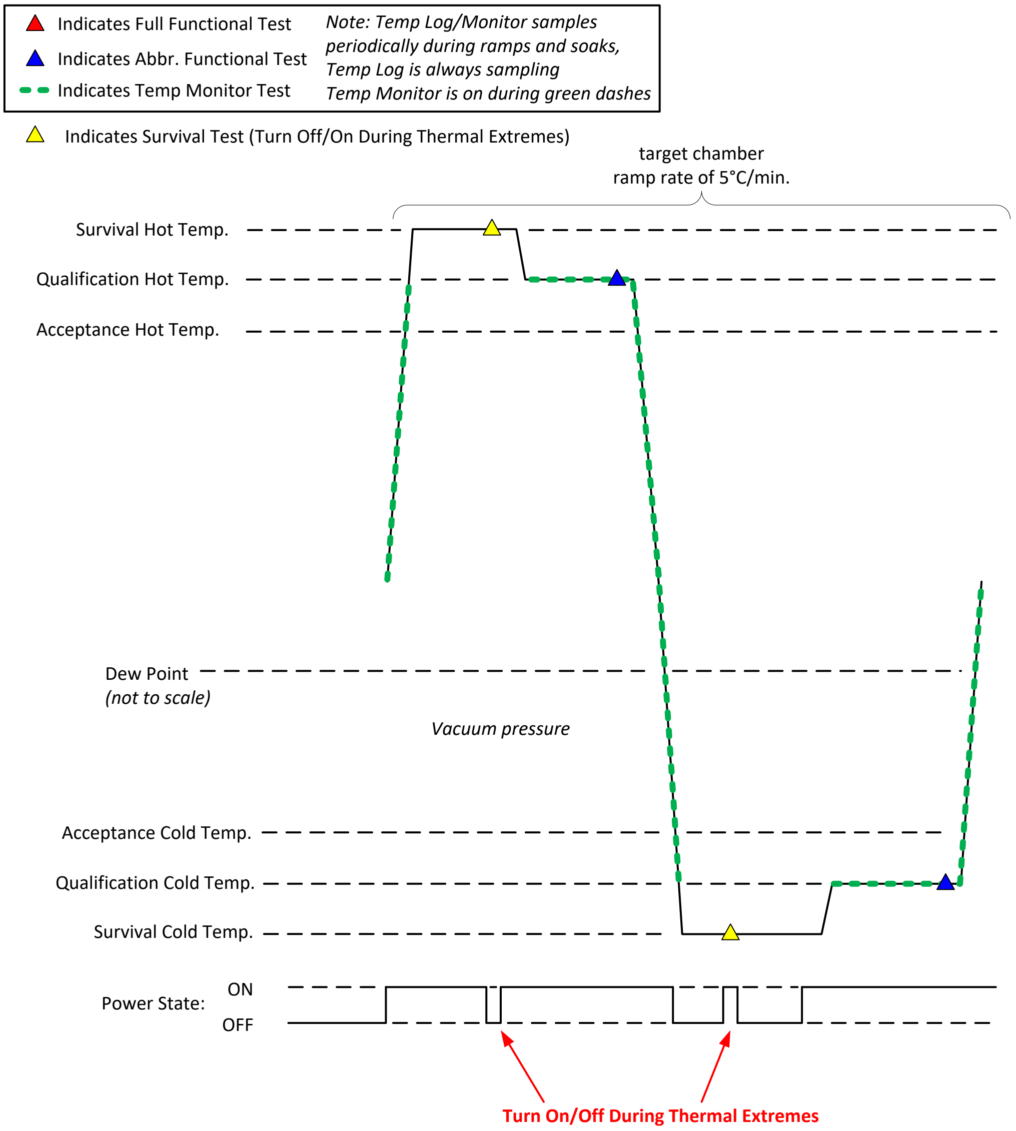

Indicates Full Functional Test
Indicates Abbr. Functional Test
Indicates Temp Monitor Test
Note: Temp Log/Monitor samples periodically during ramps and soaks, Temp Log is always sampling Temp Monitor is on during green dashes
Indicates Survival Test (Turn Off/On During Thermal Extremes)
target chamber ramp rate of 5°C/min.
Survival Hot Temp.
Qualification Hot Temp.
Acceptance Hot Temp.
Dew Point
(not to scale)
Vacuum pressure
Acceptance Cold Temp.
Qualification Cold Temp.
Survival Cold Temp.
Power State:
ON
OFF
Turn On/Off During Thermal Extremes

Figure: Typical Thermal Vacuum Cycle

Vibration Test

Objective

For Qualification testing, the objective of the Vibration Test is to demonstrate the ability of the unit to endure the extreme expected vibration environment (6dB higher than acceptance) for 3 minutes in each axis. Refer to MIL-STD-1540E for details of vibration environment definition and test level derivation.

For Acceptance testing, the objective of the Vibration Test is to reveal material and workmanship defects which are reactive to the vibration environment.

Description

The unit will be mounted to a fixture through the normal mounting points of the unit. The same test fixture should be used in the qualification and acceptance vibration tests. For units and higher-level assemblies, attached wiring harnesses up to the first attachment point, instrumentation, and other connecting items should be included as in the flight configuration if analysis indicates the connected items have a significant impact on the response. The suitability of the fixture and test control means will have been established prior to the qualification testing. Appropriate dynamic instrumentation will be installed to measure vibration responses. The component will be tested in each of 3 orthogonal axes.

For components that are unpowered and do not have power present at their connectors during launch and ascent, random vibration testing will be performed under similar conditions.

For components that have power present at one or more connectors during launch and ascent, the random vibration testing will be performed with these voltages present on the connector(s).

When necessary to prevent unrealistic input forces or component responses for components whose mass exceeds 23 kilograms (50 pounds), the spectrum may be limited or notched, but not below the minimum test spectrum for a component. The minimum spectrum for a component whose mass exceeds 23 kilograms (50 pounds) should be evaluated on an individual basis.

The vibration test does not apply to a component having a large surface causing its vibration response to be due predominantly to direct acoustic excitation.

Test Level and Duration

The recommended test levels and duration required for units, per MIL-STD-1540E, are as follows:

Qualification	6 dB above acceptance for 3 minutes/axis
Protoqualification	3 dB above acceptance for 2 minutes/axis
Acceptance	Envelope of the following for 1 minute/axis: • Maximum Predicted Environment (MPE) • Minimum level

The minimum spectrum applicable to unit-level vibration testing is the Envelope of the following two spectrums:

- Maximum Predicted Environment (MPE) for the unit
- Minimum Level

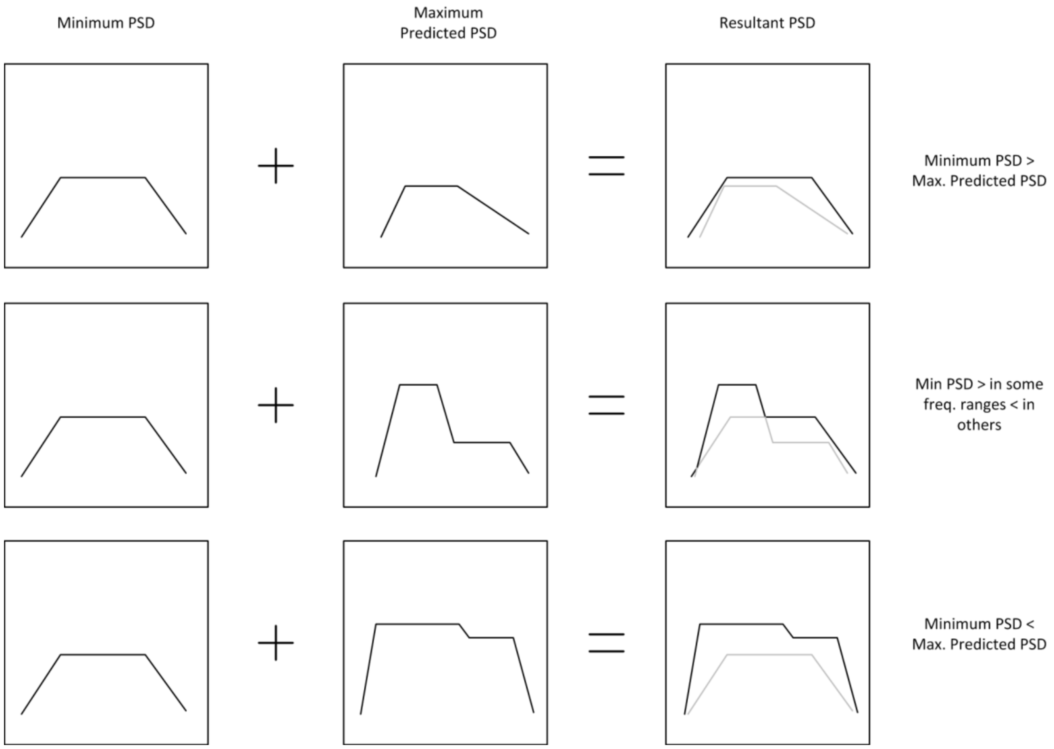

Minimum PSD
Maximum Predicted PSD
Resultant PSD
+
=
Minimum PSD > Max. Predicted PSD
Min PSD > in some freq. ranges < in others
Minimum PSD < Max. Predicted PSD

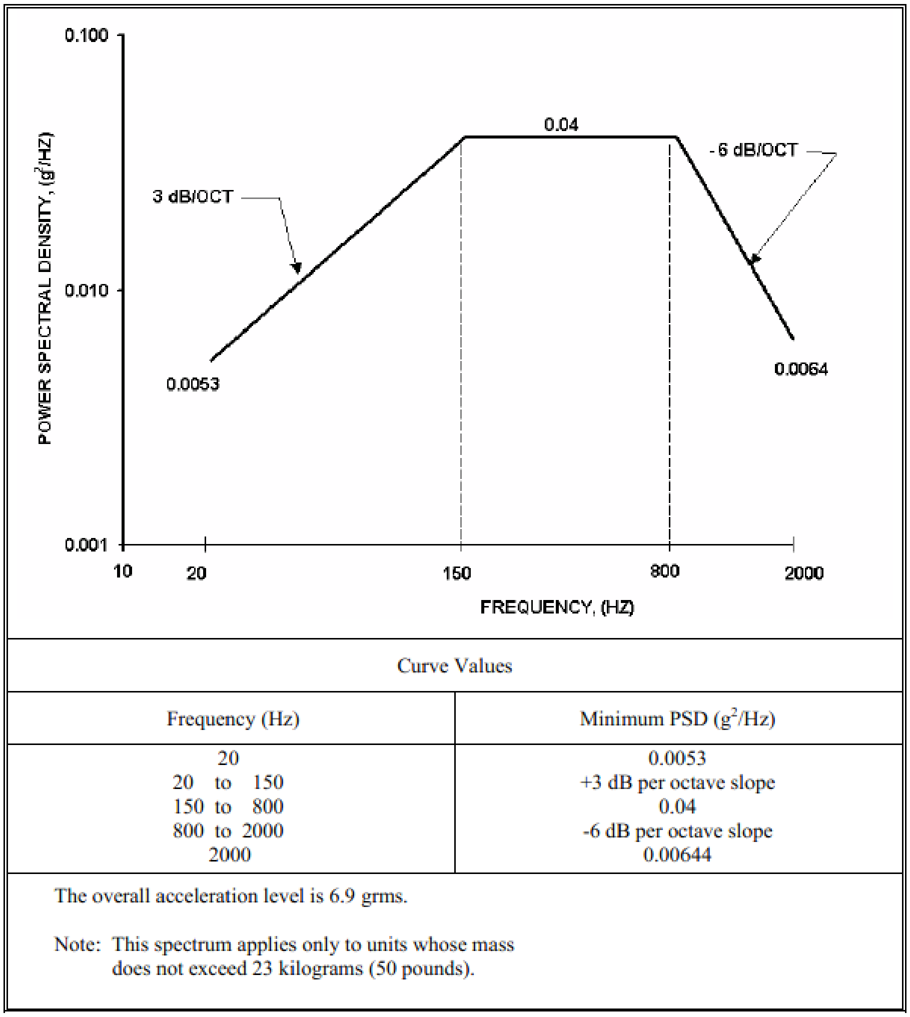
Figure: Applied Random Vibration Spectrum Envelope

Fixture Evaluation

The vibration fixture will be verified by test to uniformly impart motion to the unit under test and to limit the energy transfer from the test axis to the other two orthogonal axes (crosstalk). The crosstalk levels should be lower than the input for the respective axis. In 1/6-octave bands above 1000 Hz, exceedances of up to 3 dB are allowed provided that the sum of their bandwidths does not exceed 300 Hz in a cross axis. The dynamic test configuration (fixture and component) will be evaluated for crosstalk before testing to qualification levels.

Module Acceptance Random Vibration

For Module Acceptance, the minimum random vibration levels to demonstrate quality of workmanship are shown below.



Curve Values

Frequency (Hz)	Minimum PSD (g²/Hz)
20	0.0053
20 to 150	+3 dB per octave slope
150 to 800	0.04
800 to 2000	-6 dB per octave slope
2000	0.00644

The overall acceleration level is 6.9 grms.

Note: This spectrum applies only to units whose mass does not exceed 23 kilograms (50 pounds).

Figure: Module Acceptance Workmanship Random Vibration Spectrum

Acoustic Test

Objective

For Qualification testing, the objective of the Acoustic Test is to demonstrate the ability of the unit to endure the extreme expected vibration environment (6dB higher than acceptance) for 3 minutes in each axis. Refer to MIL-STD-1540E for details of acoustic environment definition and test level derivation.

For Acceptance testing, the objective of the Acoustic Test is to reveal material and workmanship defects which are reactive to the acoustic environment.

Description

The component in its ascent configuration will be installed in an acoustic test facility capable of generating sound fields or fluctuating surface pressures that induce component vibration environments sufficient for unit qualification or acceptance.

The unit should be mounted on a flight-like support structure or reasonable simulation thereof. Appropriate dynamic instrumentation will be installed to measure vibration responses.

Control microphones will be placed at a minimum of 4 well-separated locations at one half the distances from the test article to the nearest chamber wall, but no closer than 0.5 meter (20 inches) to both the test article surface and the chamber wall.

Testing will be performed with the component unpowered to emulate the power state during launch.

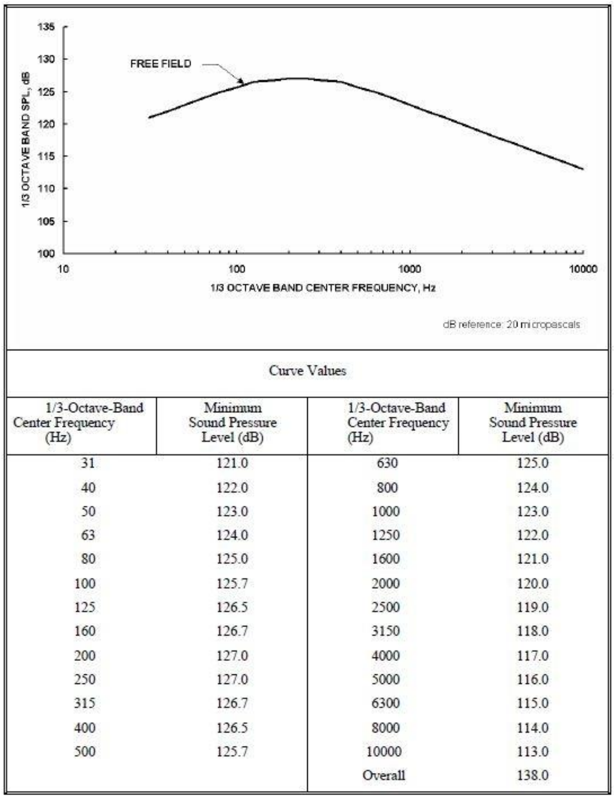
Test Level and Duration

The test levels and duration required for units are as follows:

Qualification	6 dB above acceptance for 3 minutes/axis
Protoqualification	3 dB above acceptance for 2 minutes/axis
Acceptance	Envelope of Maximum Predicted Environment (MPE) and minimum level for 1 minute/axis

The minimum spectrum applicable to unit-level vibration testing is the Envelope of the following two spectrums:

- Acoustic Maximum Predicted Environment (MPE) for the launch vehicle after the transfer functions to the unit.
- Minimum Level, see Figure below



Curve Values

1/3-Octave-Band Center Frequency (Hz)	Minimum Sound Pressure Level (dB)	1/3-Octave-Band Center Frequency (Hz)	Minimum Sound Pressure Level (dB)
31	121.0	630	125.0
40	122.0	800	124.0
50	123.0	1000	123.0
63	124.0	1250	122.0
80	125.0	1600	121.0
100	125.7	2000	120.0
125	126.5	2500	119.0
160	126.7	3150	118.0
200	127.0	4000	117.0
250	127.0	5000	116.0
315	126.7	6300	115.0
400	126.5	8000	114.0
500	125.7	10000	113.0
		Overall	138.0

Figure: MIL-STD-1540E Unit and Vehicle Minimum Acoustic Levels

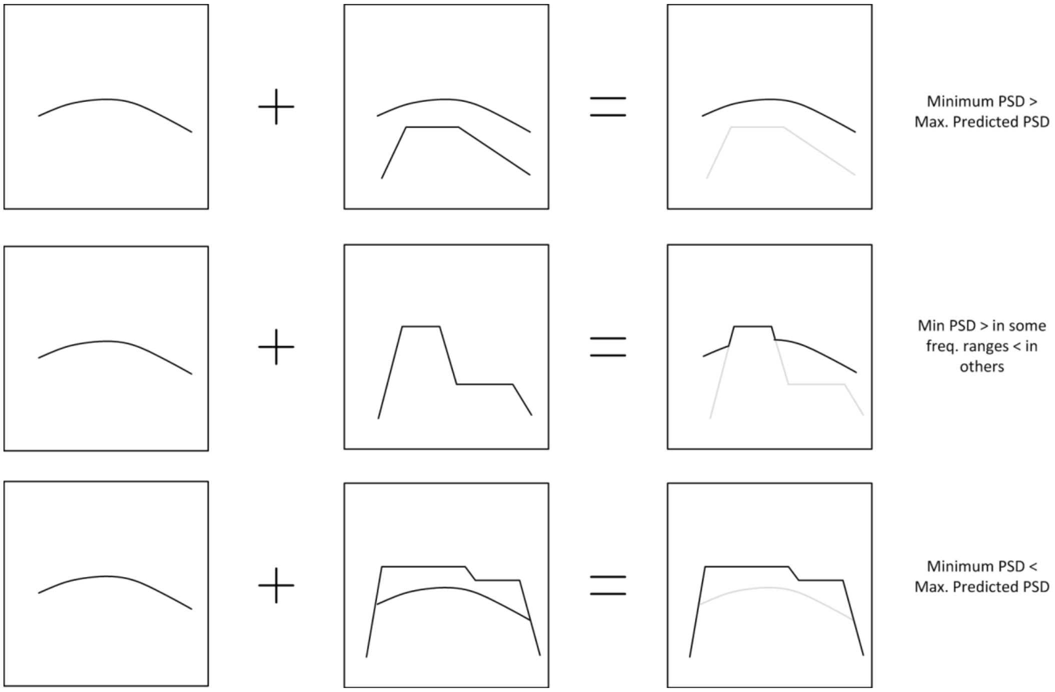


Figure: Applied Acoustic Vibration Spectrum Envelope

Shock Test

Objective

For Qualification testing, the objective of the Shock Test is to demonstrate the capability of the unit to survive shock and meet requirements after exposure to a margin over the maximum predicted shock environment.

Shock testing is not performed on flight units, but is performed on protoqualification units.

Description

The unit will be mounted to a fixture through the normal mounting points of the unit. If shock isolators are to be used in service, they will be installed. The selected test method will be capable of meeting the required shock spectrum with a transient that has a duration comparable to the duration of the expected shock in flight. A mounting of the component on actual or dynamically similar structure provides a more realistic test than does a mounting on a rigid structure such as a shaker armature or slip table. Sufficient prior development of the test mechanism will have been carried out to validate the proposed test method before testing qualification hardware. Testing will be performed with the component unpowered to emulate the power state during in-flight shock occurrences. The test environment will comply with the following conditions:

1) A transient having the prescribed shock spectrum can be generated within specified tolerances.

2) The applied shock transient provides a simultaneous application of the frequency components as opposed to a serial application. Toward this end, it will be a goal for the duration of the shock transient to approximate the duration of the service shock event. In general, the duration of the shock employed for the shock spectrum analysis will not exceed 20 milliseconds.

A shock qualification test is not required along any axis for which both the following are satisfied:

1) The qualification random vibration test spectrum when converted to an equivalent shock response spectrum (3-sigma response for Q = 10) exceeds the qualification shock spectrum requirement at all frequencies below 2000 Hz.

2) The maximum expected shock spectrum above 2000 Hz does not exceed g values equal to 0.8 times the frequency in Hz at all frequencies above 2000 Hz, corresponding to a velocity of 1.27 meters/second (50 inches/second).

Test Level and Exposure

The shock spectrum in each direction along each of the 3 orthogonal axes shall meet the test specification for that direction. For vibration or shock isolated units, the lower frequency limit of the response spectrum shall be below 0.5 times the natural frequency of the isolated unit. A minimum number of shocks will be imposed to meet the amplitude criteria in both directions of each of the 3 orthogonal axes.

The test levels and duration required for units are as follows:

Qualification	MPE + 3 dB, applied 2 times
Protoqualification	MPE + 3 dB, applied 1 time
Acceptance	Not Performed

The MPE for each unit is should be determined by the program.

Electromagnetic Interference and Compatibility

Objective

To demonstrate that the electromagnetic interference characteristics (emission and susceptibility) of the unit, under normal operating conditions, do not result in malfunctions of the unit. It also demonstrates that the unit does not emit, radiate, or conduct interference which could result in malfunction of other units.

Description

Applicable tests may include: conducted emissions, conducted susceptibility, radiated emissions, radiated susceptibility, DC magnetic field emissions, and DC magnetic field susceptibility. An evaluation will be made of each component to determine which tests will be performed as the baseline requirements.

Test Tolerances

Unless stated otherwise, the specified test parameters should be assumed to include the maximum allowable test tolerances listed in Table below.

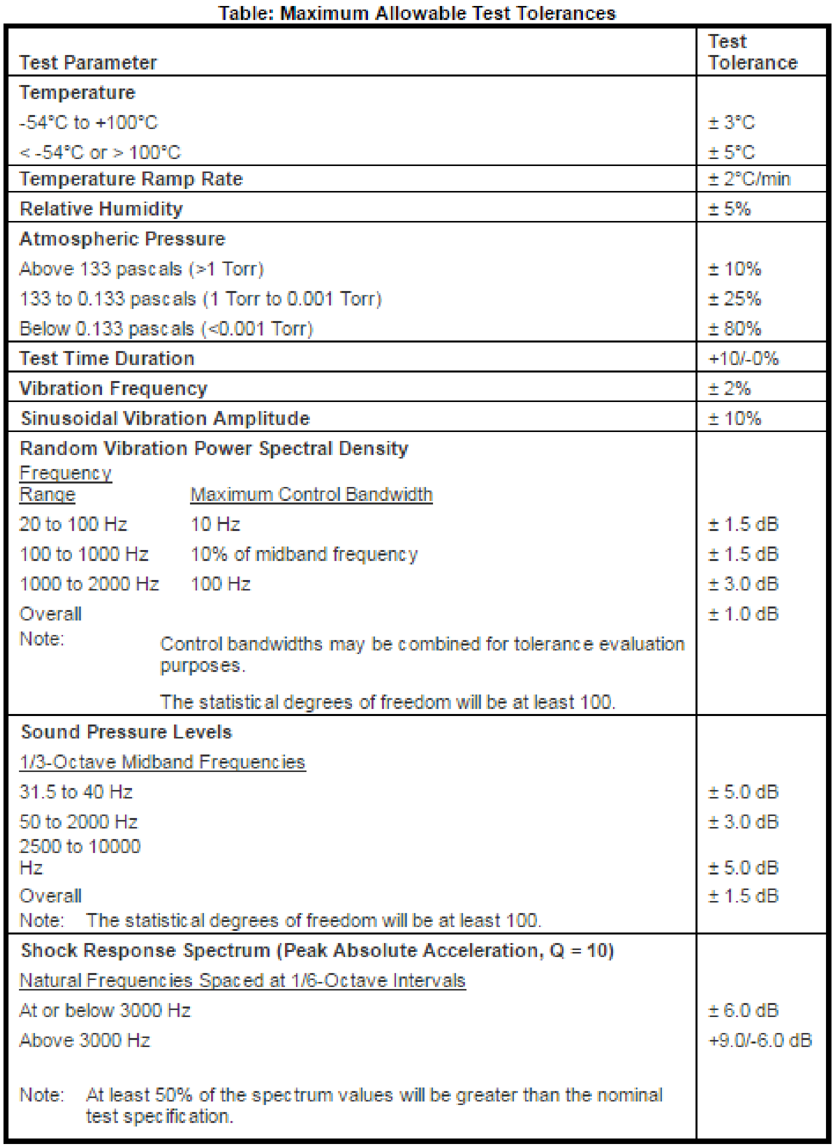


Table: Maximum Allowable Test Tolerances

Test Parameter		Test Tolerance
Temperature		
-54°C to +100°C		± 3°C
< -54°C or > 100°C		± 5°C
Temperature Ramp Rate		± 2°C/min
Relative Humidity		± 5%
Atmospheric Pressure		
Above 133 pascals (>1 Torr)		± 10%
133 to 0.133 pascals (1 Torr to 0.001 Torr)		± 25%
Below 0.133 pascals (<0.001 Torr)		± 80%
Test Time Duration		+10/-0%
Vibration Frequency		± 2%
Sinusoidal Vibration Amplitude		± 10%
Random Vibration Power Spectral Density		
Frequency Range	Maximum Control Bandwidth	
20 to 100 Hz	10 Hz	± 1.5 dB
100 to 1000 Hz	10% of midband frequency	± 1.5 dB
1000 to 2000 Hz	100 Hz	± 3.0 dB
Overall		± 1.0 dB
Note:	Control bandwidths may be combined for tolerance evaluation purposes. The statistical degrees of freedom will be at least 100.	
Sound Pressure Levels		
1/3-Octave Midband Frequencies		
31.5 to 40 Hz		± 5.0 dB
50 to 2000 Hz		± 3.0 dB
2500 to 10000 Hz		± 5.0 dB
Overall		± 1.5 dB
Note: The statistical degrees of freedom will be at least 100.		
Shock Response Spectrum (Peak Absolute Acceleration, Q = 10)		
Natural Frequencies Spaced at 1/6-Octave Intervals		
At or below 3000 Hz		± 6.0 dB
Above 3000 Hz		+9.0/-6.0 dB
Note: At least 50% of the spectrum values will be greater than the nominal test specification.		

Test Operation

Defect and Failure Tracking Procedures

Spaceflight programs must have a system in place to log the defects, test discrepancies, and test failures associated with each of the flight components. This system will contain the defect or failure information, the disposition steps, the review board attendees, and final resolution.

Test Entry Criteria

Prior to beginning qualification testing, the program should have implemented the corrective actions or design improvements associated with any design flaws, materials defects, or inadequate manufacturing processes found during the development phase.

Prior to beginning acceptance testing, the program should have implemented the corrective actions or design improvements associated with any design flaws, materials defects, or inadequate manufacturing processes found during qualification testing.

Test Exit Criteria

Qualification testing is complete when all of the qualification reports have been generated, reviewed, and accepted by Systems Engineering, Payload I&T Engineering, and Quality Assurance.

Acceptance testing is complete when all of the flight unit acceptance data packages have been generated, reviewed, and accepted by Systems Engineering, Payload I&T Engineering, and Quality Assurance.

Data Storage and Management

Test data collected during qualification and acceptance testing will be transferred from the local PC or test equipment to a server location that is regularly backed up. Test data will be organized in a database-like manner that lends itself to indexing and querying tools. This will greatly aid trend analysis, dispositions for test failures, and the design and integration effort as a whole.

Software Test/Development Concepts

Topic List

The following papers were selected from a reading list compiled for Professor James Collofello's fall 2015 CSE 565 course title, "Software Verification, Validation, and Testing":

Topic	Paper Title
Scenario Testing	An Introduction to Scenario Testing
Regression Testing	An Empirical Study of Regression Test Selection Techniques
System Test Prioritization	Improving Test Efficiency through System Test Prioritization
Defect Tracking	Enhancing Defect Tracking Systems to Facilitate Software Quality Improvement

Scenario Testing

Scenarios are hypothetical stories that are used to help a person think through a complex problem or system. A scenario test is a test based upon a scenario. The ideal scenario test has several characteristics:

- The test is *based on a story* about how the program is used, including information about the motivations of the people involved.
- The story is *motivating*. A stakeholder with influence would push to fix a program that failed this test. (Anyone affected by a program is a stakeholder. A person who can influence development decisions is a stakeholder with influence.)
- The story is *credible*. It not only *could* happen in the real world; stakeholders would believe that something like it probably *will* happen.
- The story involves a *complex use* of the program or a *complex environment* or a *complex set of data*.
- The test results are *easy to evaluate.* This is valuable for all tests, but is especially important for scenarios because they are complex. (Kaner, 1-2)

Scenario testing has many applications and uses. Early in testing, *use scenario testing to learn the product.* Once the product is more defined, scenarios can be used to *connect to documented software requirements,* especially requirements modeled with use cases. A scenario test is an instantiation of a use case (take a specific path through the model, assigning specific values to each variable). More complex scenario tests are built up by designing a test that runs through a series of use cases. A customer facing use of scenario testing is e*xposing failures to deliver desired benefits*. Scenario testing can verify whether or not a product can deliver promised capability, especially when a multitude of features are used in combination and/or sequence. Tests of individual features and mechanical combination tests of related features or their input variables (using such techniques as combinatorial testing or orthogonal arrays) are not designed to provide this kind of check (Kaner, 2-3).

Regression Testing

In a hardware or software system, regression testing is performed, with the hope to find errors caused by their changes and improve confidence that their modifications are correct. "Regression testing is the process of validating modified software to detect whether new errors have been introduced into previously tested code and to provide confidence that modifications are correct" (Graves, 184). To support this process, an initial test suite is created, upon which the regression test methodology is selected. "The simplest regression test methodology, *retest all*, repeats every test case in the initial test suite" (Graves, 185). The exact duplication of the full test suite ensures complete coverage of all possible failures (that the initial test suite can catch), but may be too expensive due to the amount of time required for a full retest. "This trade-off between the time required to select and run test cases and the fault detection ability of the test cases that are run is central to regression test selection" (Graves, 184).

Typical Regression Test

Before different regression test selection techniques are described, let *P* be a procedure or program; let *P'* be a modified version of *P*; and let *T* be a test suite for *P*.

A typical regression test proceeds as follows:

1. Select $T' \subseteq T$, a set of test cases to execute on *P'*.
2. Test *P'* with *T'*, establishing *P'*'s correctness with respect to *T'*.
3. If necessary, create *T''*, a set of new functional or structural test cases for *P'*.
4. Test *P'* with *T''*, establishing *P'*'s correctness with respect to *T''*.
5. Create *T'''*, a new test suite and test execution profile for *P'*, from *T*, *T'*, and *T''*.

Regression Test Selection Techniques and Approaches

Three families of regression test selection techniques and two additional approaches (often used in practice) are described below.

- Minimization Techniques
    - Minimization techniques attempt to select a minimal set of test cases that yield coverage for the changed or modified portion of P.
- Dataflow Techniques
    - Dataflow techniques select test cased that exercise data interactions that have been affected by modification.
- Safe Techniques
    - Techniques that are not safe fail to select a test case that would have revealed a fault in the modified program. When an explicit set of safety conditions can be defined, safe regression tests guarantee that the selected subset of test, *T', contains all test cases in the original set T, that have the ability to reveal faults in P'.*
- Ad Hoc / Random Techniques
    - This technique is generally used when time constraints prohibit the retest-all approach and no test selection tool is available. Developers will select test cases based upon "hunches", intuition, or loose associations of test cases with certain functionality. A random technique is to randomly select a predetermined number of test cases from the full list of test cases.
- Retest-All Technique
    - This technique simply retests all existing test cases.

Improving Test Efficiency through System Test Prioritization

Software testing is a strenuous and expensive process consuming at least 50% of the total development cost. “Our research goal is to develop and validate a system-level teste case prioritization scheme for identifying the more severe failures earlier in system test” (Srikanth & Banerjee 1176). This test case prioritization scheme has the purpose of identifying the more sever failures earlier in system test. The following are the four prioritization factors proposed in the paper:

- Customer Assigned Priority
- Developer-Perceived Implementation Complexity
- Fault Proneness
- Requirements Volatility

Customer Assigned Priority

Customer Assigned Priority is a measure of the importance of the requirement to the customer. The customer assigns a value for each requirement ranging from 1 to 10 where 10 is the requirement with the highest customer priority.

Faults that occur within normal execution inherently occur frequently. A majority of time should be spent finding these types of faults. The focus on customer requirements during development leads to improved customer satisfaction and perceived value. The goal of identifying and thoroughly testing the requirements with the highest customer value is to increase the business value to the customer. The requirements of higher value should be tested earlier and more thoroughly, which may be more important when testing efforts are cut short due to schedule pressures (Srikanth & Banerjee 1179).

Developer-Perceived Implementation Complexity

Developer-Perceived Implementation Complexity is a subjective measure of how difficult the implementation of a requirement is perceived to be by the development team. Each requirement is analyzed to assess the anticipated implementation complexity and is assigned a value ranging from 1 to 10; the larger value indicating higher complexity (Srikanth & Banerjee 1179).

Fault Proneness

Fault Proneness is a measure that allows the development team to identify the requirements which have had the most failures in the previous release.

As the system evolves through several versions, developers can use the data collected from prior versions to identify requirements that are likely to be error prone. This metric is based upon the number of field failures and in-house system test defects found in the code that implements the requirement. Failure Proneness is not considered for new requirements, only requirements that have already been in a released product.

Let the *i*-th requirement have a suite of test cases $T_i = \{t\}$ which test the requirement *i*. Let the set of defect $D_t = \{d\}$ be the defects discovered by executing a test suite *t*. Let there be an additional set of defects $D_i = \{d\}$ that represent field failures reported against requirement *i*. This additional set of defects is needed since field-reported failures, in this scenario, do not have associated test cases. Finally each defect d has a severity value V_d that is an exponential function of the defect severity. More specifically each defect has a severity of 1, 2, 3, or 4, with corresponding severity of 2^4, 2^3, 2^2, 2^1 (Srikanth & Banerjee 1179).

The fault proneness of the *i*-th requirement, FP_i, is directly related to the number of failures and the severity of the reported failures. The FP values are normalized to a scale of 0 to 10 as shown in the equation below:

$$S_i = \sum_{\forall t \in Ti} \left(\sum_{\forall d \in Dt} V_d \right) + \sum_{\forall d \in Di} V_d$$

$$FP_i = \frac{S_i}{\max_{\forall i} \{S_i\}} \times 10$$

Failure Proneness Equation (Srikanth & Banerjee 1179)

Requirement Volatility

Requirements Volatility is a measure based on the number of times a requirement has been changed in the development cycle normalized to a range of 0-10.

Let the *i*-th requirement have E_i recorded changes. The requirements volatility for the i-th requirement, RV_i, is computed as shown in the equation below:

$$RV_i = \frac{R_i}{\max_{\forall i} \{R_i\}} \times 10$$

Requirement Volatility Equation (Srikanth & Banerjee 1179)

Test Case Execution Order

For every requirement, the equation below is used to calculate a Prioritization Factor Value.

$$PFV_i = \sum_{j=1}^{4} (\text{Factor Value}_{ij} \star \text{Factor Weight}_j)$$

Prioritization Factor Value Equation (Srikanth & Banerjee 1181)

In the equation above, PFV_i represents the Prioritization Factor Value for requirement *i*, which is the summation of the product of factor value and the assigned factor weight for each of the factors. Factor Value_{ij} represents the value for factor *j* for requirement *i*, and Factor Weight *j* represents the factor weight for the *j*th factor for a particular product. PFV is a measure of the importance of testing a requirement. A matrix representation of the dependence of the PFV vector

$$\begin{pmatrix} \mathrm{PFV}_1 \\ \cdots \\ \cdots \\ \cdots \\ \mathrm{PFV}_n \end{pmatrix}_{(n*1)}^{P=V_w} = \begin{pmatrix} R_1^{\mathrm{CP}} & R_1^{\mathrm{IC}} & R_1^{\mathrm{RV}} & R_1^{\mathrm{CFP}} \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ R_n^{\mathrm{CP}} & R_n^{\mathrm{CP}} & R_n^{\mathrm{CP}} & R_n^{\mathrm{CP}} \end{pmatrix}_{(n*4)} \begin{pmatrix} w_{\mathrm{CP}} \\ w_{\mathrm{IC}} \\ w_{\mathrm{RV}} \\ w_{\mathrm{FP}} \end{pmatrix}$$

Vector Form of Prioritization Factor Value Equation (Srikanth & Banerjee 1181)

The computation of PFV for a requirement is used to compute the weighted priority (WP) of its associated test cases. WP of the test case is the PFV contribution of the requirement(s) the test cases maps to as shown in the equation below. Let there be *n* total requirements for a product/release, and test case *j* maps to *i* requirements. Let X_j denote the set of requirements to which the test case *j* is mapped.

$$\mathrm{WP}_j = \frac{\sum_{X \in X_j}^{i} \mathrm{PFV}_x}{\sum_{y=1}^{n} \mathrm{PFV}_y}$$

Weighted Priority Equation (Srikanth & Banerjee 1181)

WP_j is an indication of the priority of running a particular test case. The test cases are ordered for execution based on the descending order of WP values such that the test case with the highest WP value is run first and so on (Srikanth & Banerjee 1181).

Enhancing Defect Tracking Systems to Facilitate Software Quality Improvement

Software companies typically apply data from defect tracking systems to ensure reported bugs and defects eventually get fixed. The data in these systems has the potential to be used in software quality assessment (SQA) efforts and planning efforts for future software process improvement (SPI) intiatives. However, the study performed by Li, Stalhane, Conradi, & Kristiansen found that "most of the data entered in these systems was never used, irrelevant, unreliable, or difficult to apply to SQA and SPI".

Goals, questions, and metrics

A DTS aims to reduce the defect density and improve defect fixing efficiency. The DTS should provide a Quality Assurance (QA) manager with information that they could use to answer the following questions:

- What are the main defect types?
- What can the company do to prevent defects in a project's early stages?
- What are the reasons for the actual defect-fixing effort?

Validation and Follow-up

There are several attributes that a DTS should use to classify defects for their projects:

- Fixing Type
    - A set of values categorizing developers' defect-fixing activities
- Effort
    - Example A
        - Scale
            - *Simple*: developers would spend less than 20 minutes total effort to reproduce, analyze, and fix a defect
            - *Medium:* developers would spend between 20 minutes and 4 hours
            - *Exhaustive:* developers would spend more than 4 hours
        - Comments
            - This simplified Likert scale is recommended because requesting a more precise estimate of time required to fix may not be cost effective
    - Example B
        - Scale
            - Quick-fix
            - Time-consuming
        - Comments
            - Two categories can be used if the focus is to just pick out costly defects and focus on only them.
- Severity
    - Values defined a defect's impact on the software functionality and the customer's experience.
- Root Cause:
    - Project entities such as requirements, design, development, and documentation characterized each defect's origin.

Motivating Users

One major issue in improving DTSs is the pressure of meeting delivery deadlines, which makes developers believe they don't have time to fill in new defect data attributes. DTS users must be convinced that the data collection is in their interest and won't take much time.

Potential Pitfalls in Defect Data Quality

- First round analysis:
    - Developers may forget to reclassify a defect if it took longer to investigate than originally thought. Inferences made from the data set with the DTS will be affected as the number of improper classifications accumulates if this pitfall becomes systemic.
- Default values:
    - If a DTS system presents attribute values in a drop down with the first value set to a default, it could lead to a developer skipping to enter a value because they saw the system provided a default.
- Orthogonal values:
    - Defect attributes can be designed to be orthogonal, requiring that a single choice is made for each defect attribute. However, the study found that multiple choice is sometimes more applicable. Care should be taken to determine whether an attribute should be presented as a single choice or a multiple choice list.

Lessons learned from root cause analysis

Developers were found to know what was happening in their current build of code, but could not trace a defect’s causes to earlier stages of a project. Thus, different people will probably specify the root-cause attribute from own (different) perspective. If this is true, who should conduct the root cause analysis and how can a DTS system and process resolve conflicting proposals? DTSs can provide useful data to facilitate root cause analyses, but they can’t necessarily substitute for human expertise regarding the system.

Applications of Software Test Concepts to Hardware Design Paradigms

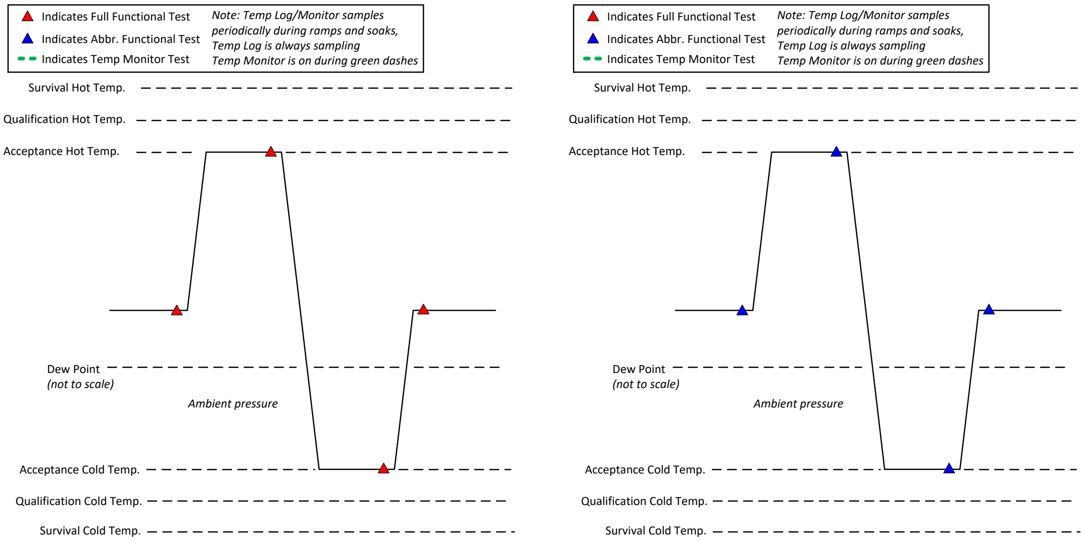
The following sections are discussions of possible applications of each software verification, validation, and testing strategy to hardware design paradigms.

Regression Testing

Functional Tests over Temperature (Full versus Abbreviated)

Any device under test (DUT) has a set of performance requirements based upon the specification that it was designed to. To test the DUT, specialized test equipment (STE) is designed. An input to the STE is the test limits file. The test limits file defines, at a minimum the following for each measurement: date/time (down to seconds in a standard time zone), measurement name, lower/upper limit (with a standard precision on both sides of the decimal), and a logical expression (relating the two limits).

For full coverage, the entire test list could be used. In the diagram below, a red triangle indicates a Full Functional Test. This is meant to convey that 100% of the test list is used. However, if 100% coverage is prohibitive or unnecessary, an abbreviated functional test could be used. In the diagram below, a blue triangle indicates an abbreviated functional test.



🟥 Indicates Full Functional Test
🟦 Indicates Abbr. Functional Test
🟩 Indicates Temp Monitor Test
Note: Temp Log/Monitor samples periodically during ramps and soaks, Temp Log is always sampling Temp Monitor is on during green dashes
Survival Hot Temp.
Qualification Hot Temp.
Acceptance Hot Temp.
Dew Point (not to scale)
Ambient pressure
Acceptance Cold Temp.
Qualification Cold Temp.
Survival Cold Temp.
🟥 Indicates Full Functional Test
🟦 Indicates Abbr. Functional Test
🟩 Indicates Temp Monitor Test
Note: Temp Log/Monitor samples periodically during ramps and soaks, Temp Log is always sampling Temp Monitor is on during green dashes
Survival Hot Temp.
Qualification Hot Temp.
Acceptance Hot Temp.
Dew Point (not to scale)
Ambient pressure
Acceptance Cold Temp.
Qualification Cold Temp.
Survival Cold Temp.

Figure: Full/Abbreviated Functional Tests over the Acceptance Temperature Ranges

There are several factors that should be considered when comparing the two thermal cycles illustrated in the above figure. First, the difference in test coverage achieved by an abbreviated functional test relative to a full functional test is not indicated. Depending on the DUT, cost of the STE, time required for each item in the test list, the coverage difference could be small or large. The second is that there is no indication of any testing being performed during the soaks during thermal soaks or during temperature ramps. As stated in the Environmental Stress Screening Section, there are flaws that are precipitated by only the temperature ramps (due to the difference in coefficients of thermal expansion between materials leading to intermittent failures).

One recommendation is that a Temp Monitor List be created. The Temp Monitor List is a further reduced test list that is meant to contain all necessary perceptive performance tests that provide coverage of flaws precipitated by temperature transitions and soaks at temperature extremes. For comparison, the figures below are the cycles shown above, with the addition of a green dotted line on the temperature ramps and soaks during which the Temp Monitor Test List is executed.

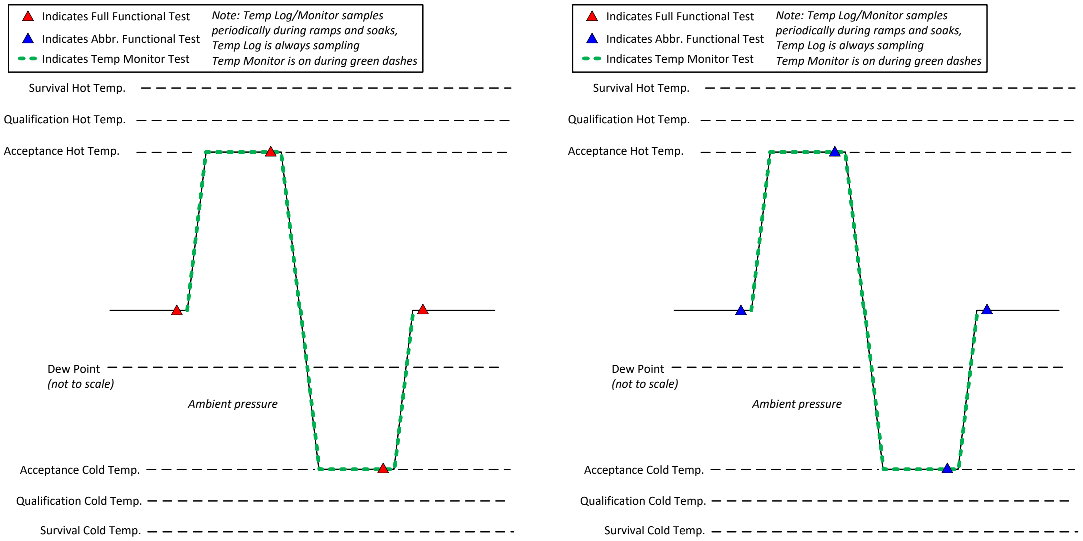


Figure: Full/Abbreviated Functional Tests over the Acceptance Temperature Ranges with Temp Monitoring during Ramps

Thermal Cycle Ramp Rate

First, let's assume that Temp Monitoring is required. Page 4-12 in MIL-HBK-344A, "Environmental Stress Screening of Electronic Equipment" identifies the following parameters as thermal cycle screening parameters when calculated defect precipitation probability:

- Temperature Range, in degC
  - The difference between the maximum and minimum applied external (chamber) temperature ($T_{max} – T_{min}$). Temperatures are expressed in degC. Care should be taken when T_{min} is negative to not subtract incorrectly.
- Temperature ramp rate, in degC/min
  - This parameter is the average rate of change of the temperature of the item to be screened as it transitions between T_{max} and T_{min} and is given by:

$$T_R = \frac{[\left(\frac{T_{max}-T_{min}}{t_1}\right)+(\frac{T_{max}-T_{min}}{t_2})]}{2}$$

- Dwell
  - Maintaining the hardware temperature constant, once it has reached the maximum (or minimum) temperature, is referred to as dwell. The duration of the dwell is a function of differences in the thermal mass of the items being screened.
- Number of Cycles
  - The number of transitions between temperature extremes (T_{max} or T_{min}) divided by two.

The ramp rate should be high enough to achieve a sufficient precipitation efficiency probability for your ESS program (see MIL-HDBK-344A for predictive equations based upon the above factors). The other factors that are recommended to be considered are:

- Coverage of Temp Monitor list relative to flaws to be precipitated during ramp events
- Temperature ramp rate, in degC/min, considering test time versus schedule cost
    - Assuming fixed dwell times at temperature extremes and a constant predicted temperature range for acceptance high and low temperatures, ramp rate and number of cycles are the two inputs required to calculate duration of the thermal cycle test event.
- Time it takes for one sample (one iteration through your Temp Monitor List)
    - For example a poll of a power supply for its current and voltage readings is a quick measurement. Two examples of longer measurements, in order of increasing test time, are a single frequency S-parameter measurement, a swept S-parameter measurement over the DUT's frequency range, and a swept Noise Figure measurement over the DUT's frequency range.
- Sampling Rate
    - How often does the Temp Monitor List get executed?
        - Every N minutes?
        - Every M degC?
    - Does the rate chosen provide enough coverage over the thermal cycle ramp to detect intermittent failures?
- Lifetime concerns of any components
    - Does the Temp Monitor List contain any state change for the hardware that accelerates wear of the hardware?
        - An example is a mechanical switch (which generally has an estimated number of switch events that it can handle throughout its life)

Block Chain for Manufacturing Process

The culmination of an ESS process and acceptance test plan will be test flows defining what sequence of test events the program decides to implement. Given that the test flow is a sequential series of test events, with a successful test triggering the formal transition to the next test step. An example of how a program traditionally controls where hardware is at in the test flow is the concept of a traveler. Traditional travelers were paper documents that travel with the assembly as it continues along a predefined route through the test flow. Obviously electronic systems are available and can be implemented.

When yield is high and tests are passed, the entire process works as theorized. Throughput of hardware is maintained, test limits can be optimized to determine if measurements are correlated and the test list can be reduced, and proper test limits ensure performance is maintained and workmanship defects are screened out.

When a device fails a test, great care must go into identifying root cause of the failure. The state of the system must be considered in either hardware or software tests. For an electrical test of hardware, a loose cable or operator error can cause a test failure that is not indicative of a failure of the device under test. However, a device failure triggers a thorough review process involving all technical parties called a Failure Review Board, or FRB. One artifact of the test failure is a Non-conforming material report, or NMR. This captures the state of the test and what measurements failed. The technician takes notes on the state of the test setup and then gets their quality assurance engineer. The purpose of the FRB and NMR is to help bring in the technical knowledge and documentation necessary to implement a program's Failure Review and Corrective Active System, or FRACAS procedure. The FRB, NMR, and FRACAS procedure will lead to a root cause and corrective action. An example of corrective action is a rework and repair event to replace a surface mount component that had a loose solder joint. The entire process from test failure to the rework and repair event is outside of the main test flow. While the conclusion of the rework and repair event is an important milestone in fixing the root cause of the failure, the repair and rework event itself is subject to an possibility of introducing an workmanship defect into the device. A common final action of the FRACAS procedure is to enforce some penalty test on the device to ensure that the rework and repair event fixed the state of the device and did not introduce workmanship defects. Two common tests are a full functional thermal cycle over acceptance temperature range, a random vibration test to workmanship/acceptance levels, or a sequential combination of both.

A proposal to keep track of the test data, the process step that the hardware is currently at, and ensure a linkage between all relevant data is the blockchain (T. Beck, personal communication, April 12, 2017). While the blockchain could be applied to any change event, below is a discussion of what a blockchain is and its possible application to manufacturing/test-flow steps.

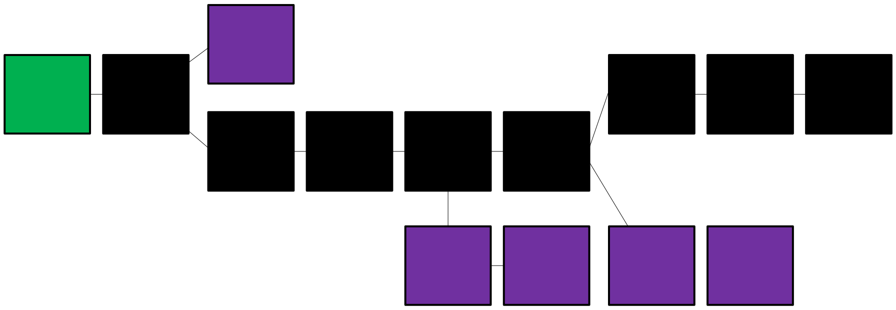


Figure: Blockchain formation.

The main chain (black) consists of the longest series of blocks from the genesis block (green) to the current block. Orphan blocks (Purple) exist outside of the main chain.

Consider the below test flow diagram, which is an example of a test flow diagram for a module level assembly's acceptance/workmanship testing. The sequence of tests is:

- an inspection test consisting of the measurement of physical parameters and pictures
- an electrical test consisting of abbreviated functional tests over acceptance temperature ranges
- an mechanical test consisting of workmanship random vibration
- an electrical test consisting of four unpowered thermal cycles to survival temperature ranges with no Temp Monitor testing
- an electrical test consisting of two powered thermal cycles to acceptance temperature ranges with Temp Monitor Testing
- an electrical test consisting of full functional tests over acceptance temperature ranges
- an inspection test consisting of final pictures and inspection of the device

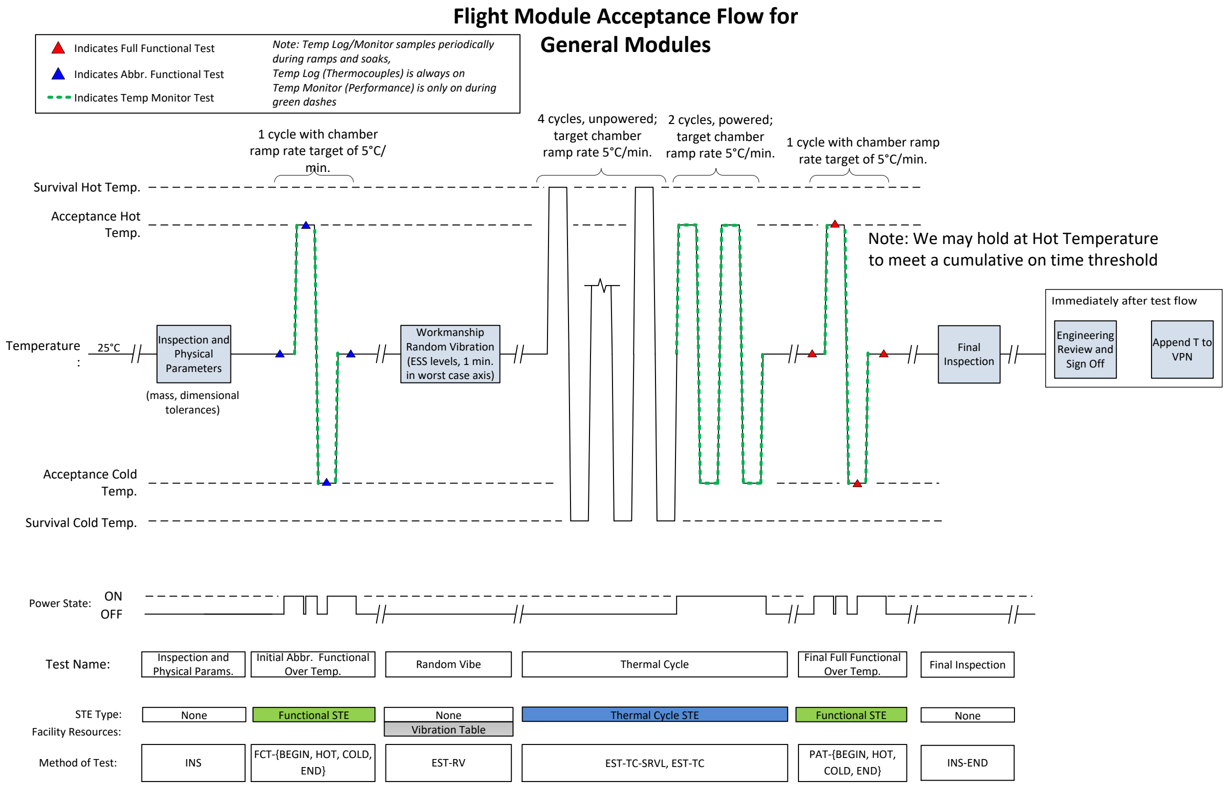

Flight Module Acceptance Flow for
General Modules
Indicates Full Functional Test
Indicates Abbr. Functional Test
Indicates Temp Monitor Test
Note: Temp Log/Monitor samples periodically during ramps and soaks,
Temp Log (Thermocouples) is always on
Temp Monitor (Performance) is only on during green dashes
1 cycle with chamber ramp rate target of 5°C/ min.
4 cycles, unpowered; target chamber ramp rate 5°C/min.
2 cycles, powered; target chamber ramp rate 5°C/min.
1 cycle with chamber ramp rate target of 5°C/min.
Survival Hot Temp.
Acceptance Hot Temp.
Note: We may hold at Hot Temperature to meet a cumulative on time threshold
Temperature :
25°C
Inspection and Physical Parameters
(mass, dimensional tolerances)
Workmanship Random Vibration (ESS levels, 1 min. in worst case axis)
Final Inspection
Immediately after test flow
Engineering Review and Sign Off
Append T to VPN
Acceptance Cold Temp.
Survival Cold Temp.
Power State:
ON
OFF
Test Name:
Inspection and Physical Params.
Initial Abbr. Functional Over Temp.
Random Vibe
Thermal Cycle
Final Full Functional Over Temp.
Final Inspection
STE Type:
Facility Resources:
None
Functional STE
None
Vibration Table
Thermal Cycle STE
Functional STE
None
Method of Test:
INS
FCT-{BEGIN, HOT, COLD, END}
EST-RV
EST-TC-SRVL, EST-TC
PAT-{BEGIN, HOT, COLD, END}
INS-END

Figure: Example Module Acceptance Test Flow

The Method of Test unique identifier, illustrated in the bottom row of boxes in the above test flow, is an example of how a program could track each test events data. The current revision of the test flow will have a start and stop time indicating when productions and operations were implementing that version. Considering for this example on one revision of a test flow, the method of test unique identifier (that has some mapping to operations routing steps) could be considered a block in the block chain. The main blockchain is the longest sequence of black blocks following the green genesis block. In this application, the main blockchain is the desired test flow.

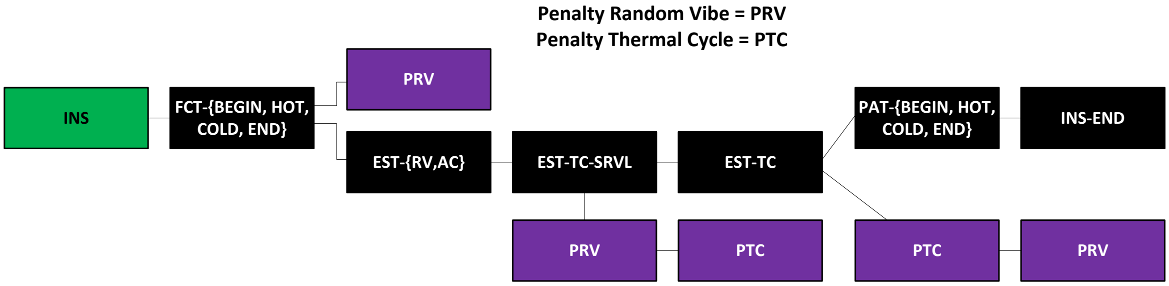

Penalty Random Vibe = PRV
Penalty Thermal Cycle = PTC
INS
FCT-{BEGIN, HOT, COLD, END}
PRV
EST-{RV,AC}
EST-TC-SRVL
EST-TC
PAT-{BEGIN, HOT, COLD, END}
INS-END
PRV
PTC
PTC
PRV

Figure: Blockchain of Test Flow's Method of Tests

Since test failures are not predictable with 100% accuracy, the blockchain method of tracing back to the main blockchain would allow a linked list like organization of manufacturing and test data. The above diagram has the appearance of a binary tree, but that analogy breaks down if there is no requirement of a maximum of one orphan path. Orphan blocks or blockchains could be added as a result of the FRB scenario, NMR report, and FRACAS procedure.

Scenario Testing

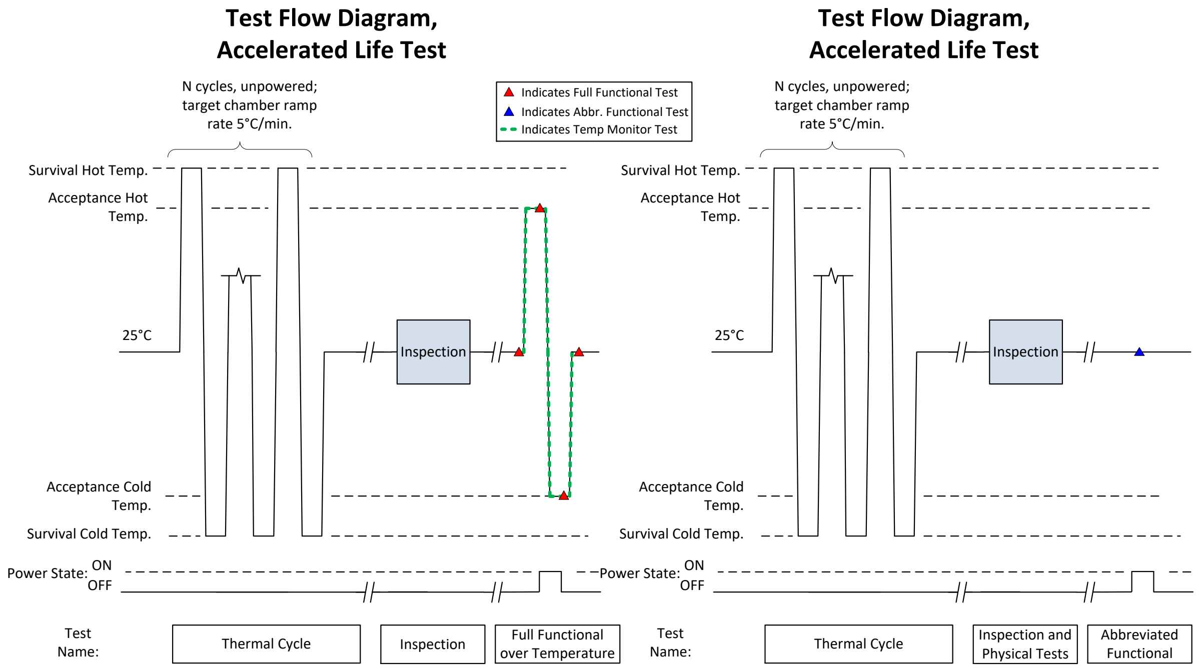
A relatively new practice in spaceflight hardware design is the application reliability engineering. Most systems have a mission life, meaning that systems are designed with the requirement that they will meet the systems performance requirements over their mission life. It is common for geostationary satellites to have mission lives upwards of 10-15 years.

A common test to validate these reliability models is Accelerated Life Testing or ALT for short. Please note that there is also (Highly Accelerated Lift Testing). Accelerated life testing is meant to predict the reliability of the system by subjecting the system to stresses similar to those found in actual service. Reliability can be defined as "the probability that a product ", or system, "will perform its intended function for a specified interval under stated conditions (such as cycling in a specified manner over a certain range of temperatures and vibration spectra). This probability is determined experimentally with repeated tests". The goal of ALT is to determine the systems mean-time-between-failure (MTBF) and mean-time-to-failure (MTTF) (Fundamentals of HALT/HASS Testing, 8).

- Mean-Time-Between-Failure (MTBF):
  - A basic measure of reliability for repairable items. This measure is the mean number of life units (hour, years, etc.) during which all parts of the item perform within their specified limits, over a particular measurement interval, under stated conditions.
- Mean-Time-To-Failure (MTTF):
  - A basic measure of reliability for non-repairable items. It is the total number of life units of an item population divided by the total number of failures within that population, over a particular measurement interval, under stated conditions.

Accelerated Life Testing to emulate system life

Consider a scenario where a system's life can be characterized by 200 thermal cycles to an acceptance temperature range. A scenario test that can be created is a sampling of some test event every N cycles up to a total of 200 cycles to emulate the system's mission life. The above discussion of regression testing is illustrated as well by the juxtaposition of different electrical tests for each sample. While the program could reduce beginning of life, it is encouraged to increase test coverage during the middle and end of life as the probability of failure increases.



System Test Prioritization

The software test case prioritization proposed in Srikanth & Banerjee's paper outlines the following four prioritization factors:

- Customer Assigned Priority
- Developer-Perceived Implementation Complexity
- Fault Proneness
- Requirements Volatility

A discussion of these factors and considerations from their application to a spaceflight hardware/software assembly are discussed in the sections to follow.

Customer Assigned Priority

Test case prioritization for space vehicle assembly testing is no different, like most systems, should have a customer assigned priority for each specification, from which test cases are mapped. The scheme presented in the paper assigning a value from 1 to 10 is recommended to be implemented at all levels of assembly.

Developer-Perceived Implementation Complexity

This category should be expanded to include the different types of developers. Large software systems have architects, developers, unit testers, and other contributors that contribute to the complexity and implementation of the system. The same is true for spaceflight hardware. Considering an assembly or sub-assembly as the top level for a moment (whatever the DUT is), the following categories are broken out:

- Module Lead Implementation Complexity
    - Generally the module lead implements the DUT hardware or software design
        - A factor consisting of the product of the implementation cost of the specific feature and the predicted cost in time and money to change/revise the physical

board, component, or software build. Both multiplicands and the product should be normalized to a value from 1 to 10.

- STE Lead Implementation Complexity
  - Combinatorial test implementation
    - Voltage Tolerance
    - Temperature Tolerance
- Test Manager
  - Cost of implementation of the hardware and facility resources necessary for the test, normalized to a normalized to a value from 1 to 10.

The program can decide on a relative weighting for each of the three groups depending on how cost and schedule constrained the program financial and temporal budgets are.

Fault Proneness

Fault proneness should consider both the STE and the DUT, if necessary.

- Repeatability of the STE's measurements
  - What is the measurement precision?
    - Generally, as part of the qualification of a STE, a report is generated detailed the measurement repeated to get a sense of measurement precision assuming a proper calibration
    - A factor should be assigned to a value normalized to a value from 1 to 10.
  - What is the calibration precision (this gets into measurement accuracy as well)?
    - Electrical calibration is generally valid for a certain period of time and is valid within a temperature range.
    - A factor should be assigned to a value normalized to a value from 1 to 10.
- Sensitivity of the STE to problems with specific test
  - Is the temperature chamber or control software difficult?
  - An optional factor or modification to the Fault Proneness factor should be considered if there are issues with the measurement set up that introduce some statistical noise on measurement confidence.
- Device under test (DUT) sensitivity
  - The test limits and specification limits should account for the device under test's sensitivities.

A factor should be assigned that is the product of the STE's measurement precision and calibration precision normalized to a value from 1 to 10. Estimates and additional weighting factors can/should be used if deemed necessary.

Note: The STE test limits will be wide when the STE is under development and should narrow to measure the device accurately given the above STE repeatability after STE certification and initial production feeds back information regarding in process shift and yield.

Requirements Volatility (Type)

Requirements Volatility will be tracked throughout the program as per a proper requirements management plan to get a sense of changes to assemblies and sub-assemblies. A recommended alternative for initial test case prioritization and good system test practices is a group of requirements type factors, listed below. Requirements volatility could be added if desired.

- Requirements Boundary Condition
  - Types:[Objective, Threshold, Both]

    - A factor should be defined to indicate which type of boundary condition a requirement has.
        - For example, a return loss specification on an S-Parameter measurement may have both a threshold of -10 dB, but an objective of -15 dB. If either is a used, a factor should be determined to separate importance of each type.
        - A more dramatic example is a power budget. If the allocation for a high power transmitter is 5 Watts, it should be identified if that is a threshold. A spacecraft operates on a finite power resource depending on the efficiency and output of the solar panels and power converters. The transmitter failing that threshold limit will break the system's power budget. The factor should be weighted highly and tested accordingly.
        - A factor should be assigned to a value normalized to a value from 1 to 10. A modification should be determined if a program wants to separate the two types and reserve a portion of the normalized scale for each or include a weighting factor based upon risk of crossing the threshold to the system.
- Requirements Type
    - Types: [Performance, Connectors/Interface, Thermal, Power]
        - Each type of requirement should be evaluated to determine the priority of testing each type. For example, it is often a good practice to exercise a limited performance test through connectors after rework and repair events because they are good indicators of ESD, device failures, or anything that affects a large group of the schematic slash path through an assembly. Each factor and combination of factors should be given a default or specific weight and/or value.
        - A factor should be assigned to a value normalized to a value from 1 to 10. If a program chose to assign individual values, a normalized sum of the products of weights and values should be calculated and used.

A factor should be assigned that is the product of the requirements boundary condition factor and the requirements type factor that is normalized to a value from 1 to 10. Estimates and additional weighting factors can/should be used if deemed necessary.

Conclusion

A program should determine how to accumulate and implement a system test prioritization scheme. Multiple factors, normalized values, factor weights, and examples applied to spaceflight hardware were presented. The implementation and type of assembly should be considered when a program or assembly looks to implement any of the above discussion.

Defect Tracking Systems

The functional goal of a defect tracking systems for spaceflight hardware of ensuring that all components used in the system are proven to be reliable. However, due to the prohibitive cost to fix a fielded spaceflight component, holistic defect tracking systems are implemented, often at cost, to ensure a certain reliability target for the systems mission life.

The additional design, traceability, and testing required to qualify a component from commercial grade to a military and space grade comes at a cost.

For example, most active SMT components used in spaceflight hardware are radiation tested and burned-in to stabilize performance. Regarding radiation testing, exceptions can be granted on a case by case basis for commercial parts that are independently radiation tested or proven to be sufficiently shielded in their system (often with the unit of equivalent thousands of an inch of Aluminum shielding). Regarding performance burn-in, SMT parts are often burned by the piece part supplier. For both cases, the program pays for the creation and tracking of all the documentation that accompanies every individual part and every lot of parts that goes into their systems.

End Item Data Package

An end item data package in spaceflight hardware systems is a compilation of the documents and processes necessary to prove that the system, at each assembly level by serial number, properly meets it performance, material, defect prevention, and reliability requirements. The information and processes required to properly accumulate the end item data package allow systems and quality engineering to answer the below questions presented in Li, Stalhane, Conradi, & Kristiansen's paper titled "Enhancing Defect Tracking Systems to Facilitate Software Quality Improvement."

- *What are the main defect types?*
- *What can the company do to prevent defects in a project's early stages?*
- *What are the reasons for the actual defect-fixing effort?*

Example EIDP Definition

An example program's definition of an End Item Data Package is shown below:

End Item Data Package

Each assembly shall have an End Item Data Package (EIDP) to document the following:

- Full traceability
- As-Built Report vs. As Designed with a completed As Built Database Template
- Actual test and physical data in accordance with the approved acceptance test procedure.
- Evidence of Mandatory Inspection points (MIPS) and successful completion.
- List of approved waivers or deviations
- List of closed non-conformances
- Certificate of Conformance
- As-built configuration photographs

Note: Provided as electronic copies.

Certificate of Conformance

The Certificate of Conformance (C of C) shall contain the following information:

- Part number and revision
- Part Name
- Serial number(s) and date codes, as required
- Quantity of items inspected
- Company's name and address
- Signature and title of recognized Quality Assurance representative

Data Traceability Architecture

The EIDP in practice is a compiled report containing all information defined above. During the data collection and implementation phases, databases should be used to store data, meta-data, and pointers to connect the below information:

- NMR Number on test failure
- Method of Test for Test Data
- Operations Routing Number for step in manufacturing flow

If multiple databases are used, a unique record should be exchanged when between systems when they communicate. For example, if the test data in a production test database triggers the creation of a NMR and that data is exchanged across systems, both databases should have a handshake that exchanges the NMR # and the Method of Test. If a program uses different unique identifiers, they should be substituted.

Additionally, if a data analysis tool is used to query the production test database, records should be kept of the mapping of test flows to operations routes. Test flows and operation routes are both representations of an assemblies test life cycle, with different information abstracted based upon the intended audience and user. Both documents should be kept in sync, with a maintained record of effective dates to effectively query databases. If necessary, a lookup of matching revisions should be kept (in case there is not a simultaneous release for both forms).

The successful implementation of a holistic group of systems will greatly enable the FRB, FRACAS, and NMR disposition scenarios that involve multiple disciplines/stakeholders from within a program. Below is a brief capture of examples of what records are generated from different stakeholders.

Requirements from stakeholders

Quality Assurance

- Evidence of Mandatory Inspection points (MIPS) and successful completion.
- List of approved waivers or deviations
- List of closed non-conformances
- Certificate of Conformance
- As-built configuration photographs

Technicians

- Mate/De-mate Logs
  - Electro-mechanical connections have a finite number of mate and de-mate events that they can withstand before failure
- Notes during each test event
  - They are possibly the only person that has firsthand exposure to the specific test event that failed. Of course, there is logged data, but the firsthand account will greatly help the FRB investigation

Test Engineering Leads

- Specific technical knowledge of their STE:
    - How to troubleshoot certain STE software failures
    - How to troubleshoot certain STE hardware failures
- Test Procedures for each type of test

Assembly Design Leads

- Review of all test data
- Specific technical knowledge of their hardware:
    - How to troubleshoot certain device failures

System Test

- Report summarizing cumulative operating time for the hardware at each assembly level
- Tools enabling the compilation of test data

Summary and Conclusion

As we have seen, there are common best practices between both the Software and Spaceflight industries. Critical applications in both industries go through rigorous screening and testing before the start of their fielded lives. Undetected failures can appear at the beginning of life or any time after. While all the above are true to both industries, the financial and temporal cost for their respective implementation are, on average, subject to vastly different assumptions.

We've discussed applications, with comments provided when assumptions differ, of the following software concepts: Scenario Testing, Regression Testing, System Test Prioritization, and Defect Tracking.   The applications of agile methodologies, open source, and other software best practices are relatively immature in Spacecraft Integration & Test field. That being said, the author hopes that the paper was beneficial in documenting current practices in the spaceflight hardware integration and test field and possible applications from a prior literature in the software verification, validation, and test field.

The author will continue to develop on the ideas presented in this paper and will continue their education and dissemination of that work whenever possible.

Acronym List

Acronym	Definition	Acronym	Definition
ALT	Accelerated Life Testing	MPE	Maximum Predicted Environment
AT	Acceptance Test	MTBF	Mean Time Between Failures
BIT	Built-In-Test	MTTF	Mean Time To Failure
C of C	Certificate of Confromance	MOT	Method of Test
dB	Decibal	MIL-HDBK	Military Handbook
D	Defect	MIL-STD	Military Standard
DTS	Defect Tracking System	MMIC	Multi-mode Integrated Circuit
DVT	Design Verification Test	NMR	Non-Conforming Material Report
DUT	Device Under Test	PWA	Printed Wiring Assembly
DC	Direct Current	PFV	Prioritization Factor Value
ESD	Electro-Static Discharge	QT	Qualification Test
EIDP	End Item Data Package	RF	Radio Frequency
EM	Engineering Module	RFIC	Radio Frequency Integrated Circuit
ESS	Environmental Stress Screening	RV	Random Vibration
EST	Environmental Stress Test	SW	Software
FR	Failure Report	SPI	Software Process Inmprovement
FRACAS	Failure Report and Corrective Action System	SQA	Software Quality Assessment
FRB	Failure Review Board	STE	Specialized Test Equipment
FP	Fault Proneness	SMT	Surface Mount Technology
FET	Field Effect Transistor	T	Temperature
Hz	Hertz	TC	Temperature Cycle
HPA	High Power Amplifier	TDR	Test Data Record
HALT	Highly Accelerated Life Testing	TRC	Test Record Classification
IT	Information Technology	TVAC	Thermal Vacuum Cycle/Chamber
I&T	Integration and Test	W	Watt
LNA	Low-Noise Amplifier	WP	Weighted Priority
MIPS	Mandatory Inpsection Points		

Works Cited

"Fundamentals of HALT/HASS Testing." (n.d.): n. pag. *HALT_HAAS_WP*. Kiethley Instruments, 2000. Web. 23 Apr. 2017. <http://www.tek.com/sites/tek.com/files/media/document/resources/HALT_HASS_WP.pdf>.

Graves, T.l., M.j. Harrold, J. Kim, A. Porter, and G. Rothermel. "An Empirical Study of Regression Test Selection Techniques." *ACM Transactions on Software Engineering and Methodology* 10.2 (April 2001): 184-208.

Kaner, Cem. "An Introduction to Scenario Testing." *Software Testing & Quality Engineering (STQE)* Oct. 2003: n. pag. Web.

Li, Jingyue, Tor Stalhane, Reidar Conradi, and Jan M. W. Kristiansen. "Enhancing Defect Tracking Systems to Facilitate Software Quality Improvement." *IEEE Software* 29.2 (2012): 59-66. Web.

Srikanth, Hema, and Sean Banerjee. "Improving Test Efficiency through System Test Prioritization." *Journal of Systems and Software* 85.5 (2012): 1176-187. Web.

United States of America. Departments of the Army, Navy, and Air Force. *Tri-Service Environmental Stress Screening Guidelines*. By Lloyd K. Mosemann, W. J. Willoughby, and Stephen R. Burdt. Alexandria, VA 22333-0001: AMCRD-1C, 1993.

United States of America. Department of Defense. Army, Navy, Air Force. Environmental Stress Screening (ESS) of Electronic Equipment. Washington, D.C.: Department of Defense, 1993. Print. MIL-HDBK-344A.