The Journal of Reliability, Maintainability, and Supportability in Systems Engineering

Winter 2018–19

Table of Contents Winter 2018–19

Editor's Note 3
John Blyler

A System Design Method to Reduce Cable Failure Propagation Probability in Cable Bundles 5
Douglas L. Van Bossuyt
Bryan M. O'Halloran
Nikolaos Papakonstantinous

Reliability is More Art than Science 13
Dev Raheja

Misperceptions of Systems Engineering: Availability & Reliability 20
Ron Carson, Ph.D., ESEP

Improving Reliability Throughout the Product Life Cycle 22
Christopher Laplante

Editor's Note

John Blyler

Quotes of Interest in this issue:

"Given the nature of cables where energy and signal functions are shared between major subsystems, the potential for failure propagation is significant."

"Reliability definitions must include "shall nots."

"Should system architects be in the business of specifying system availability and reliability?"

Evaluation of results from Reliability Growth Testing (RGT)…can determine if HALT should be redone or modified…"

Welcome to the Winter 2018–19 RMS Journal. We begin this issue with a reliability topic seldom covered in the past but one that affects most major electronic systems, namely, the routing and management of bundles of cables. Bossuyt, Papakonstantinou and O'Halloran return to the journal with a paper that presents a method for assessing the reliability of cable routing. They introduce a Cable Routing Failure Analysis (CRFA) method that integrates with system architecture tools such as functional modeling and function failure analysis. The CRFA method provides a novel way of analyzing cable routing and determining if cable routing schemes are below a desired system failure probability threshold.

The next several articles deal with the art and misconceptions of systems reliability. Dev Raheja presents an outside-the-box view of how to get the best reliability results by aiming at zero failures for the entire product life cycle. To achieve the goal, one must first understand what are the right things to do to achieve a zero failure rate. Steps toward increased creativity and a system perspective are keys to this approach.

In Ron Carson's piece on availability and reliability misconceptions, a "que sera sera" or best-effort attitude is discouraged as it may sacrifice desired system-level performance. Too often the acquirer or sponsor is forced to accept less-than-needed system reliability because "it's the best we can do" given architecture and technology choices. Carson argues that a better outcome can be achieved when system architects specify system availability and reliability from the start of the product life cycle.

The last article shifts attentions to early reliability testing activities in system development. Christopher Laplante discusses the integration and importance of accelerated life techniques into different phases of the typical life cycle, with a focus on Highly Accelerated Life Testing (HALT) and Highly Accelerated Stress Screens (HASS).

As always, I hope you find this issue to be of professional value. Please don't hesitate to share your comments and potential future articles with me via the email below. Cheers!

—John

A System Design Method to Reduce Cable Failure Propagation Probability in Cable Bundles

Douglas L. Van Bossuyt
Bryan M. O'Halloran
Nikolaos Papakonstantinous

Summary & Conclusions

This paper presents a method of assessing cable routing for systems with significant cabling to help system engineers make risk-informed decisions on cable routing and cable bundle management. We present the Cable Routing Failure Analysis (CRFA) method of cable routing planning that integrates with system architecture tools such as functional modeling and function failure analysis. CRFA is intended to be used during the early conceptual stage of system design although it may also be useful for retrofits or overhauls of existing systems.

While cable raceway fires, cable bundle severing events, and other common cause cable failures (e.g., rodent damage, chemical damage, fraying and wear-related damage, etc.) are known to be a serious issue in many systems, the protection of critical cabling infrastructure and separation of redundant cables is often not taken into account until late in the systems engineering process. Cable routing and management often happens after significant system architectural decisions have been made. If a problem is uncovered with cable routing, it can be cost-prohibitive to change the system architecture or configuration to fix the issue and a system owner may have to accept the heightened risk of common cause cable failure. Given the nature of cables where energy and signal functions are shared between major subsystems, the potential for failure propagation is significant.

Through a more complete understanding of power and data cabling requirements during system architecting, a system design can be developed that minimizes the potential for collocation of critical cable infrastructure. Reductions in critical cabling collocation may lead to a reduction in potential failure propagation pathways. The CRFA method presented in this paper relies on functional failure propagation probability calculation methods to identify and avoid potential high-risk cable routing choices. The implementation of the CRFA method may help system engineers to design systems and facilities that protect against cabling failure propagation events (cable raceway fires, cable bundle severing events, etc.) during system architecture. Implementing CRFA in the system architecture phase of system design may help practitioners to increase system reliability while reducing system design costs and system design time.

1. Background

The CRFA method presented in this paper relies upon several key areas of existing research and industry methods including complex system design, Functional Failure Modeling (FFM), and Probabilistic Risk Assessment (PRA). The important aspects of each area necessary to understand and make use of the CRFA method are reviewed in this section.

With increasing system complexity, design methods used for relatively simple product design are replaced by design methodologies specifically suited for complex systems [1, 2]. Functional modeling is often used in the early conceptual phase of system design (generally referred to as system architecture although this definition is not universally accepted) [3]. Functional models represent basic system functions and the basic flows of information, material, or energy transferred between individual functions and through the system boundary [1]. Individual functions perform actions on energy, material, or information flows [4]. Functional modeling as generally practiced in system architecting efforts often only analyzes nominal system configurations and states. Extensions to functional modeling have been developed over the last decade to analyze potential failure propagation paths and determine mitigation strategies [5]. Function Failure Identification Propagation (FFIP) was developed to model failure flows propagating through system functions and the resulting system-level failure outcomes [3, 6]. FFIP can be used to predict failure propagation paths and failure outcomes. However, FFIP cannot account for failures that cross functional boundaries or most common cause failures. The Function Failure Design Method (FFDM) provides a Failure Modes and Effects Analysis (FMEA)-style failure analysis tool to be used with functional modeling [7, 8, 9, 10]. FFDM can be used to find a large variety of potential failure modes for individual functions but FFDM cannot analyze failure propagations across non-nominal flow paths or common cause failure events. The Uncoupled Failure Flow State Reasoner (UFFSR) was developed to address the issue of analyzing uncoupled failure flow propagation in FFM [11, 12]. The UFFSR provides a geometric basis for analyzing failure flow propagation across uncoupled functions. An extension of UFFSR was developed to model failure flow arrestor functions in functional modeling. The Dedicated Failure Flow Arrestor Function (DFFAF) method replicates placing physical barriers between redundant systems to prevent a failure in one system from crossing an air gap to the other system [13]. Other methods such as Function Flow Decision Functions (FFDF) [14], a

method of developing prognostic and health management systems via functional failure modeling [15], the Time Based Failure Flow Evaluator (TBFFE) method [16], and methods to understand potential functional failure inputs to systems that are hard to predict [17] have added additional capabilities to FFM in an effort to develop a more complete FFM toolbox for practitioners.

PRA is a well-established discipline of risk analysis with over 50 years of heritage for complex systems used in a variety of industries including aerospace, petroleum, automotive, and civilian nuclear power, among other areas. System failure models are developed using event and fault trees where event trees generally show the progression of a failure through systems and fault trees generally show the progression of failure within systems. Probabilistic failure data is attached to basic failure events and through Bayesian statistical methods and Boolean algebra, a probabilistic system failure rate can be calculated. However, PRA in its basic form does not capture emergent system behavior during failure events. Instead, specific methodologies are used to assess specific emergent system behavior such as during fire or flood events in civilian nuclear reactors [18, 19, 20, 21, 22, 23, 24, 25]. While many emergent system behaviors are identified by fire and flood analysis, other emergent system behaviors can remain hidden from analysts [26, 19, 27, 28].

Common cause failure in particular has had significant attention paid over the course of PRA methodological development. Failure inducing events such as maintenance errors across a series of identical, redundant valves can lead to a common cause failure of all maintained valves. Fire and flood events often can become common cause failures, causing failure of every system in a specific area of a system. Other examples include explosive, toxic, or radioactive gas clouds; salt mine or hard rock tunnel collapse; airplane, space debris, meteor, and other impacts; and explosive deconstruction of rotating turbomachinery sending out shrapnel. Several methods have recently been developed to address common cause failure in functional modeling [29, 30, 31, 32, 33, 34, 35, 36]. However, no method currently exists in the FFM toolbox to address the issue of common cause failure events destroying or disabling multiple cables routed through the same cable pathways, ducts, raceways, bulkhead or wall penetrations, or other cable routing methods. Most efforts in cable management to prevent common cause failures focus on separating redundant and backup system cabling; isolating control, motive power, and instrumentation cabling from one another; and ensuring adequate breaker coordination to prevent ground fault wire ignition events in cable raceways. These efforts are typically performed after system architecting efforts have been completed and ignore potential benefits of analyzing and planning cable routing and bundling in the early phases of design.

2. Methodology & Case Study

The CRFA method presented in this section provides practitioners a useful method to develop a better understanding of cable routing and management during system architecture from a risk-based perspective. This section details the CRFA methodology and presents a case study of cable routing in a simplified Pressurized Water Reactor (PWR) nuclear power plant primary coolant loop pumping room where three redundant pumping systems are co-located. Two pumps are required to be active at all times for proper core cooling with the third pump acting as a "swing" pump for maintenance purposes or coming online during a failure

CABLE GROUPS
Cable group: Group331
CONTROL_SIGNAL_2
POWER_BUS_1
POWER_BUS_2
POWER_BUS_3
Group failure probability: 0.0077
System fails: true
Cable group: Group415
CONTROL_SIGNAL_3
POWER_BUS_1
POWER_BUS_2
POWER_BUS_3
Group failure probability: 0.0077
System fails: true
Cable group: Group252
CONTROL_SIGNAL_2
CONTROL_SIGNAL_3
POWER_BUS_1
POWER_BUS_2
Group failure probability: 0.0074
System fails: true

Table 1: Representative CRFA results including cable groupings with highest system failure probabilities for the primary coolant loop pumping room case study.

event involving one of the other pumps.

Step 1 of the CRFA method is to develop a functional model. Figure 1 shows the functional model of the pump room.

Step 2 involves calculating the system failure probabilities and failure flow paths using FFIP or other related FFMs as desired. Here we use FFIP to calculate the failure rate of the system. In the case study, the system failure rate is calculated using FFIP at 5.3E-4/yr.

Step 3 associates failure probabilities with individual cables failing leading to a potential common cause failure event of all co-located cables. A practitioner used to the FFIP methodology can think of this step as adding another functional block into the functional model to represent a cable, rather than using a functional flow to represent the transmission of signal, energy, or material. For those who are more familiar with PRA, this is similar to adding a basic event of a common cause failure to a fault tree. For the purposes of the case study presented to illustrate CRFA method presented here, cables are defined as any electrical physical conveyance device which is generally referred to as a cable, wire, conductor, etc. The authors have found that CRFA can also be used with optical cables, pneumatic and hydraulic hoses and hard piping, and some bulk material transport systems (e.g., conveyor belts, pneumatic tubes, slurry chutes, etc.). In the case study, individual cable failure rates were chosen from an appropriate and proprietary generic cabling failure database.

Step 4 determines all possible cable groupings. In this step, the practitioner can identify any specific cables that cannot be located next to other cables for regulatory or other reasons, and any specific cables that must be co-located. For example, if three cables are being analyzed, there are nine total possible cable combinations. The case study has a total of 12 cables with 516 possible combinations.

Step 5 analyzes system failure probability when two or more cables are co-located in a raceway. The cable failure probabilities from Step 3 are used to determine if all cables in a cable bundle may fail simultaneously. FFIP is run with each potential cable grouping identified in Step 4. Results for each cable grouping are kept separate and rank ordered from highest to lowest system failure probability.

Step 6 sets the maximum threshold for system failure probability. The authors advise that the threshold be set above the base FFIP calculation as FFIP does not generally take into account common cause cable failure. Then all cable groupings that exceed the threshold value are marked as unaccept-

able configurations from a risk perspective. All cable groupings that were not marked as unacceptable configurations are thus acceptable from a risk perspective and can be used, assuming no other mitigating circumstances, in physical system design. If no cable configuration is acceptable, this indicates a redesign of the functional model is needed. Additional redundant systems or redundant cables may also be warranted. Table 1 presents partial results from the case study where a total of 516 potential cable groupings were identified, 210 groupings were rejected due to co-location exclusions (Step 4), and 313 groupings were eliminated due to exceeding the maximum threshold set in Step 6, resulting in 38 potential cable routing configurations meeting all criteria identified in the CRFA method.

The CRFA method is now complete. Periodically through the rest of the conceptual design phase, CRFA should be re-run to verify that appropriate cable groupings and separations are maintained to meet failure probability expectations. When moving from system architecture and early system design into physical system design and layout, the information from CRFA can then be used to develop cable raceways and locate individual cables.

3. Discussion

The CRFA method presented in the previous section has been implemented in software and automated. Figure 2 presents the Graphical User Interface (GUI) of the CRFA software tool that the authors developed. The case study in this paper was prepared using the software implementation of CRFA. In the future, the CRFA software is slated for integration with a larger effort to develop a complete FFM software toolkit.

In the authors' experience, evidence of the success of CRFA can often be seen in redundant systems cabling being isolated from one another. Often this is because of Step 4 identifying cables that cannot be co-located. However, the authors have observed CRFA identifying on its own that redundant system cabling should not be co-located due to increased system failure probability. It is also possible that if the maximum threshold set in Step 6 is sufficiently high, redundant system cabling isolation may not be observed. This is potentially indicative of too high of a threshold being set or may also indicate that redundant system cabling is unnecessary. It is recommended that further review of the results and a deeper understanding of why certain cables are more or less isolated is sought before moving forward if either case is identified.

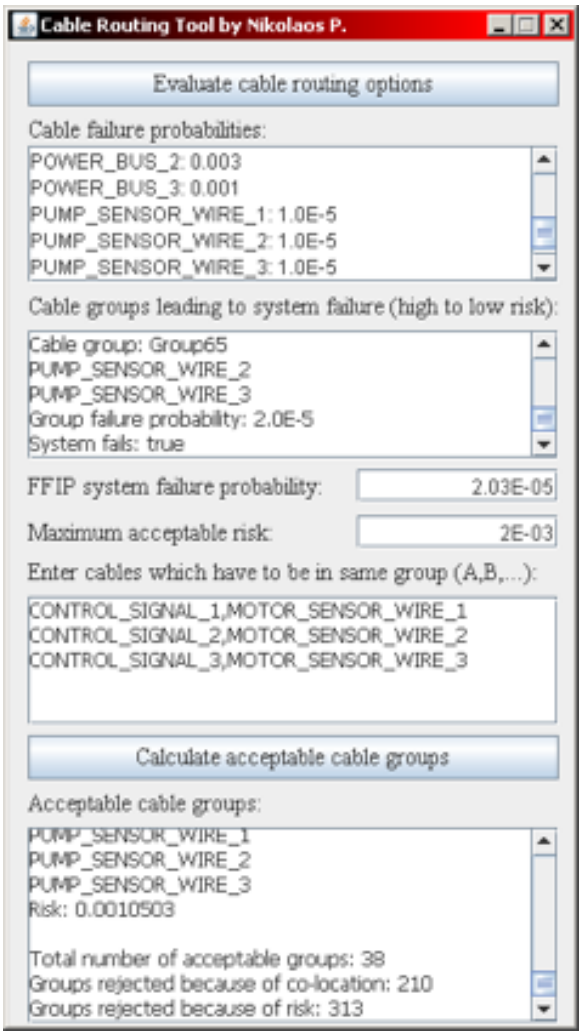

Figure 2: The GUI of the software implementation of CRFA.

While small-scale cable routing studies can be conducted using PRA tools and larger complex system cable routing analysis can be performed using specialized methods, the method presented in this paper integrates cable routing failure analysis with other FFMs, allowing a more holistic and integrated approach to system risk analysis. CRFA also provides the capability of analyzing common cause cable failures much earlier in the system design process during system architecture than existing methods allow. Shifting the analysis of common cause failures from cable routing to earlier in the system design process may save both time and money in the design process.

In the case where PRA is used to analyze cable failures without analyzing fire, flood, or missile (turbomachinery shrapnel) common cause failure, the PRA results will likely underestimate failure probability. Even when analyzing the fire, flood, or missile common cause failure sources, the results will likely not present as full and accurate of a picture of cable grouping failure risks as CRFA does.

CRFA has been used to conduct analysis on a variety of systems including civilian nuclear power plants of several types, aerospace systems, automotive systems, and defense systems. The results are promising and have been useful for practitioners to understand how cable routing and management can be greatly impacted by system architectural decisions. Feedback from some users of CRFA indicate a desire for CRFA to be integrated into commonly used model based systems engineering (MBSE) tools.

Further development of CRFA is anticipated including a more nuanced approach to cable bundling. CRFA assumes that all cables co-located in a raceway will all fail simultaneously when a common cause failure event occurs. However, not all common cause failure events will cause all cables to fail. For instance, a very hungry rat will not simultaneously eat through all data cables in a large bundle. A potential extension of CRFA may be to include aspects of TBFFE in the modeling of cable bundle failures to represent failure of cables in a bundle over time. Thus, CRFA is a conservative method in this regard. Another area of future improvement for CRFA is integrating the method with uncoupled failure flow methods such as UFFSR. Uncoupled failure flows can be accounted for to some degree in Step 3 by assigning failure probabilities for common cause cable failures from potential uncoupled sources such as missiles or floods (of cable insulation-eating liquids). However, some sources of uncoupled failure flow may be missed without integration of UFFSR.

Further future work includes adding the ability to the software implementation of CRFA to automatically add redundant cabling. For instance, civilian nuclear power plants often contain three redundant sensors with three redundant cables where a functional model may only show one functional block to represent the three redundant sensors and cables. Additional automation may provide the practitioner with a more rapid development process.

4. Conclusion

The CRFA method presented here provides a novel way of analyzing cable routing and determining cable routing schemes that are below a desired system failure probability threshold. Protecting critical cabling infrastructure and separating redundant cables is vitally important to ensuring that a common cause failure does not cause a system-level failure event. Cable routing and planning currently happens late in the design process after major architectural decisions have been made and during physical system design. The CRFA method brings the analysis and

design of cable raceways and cable separation to the system architecting phase of system design using FFM as a basis for further analysis. By having a more complete understanding of cable requirements during the early phases of system design, a system architecture and design can emerge that minimizes critical cabling infrastructure co-location and identifies the need for additional redundant cabling needs. Implementing CRFA may help engineering practitioners design complex systems and facilities that guard against cable failure propagation events that could disable or destroy the core functionality of the system. Thus, system reliability is expected to be increased while driving down system risks that may otherwise have gone unaddressed.

5. Acknowledgements

This research was partially supported by United States Nuclear Regulatory Commission Grant Number NRC-HQ-84-14-G-0047 and by the Naval Postgraduate School. Any opinions or findings of this work are the responsibility of the authors, and do not necessarily reflect the views of the sponsors or collaborators. The case study or example presented in this paper may not be used or construed as an analysis of a specific system or plant and is only provided for illustrative purposes of the method.

References

1. R. B. Stone and K. L. Wood, "Development of a Functional Basis for Design," ASME Journal of Mechanical design, vol. 122, no. 4, pp. 359-370, 2000.
2. D. L. Van Bossuyt, I. Y. Tumer and S. D. Wall, "A case for trading risk in complex conceptual design trade studies," Research in Engineering Design, vol. 24, no. 3, pp. 259-275, 2013.
3. D. Jensen, T. Kurtoglu and I. Y. Tumer, "Flow State Logic (FSL) for Analysis of Failure Propagation in Early Design," in ASME International Design Engineering Technical Conference IDETC/CIE, San Diego, CA, 2009.
4. J. Hirtz, R. Stone, D. McAdams, S. Szykman and K. Wood, "A Functional Basis for Engineering Design: Reconciling and Evolving Previous Efforts," Research in Engineering Design, vol. 13, no. 2, pp. 65-82, 2002.
5. I. Y. Tumer and R. B. Stone, "Mapping Function to Failure Mode During Component Development," Research in Engineering Design, vol. 14, no. 1, pp. 25-33, 2003.
6. T. Kurtoglu and I. Y. Tumer, "A Graph-Based Fault Identification and Propagation Framework for Functional Design of Complex Systems," ASME Journal of Mechanical Design, vol. 130, no. 5, 2008.
7. M. Stock, R. B. Stone and I. Y. Tumer, "Going Back in Time to Improve Design: The Function-Failure Design Method," in ASME Design Engineering Technical Conference DTM, Chicago, Il, 2003.
8. K. G. Lough, R. B. Stone and I. Y. Tumer, "Function Based Risk Assessment: Mapping Function to Likelihood," in ASME International Design Engineering Technical Conference DET, Long Beach, CA, 2005.
9. M. Stock, R. B. Stone and I. Y. Tumer, "Linking Product Functionality to Historic Failure to Improve Failure Analysis in Design," Research in Engineering Design, 2005.
10. R. A. Roberts, R. B. Stone and I. Y. Tumer, "Deriving Function-Failure Information for Failure-Free Rotocraft Component Design," in ASME Design Engineering Technical Conference DETC, Montreal, Canada, 2002.
11. I. Ramp and D. L. Van Bossuyt, "Toward an Automated Model-Based Geometric Method of Representing Function Failure Propagation Across Uncoupled Functions," in ASME International Mechanical Engineering Congress and Exposition IMECE, Montreal, Canada, 2014.
12. Bryan M. O'Halloran, N. Papakonstantinou and D. L. Van Bossuyt, "Modeling of Function Failure Propagation Across Uncoupled Systems," in Reliability and Maintainability Symposium (RAMS), Palm Harbor, FL, 2015.
13. M. R. Slater and D. L. Van Bossuyt, "Toward a Dedicated Failure Flow Arrestor Function Methodology," in ASME International Design Engineering Technical Conference and Computers in Information Conference, Boston, MA, 2015.
14. A. R. Short and D. L. Van Bossuyt, "Rerouting Failure Flows Using Logic Blocks in Functional Models for Improved System Robustness: Failure Flow Decision Functions," in ASME International Design Engineering Technical Conference and Computers and Information in Engineering Conference, Boston, MA, 2015.
15. G. L'Her, D. L. Van Bossuyt and B. M. O'Halloran, "Prognostic systems representation in a function-based Bayesian model during engineering design," International Journal of Prognostics and Health Management, vol. 8, no. 2, p. 23, 2017.
16. J. Dempere, N. Papakonstantinou, B. O'Halloran and D. Van Bossuyt, "Risk Modeling of Variable Probability External Initiating Events in a Functional Modeling Paradigm," The Journal of Reliability, Maintainability, and Supportability in Systems Engineering, 2018.
17. D. L. Van Bossuyt, B. M. O'Halloran and R. M. Arlitt, "Irrational System Behavior in a System of Systems," in IEEE System of Systems Engineering Conference, Paris, 2018.
18. M. Stamatelatos and D. Homayoon, Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners, NASA, 2011.
19. US Nuclear Regulatory Commission, "Standard Review Plan for the Review of Safety Analysis Reports for Nuclear Power Plants: LWR Edition - Severe Accidents (NUREG-0800, Chapter 19)," US NRC, 2012.
20. D. L. DeMott, "PRA as a Design Tool," in Reliability and Maintainability Symposium (RAMS), 2011.
21. W. E. Vesely, "Extended Fault Modeling Used in the Space Shuttle PRA," in Reliability and Maintainability Symposium (RAMS), 2004.
22. L. Meshkat, "Probabilistic Risk Assessment for Decision Making During Spacecraft Operations," in Reliability and Maintainability Symposium (RAMS), 2009.
23. L. L. Lydia, A. J. Ingegneri, L. Ming and D. F. Everett, "Probabilistic Risk Assessment: A Practical and Cost Effective Approach," in Reliability and Maintainability Symposium,

2007.
24. J. Zamanali, "Probabilistic Risk Assessment Applications in the Nuclear Power Industry," IEEE Transactions on Reliability, vol. 47, no. 3, 1998.
25. T.-Y. Hsiao and C.-N. Lu, "Risk Informed Design Refinement of a Power System Protection Scheme," IEEE Transactions on Reliability, vol. 57, no. 2, pp. 311-321, 2008.
26. C. Dunglinson and H. Lambert, "Interval Reliability for Initiating and Enabling Events," IEEE Transactions on Reliability, vol. 32, no. 2, pp. 150-163, 1983.
27. M. Garvey, F. Joglar and E. P. Collins, "HRA for Detection and Suppression Activities in Response to Fire Events," in Reliability and Maintainability Symposium (RAMS), 2014.
28. US Nuclear Regulatory Commission, "PRA Procedures Guide: A Guide to the Performance of Probabilistic Risk Assessments for Nuclear Power Plants (NUREG/CR-2300)," US NRC, 1983.
29. S. Sierla, B. O'Halloran, T. Karhela, N. Papakonstantinou and I. Y. Tumer, "Common Cause Failure Analysis of Cyber-Physical Systems Situated in Constructed Environments," Research in Engineering Design, vol. 24, no. 4, pp. 375-394, 2013.
30. M. Myrsky, H. Nikula, S. Sierla, J. Saarinen, N. Papakonstantinou, V. Kyrki and B. O'Halloran, "Simulation-Based Risk Assessment of Robot Fleets in Flooded Environments," in IEEE Conference on Emerging Technologies and Factory Automation (ETFA), 2013.
31. N. Papakonstantinou, S. Sierla, D. C. Jensen and I. Y. Tumer, "Simulation of Interactions and Emergent Failure Behavior During Complex System Design," Journal of Computing and Information Science in Engineering, vol. 12, no. 3, 2012.
32. R. P. Hughes, "A New Approach to Common Cause Failure," Reliability Engineering, vol. 17, no. 3, pp. 211-236, 1987.
33. K. N. Fleming, A. Mosleh and R. K. Deremer, "A Systematic Procedure for the Incorporation of Common Cause Events into Risk and Reliability Models," Nuclear Engineering and Design, vol. 93, no. 2, pp. 245-273, 1986.
34. W. E. Vesely, "Estimating Common Cause Failure Probabilities in Reliability and Risk Analyses: Marshall-Olkin Specalizations," Nuclear Systems Reliability Engineering and Risk Assessment, pp. 314-341, 1977.
35. H. W. Lewis, R. J. Budnitz, W. D. Rowe, H. C. Kouts, F. Von Hippel, W. B. Loewenstein and F. Zachariasen, "Risk Assessment Review Group Report to the US Nuclear Regulatory Commission," IEEE Transactions on Nuclear Science, vol. 26, no. 5, pp. 4686-4690, 1979.
36. Idaho National Engineering and Environmental Laboratory, "Common-Cause Event Failure Insights NUREG/CR-6819," 2003.
37. B. M. O'Halloran, N. Papakonstantinou and D. L. Van Bossuyt, "Cable routing modeling in early system design to prevent cable failure propagation events," in IEEE Reliability and Maintainability Symposium (RAMS), 2016.

Reliability is More Art than Science

Dev Raheja

Introduction

The impact of reliability extends far beyond reliability itself. It establishes the costs of repairs, safety, maintenance and logistics, and the indirect costs of unavailability, downtime, and the cost of recalls throughout the life of the product. This paper presents an outside-the-box view of how to get the best results by aiming at Zero Failures for the entire expected life cycle which should result in an extraordinary ROI (return on investment). This requires more creativity like that of a experienced music symphony orchestra aiming for perfect performance. The Reliability orchestra is composed of senior management, R&D staff, Reliability staff, Manufacturing staff, Safety staff, Marketing Staff, Maintainability Engineering Staff and more.

Introduction

As we are all aware, some companies have built up a long reputation for reliability. Most often, the companies do not use that competitive advantage as a marketing tool. Someone offering a ten-year warranty or promising high availability could easily grab considerable market share, by applying the right effort at the right time. That is what Hyundai did when their market share was going downhill. As soon as they offered ten year or 100,000 mile warranty, the market share gained positive momentum for years.

Reliability is a process. If the right process is not followed, results cannot be right. The organizations will be under the impression that they are following a process and doing the right things. However, the

results can be far from what they imagine. It is hard enough to do the right things, but it is even harder to know what the right things are! That is where the art of reliability starts.

The Right Things To Do

Knowledge of right things comes from learning to use creativity just as the artists and musicians do. Just having partial facts at your fingertips does not work. One must utilize the accumulated wisdom for arriving at the right decision. This needs creative brainstorming for the zero failure requirements; theoretical knowledge alone will not do. Take the example of driving. One cannot learn to drive well from books alone; one should also know when to apply and how to apply the knowledge of hazard prevention.

Some things are always the right things to do, regardless of the industry or company. Based on my experience of 30 years in reliability engineering, the following are some right things:

- Mission failure should not be an option—component failure is.
- Design for twice the life—because it is cheaper if you do it right the first time.
- Reliability must be explicit in the system definition.
- Reliability definition must include "shall not" requirements, such as “the car shall not have sudden acceleration.”
- Design out for sneak failures.
- Reliability must be in the manufacturing process definition to prevent production defects.
- Design for cheaper, faster, and better products.
- Traditional budgets are illusions—avoid groupthink process where everyone agrees without challenging.

Mission Failure Should Not Be an Option; Component Failure is

I always talk about “Twice the Life—No Failures—500% ROI.” When I say no failures, I am talking about no mission failures. The mission for a driver is to complete the trip safely without noticeable downtime. A component can fail without causing the mission to fail. That is one reason why redundancy is built into critical functions, in hardware as well as in software.

For the same reason I give more importance Functional FMEA (failure mode and effects analysis) than to the component (FMEA). I emphasize functional FMEA for critical functions, which highlights the important components as well. By all means, if you do have the resources to perform component FMEA, please do so. This analysis can reveal risks that you may have missed in the functional FMEA. After all, even the functional analysis is only as good as the people performing it.

Sometimes including a redundant component is not feasible. If the brake of a heavy duty truck fails and redundant brake is not a choice, is it OK? In my view, as long as the brake system gives warning by prognostics health monitoring sufficiently in advance to let the driver complete the mission of delivering goods on time, then it may be acceptable. Why is it that a component failure is acceptable while mission failure is not? Usually the customers are unwilling to accept the downtime and the cost of the mission not being accomplished. NASA space flights were shut down for almost three years after the accident to the Space Shuttle Challenger. The cost of the seal that failed was minimal, but the cost of losing the entire shuttle along with the loss of life and the cost of shutting down the entire program were enormous. Thus we see that reliability has a heavy impact on availability,

downtime costs, the cost of managing the logistics after the mission failure, and even the cost of safety recalls.

Design for Twice the Life

Let me explain why we need to qualify components for twice the normal lifetime. The simple answer is that components are required to have 100% design margin according to the fundamental engineering principles to avoid warranty costs. Because this principle is often overlooked, most organizations have very high warranty costs and millions of product recalls. Designing for twice the life prevents most of the warranty costs resulting in return on investment of at least 1000% according to my experience with aerospace and commercial organizations. Therefore it is lot cheaper to design for twice the life. Those who understand this paradox and take advantage of it are the real masters of reliability. U.S. Navy requires designing for four times the life for safety related components where human lives are in danger.

The cost for designing for twice the normal life is a one-time investment, but the savings are much more than the cost of warrantees and recalls. If you think twice the lifetime is a tough goal, think five times normal lifetime a company practices. It designs brakes to last five times the advertised life with a return on investment (ROI) of several thousand percent. Once I worked for a Midwest company (later acquired by the Cooper Industries) that sold components for high voltage power transformers with the goal of zero failures for 15 years. In 1974, with an investment of just $50,000, the company was able to increase its market share by 200% in two years because of the 15-year warranty. Its ROI was the highest among the Fortune 500 electrical companies. At that time no company would have dreamed of a warranty for more than a year. The salesmen were sending flowers to our engineers because they were earning big commissions effortlessly. With twice the life approach you can offer a six-year warranty if your competitor starts offering a three-year warranty. Since the benefits are several times the cost, the net cost is zero. We can conclude that it costs nothing to accomplish twice the life if we look at the real costs of ownership.

There is another good reason to design for twice the normal lifetime. We need to cut down the cycle time for testing new products. To do this we usually have to conduct accelerated tests at least at twice the normal load, and preferably three times the normal load. Often a different but cheap fix is available, such as rounding a sharp corner, changing the chemistry or heat treating method, or using a different shape. In one case, a square shape was changed to a round shape, resulting in a ten-fold increase in lifetime and a 70% reduction in cost. In the case of an electronic product, the reliability was improved by 400% by eliminating 1,200 components.[1] This company had been steadily losing market share because of extremely low reliability. Now it is a market leader. If such improvements are made at the concept approval stage, the probable cost is only a pencil and an eraser. It may seem too simple. Yes, the solution is often simple, but knowing the root cause of the problem is not simple. It often requires a fault tree analysis by an experienced professional.

Reliability Must Be in the System Definition

My experience shows and I have confirmed it in my courses that almost all performance specifications are at least 60% incomplete or vague. The product functions are often vaguely defined. There is often nothing in the specification about modularity, reliability, safety, service-ability, logistics, human factors,

diagnostics capability, or prevention of warranty failures. Very few specifications address even obvious requirements, such as internal interface, external interface, user-hardware interface, user-software interface, and how the product should behave if and when a sneak failure occurs. Those who are trying to build reliability around a faulty specification should only expect a faulty reliability. Unfortunately, most companies think of reliability when the design is already approved. At this stage there is no budget and no time for major design changes. All a company can do is hope for a reasonable reliability and commit to do it better the next time.

The word reliability means that a product will perform all that is claimed for it in the system performance specification, for its specified life cycle. If the specification contains only 40% features, how can one even think of reliability? Reliability is not possible without accurate specifications. Therefore, writing accurate performance specifications is the pre-requisite for reliability. Such specifications should aim at zero failures for the modes that result in product recalls, high downtime, safety, and inability to diagnose. My interviews with those attending my corporate training reveal that the dealers are unable to diagnose about 50% of the problems (no-faults-found).

Reliability Definition Must Include "Shall Not"

To ensure the accuracy and completeness of a specification, only those who have the knowledge of what makes a good specification should approve it. They must ensure that the specification is clear on what the system should never do, however stupid it may sound. For example: "The SUV shall not roll over in case of snow or low tire pressure" or "There shall be no sudden acceleration in the cruise control."

In addition, the marketing and sales experts should participate in writing the specification to make sure that old warranty problems "shall not" be in the new product and that there is enough gain in reliability to give the product a competitive edge. It is not just the reliability but also the limits on downtime, product friendliness, and modularity that influence software reliability. Similarly, an analysis of downtimes should be conducted by service engineering to ensure that each fault will be diagnosed in a timely manner, repairs will be quick, and life cycle costs will be reduced by extending the maintenance cycles or eliminating the need for maintenance altogether. The specification should be critiqued for quick serviceability and ease of access. Until the specification is thoroughly written and approved, no design work should begin.

The "shall not" specification is not limited to failures. That is too simple. We must be able to see the complexity in this simplicity. This is called interconnectedness. We need to know that reliability is intertwined with many elements of life cycle costs. The costs of downtime, repairs, preventive maintenance, amount of logistics support required, safety, diagnostics, and serviceability are dependent upon the level of reliability. It is wrong to measure reliability in terms of failure rates. Reliability should be measured by reduction in life cycle costs. When reliability is high, the cost of downtime and repair is low. When I was in charge of the reliability of the Baltimore rapid transit train system design, we measured it in terms of cost per track mile. Similarly, the electric utilities measure it in terms of cost per circuit mile. Smart customers look for only one performance feature—the life cycle cost per unit of use. Those who approve the specification should concentrate on this measure.

Design Out for Sneak Failures

There is one analysis that is often ignored. It aims at failures that are difficult to predict. The sudden acceleration experienced by Audi 5000 when put in reverse gear is a typical example (took place years back). It can be as simple as a pin catching road debris on the exhaust pipes on Nissan Ultima, which could catch fire from the heat of the engine. This analysis is informally called "Unknown Unknowns Analysis". It should be performed before approving the preliminary design. One tool for this kind of analysis is called "Sneak Path Analysis". In electronics and software, it is called "Sneak Circuit Analysis". This is used for discovering hidden problems, which usually turn up in rare events, such as deployment of air bags, or when there is a major accident in which a fireman may come in contact with high voltage battery terminals. Questions are asked, such as "Will the air bag open when it is supposed to?" "Will it open at the wrong time?" "Will the system give a false warning?" Or, "Will the system behave failsafe in the event of an unknown fault?" Depending on how critical the functions are, you can use other types of analyses, such as event tree analysis, brainstorming, worst case analysis, hazard analysis and diagnostics capability analysis to discover unexpected failures. A very informal but powerful method is collecting anonymous data in which the identity of the person supplying the data is not revealed without consent. For example, when Baltimore mass transit system was designed, very few sneak problems were known. Upon interviewing the train drivers, repair crew, and system users, several hundred sneak problems were identified. The following data on a major airline, announced at a FAA/NASA workshop,[2] shows the power of this technique:

- Problems reported confidentially: ≈13,000
- Number actually in airline files: ≈2%
- Number known to FAA: ≈1%

The sneak failures are more likely to be in the embedded software where practically no reliability analysis is done. Frequently the specifications are faulty because they are not derived from the system performance specification. Peter Neumann, a computer scientist at SRI International, highlights the nature of damage from software defects:[3]

- Wrecked a European satellite launch.
- Delayed the opening of the new Denver airport by one year.
- Destroyed NASA Mars mission.
- Killed four marines in a helicopter crash.
- Induced a U.S. Navy ship to destroy an airliner.
- Shut down ambulance systems in London leading to several deaths.

As a precaution, we should perform FMEA on all critical software functions, develop reusable modules, make sure it cannot accept unreasonable input, and define the product behavior in case of an unreasonable input. Then we should pay special attention to software structure and architecture so that engineering changes are quick and do not require complete regression testing. The most preferred structure is the top-down structure where the code is partitioned functionally and the order of execution flows from top to bottom. The fault tolerant architecture is the most frequently used architecture.

Reliability Must Be in the Manufacturing Process Definition

We will consider an automotive example here. The design goal should be to make the components last at least as long as the vehicle. However, the components should not fail prematurely. Such a failure is highly undesirable, even if it does not stop the mission. It creates extra downtime and costs for

the customer, which were not in the implied contract. We may accept the risk of such a failure as long as the number of customers annoyed is insignificant. Usually premature failures are due to unreliable processes, overlooked facts, omissions in manufacturing, and mismatching of mating components. The purpose of the Six-Sigma program is to avoid producing components at the tail end of tolerances because they often result in mismatching tolerances.

If we go to the root causes of failures, we will find *Early Failures* and *Super Early Failures*. The failures that occur within a few days of use are super early failures. Products that are dead on arrival (DOA) are good examples. These are sure to upset customers. More than 95% of the time, they are due to lack of manufacturing control, such as assembling a wrong part, loose connections, improper torque, and improperly aligned assemblies.

Early failures are usually those that occur in the first three years of life for automobiles. These can be either from manufacturing variations or mismatching engineering tolerances of components. Some are due to the marginal strength of the interfaces, such as from loosening of joints, degradation of seals, and weakening of soldered joints. Customers are still concerned. There is no faith in a salesman who brags that the problem was solved and hits the customer six months later with a new problem. The remedy is to make sure the Process FMEA demonstrates full control on all critical design features.

To make sure manufacturing can control the cause of early and super early failures, the process definition must include process reliability for all the critical design features. These features are found in the potential causes of failures in the design FMEA and the process FMEA although some homework is required to identify such things as critical dimensions, hardness, a chemical property or a heat treatment measure.

Design for Cheaper, Better, and Faster Products

Companies erroneously believe that making products with higher reliability costs more money. Actually the opposite is true, if you really understand reliability. Phil Crosby's book "Quality is Free" proves this point. The computer industry has demonstrated the truth of this. Computers have become steadily more reliable while also becoming less expensive. It does take more up-front effort but it drastically reduces the time it takes to put out fires. Once, at Honeywell, a product was redesigned for higher reliability. This increased the cost of components by about 5%, but the product passed the reliability test seven months early. The company saved seven months of engineering time and seven months of testing time. The customer was pleased to get a highly reliable product ahead of schedule. The resulting savings in warranty costs for Honeywell were much higher than the investment in components.

Another success story happened at Ford. When the 1995 Lincoln Continental was evaluated for internal and external interfaces and other issues, the engineers made over 700 changes in the specification. This resulted in a reduction of the manufacturing costs from $90 million down to $30 million. The project started late but finished four months early. Only those who have accomplished such results know that making a product cheaper, better, and faster is an art as well as the science.

If you want to be able to offer longer warranties to gain market leadership (the highest reliability in the industry at the lowest cost), then you need to treat reliability like a new product. In fact, it is a product. It is a deliverable item to the customer, and the customer can tell if you delivered it or not.

The secret is a good performance specification. When everything is defined in the specification from the customer's points of

view, the detailed planning resembles that of a production of a symphony orchestra at the Kennedy Center. The script, the system performance specification, must be thoroughly reviewed and different parts of work must be performed in tandem. The chief design engineer is equivalent to the orchestra director, the project manager is equivalent to the executive producer, and team leaders are equivalent to assistant producers and directors. As soon as the specification is approved, a number of activities can be performed on parallel tracks—the design FMEA, the process FMEA, the use/misuse FMEA, serviceability fault trees to make sure the repair facility can diagnose at least 90% of the faults, safety hazard analysis, supplier qualification and qualification and process approval. It is not too early for supply chain management and manufacturing to work on improving existing processes and components while the design is still on final approval stage. If they wait to do this just before going into production, it will be impossible to achieve reliability. Then the project manager is the one responsible for the failure. Of course, these analyses have to be done right. The usual problem is that often there is no one responsible to review the quality of the work. In software development, they use independent verification and validation teams. We need to develop such teams who can assess the thoroughness of reliability analysis and mitigation actions.

Use Intuition in Budgets

Most budgets are made after the concept approval. This is all right to start with, but as the system definition changes. The budget must change too. If you are going to make 700 changes in the specification, then the initial budget is obsolete. If everyone votes YES on the budget approval, your decision should be a no. Everyone agreeing on a solution may be a symptom of groupthink. Your disagreement may force everyone to re-examine some critical issues.

My large clients easily make at least 400 design changes before the final design review. We need the power of intuition to foresee unknown unknown problems. You require a new budget, a new schedule, and a new contingency budget. Most companies, since they write poor specifications, do not see this problem until much later. They keep adding one requirement at a time and fail to adjust the budget. They end up creating special budgets later and spend several times more than they would have, had the specification been done right the first time. We have at least 100 automobile recalls a year. Some of these recalls cost billions of dollars. Once I was at a company meeting. The supplier asked the customer to describe the warranty they wish to have. One of them said (and others agreed): No warranty is the best warranty. I understood the paradox—the best warranty is the one where there is no need to file a warranty claim. In other words, a failure-free product! Badger Meter has been offering a 25-year warranty on residential water meters for over two decades because they have hardly any failures in 25 years.

There are three kinds of organizations. Those that make things happen, those that watch things happen, and those that wonder what happened. If we write holistic specifications and perform thorough analysis, then we don't have to wonder what happened. We make things happen our way.

References

1. Raheja, Dev, and Allocco, Michael, Assurance Technologies Principles and Practices, Wiley, 2006.
2. Speech by Dr. Douglas R. Farrow, Fifth International Workshop on Risk Analysis and Performance Measurement in Aviation sponsored by FAA and NASA, Baltimore, August 19-21, 2003.
3. Mann, Charles, C., Technology Review, MIT, July/August, 2002.

Misperceptions of Systems Engineering: Availability & Reliability

Ron Carson, Ph.D., ESEP

Should system architects be in the business of specifying system availability and reliability? What happens if we don't? We have already addressed the problem of managing system responses to failure in a previous article. In this article we address the need to define, up-front, the system availability and mission reliability: will the system reliably do what it's supposed to do?

System availability simply means the percentage of time the system is available to perform its intended functions with requisite performance under specified conditions. Failures and down-time for maintenance mean the system is unavailable. We can make a high-availability system by ensuring no failures (high reliability—a low probability of failure for a specified period of time under specified conditions) and little down-time for maintenance, or we can ensure that any failures are quickly addressed with rapid repair or replacement. Both reliability and maintainability contribute to availability.

It is sometimes the case that reliability requirements are considered to be (a) tradeable and likely to change or (b) solely allocated hardware. The latter situation is especially ubiquitous in software thinking because "software doesn't fail" (we may address this in a future article). The reality is that any software loss of function, even a re-boot, constitutes system non-availability, whether attributable to "failure" or some other label. And if such non-availability occurs during mission operations it constitutes a system failure.

The former situation can arise when the desired system reliability is eventually compared to current technology based on the current

system architecture (e.g., specified redundancy). Too often the acquirer or sponsor is forced to accept less-then-needed system reliability because "it's the best we can do" given the architecture and technology choices. The requirements are then revised accordingly to prove "compliance". But the user does not receive what is needed in terms of mission reliability. What is the responsibility of the system architect?

First, the system architect must work with the acquirer and other stakeholders to understand and define the system availability (% of time available) and mission reliability requirements: what rate of mission failure is the client willing to accept, with what tradeoffs in maintenance, cost, complexity, weight, power, etc. (e.g., because of redundant systems)? In commercial airplanes for example these may include "dispatch reliability", the percentage of time the airplane will leave the gate on schedule. Some small level of system unavailability may be accepted for preventive and on-condition maintenance in order to improve mission reliability.

Second, the system architect must decompose the system availability and mission reliability requirements and allocate them among system-level and lower-level functions, including maintenance. (In the graduate course I teach this is an architecting requirement on all functions.) This helps ensure that such availability and reliability will drive design decisions for component selection and lower-tier architecture, and not be simply left to chance based on the after-the-fact measured performance of hardware and software. Such an approach enables an architect to analyze use cases or mission threads and evaluate the probability that any specific use case can be completed as defined.

Finally, it is incumbent on the architect to continuously monitor the state of the system architecture during development to ensure compliance with system-level availability and mission reliability requirements (as well as all other requirements). Deferring assessments to end-of-development "verification" is a serious abrogation of architecting responsibility, as one of the important items evaluated at technical reviews during development is the current assessment of compliance with requirements. Any deficiency in compliance (a program "issue") is grounds for corrective action. And the sooner such deficiency is discovered the sooner it can be corrected, typically with lower project cost compared with later discovery and correction.

In summary, projects should adopt a top-down allocation of the true availability and reliability requirements (based on stakeholder needs) rather than a "que sera sera" or best-effort attitude toward these considerations.

References

1. Carson, Ronald S., "Integrating Failure Modes and Effects with the System Requirements Analysis", Proceedings of the International Council on Systems Engineering (INCOSE), 2004 (Toulouse, France).
2. Mark Sampson, "Model-based Product Safety & Reliability: forward vs backward looking design..." https://community.plm.automation.siemens.com/t5/Teamcenter-Blog/Model-based-Product-Safety-amp-Reliability-forward-vs-backward/ba-p/460821
3. Ron Carson, "Misperceptions of Systems Engineering - 4: Complete Requirements", https://www.linkedin.com/pulse/misperceptions-systems-engineering-4-complete-ron-carson-phd-esep/
4. ARP5580, "Recommended Failure Modes and Effects Analysis (FMEA) Practices for Non-Automobile Applications", SAE International https://www.sae.org/standards/content/arp5580/ (8 May 2012).
5. Carson, Ronald S. and Bojan Zlicaric, "Using Performance-Based Earned Value® for Measuring Systems Engineering Effectiveness", Proceedings of the International Council on Systems Engineering (INCOSE), 2008 (Utrecht, The Netherlands).

Published on December 18, 2018 on Linkedin: https://www.linkedin.com/pulse/misperceptions-systems-engineering-7-availability-ron/

This is the seventh in a series of short articles: https://www.linkedin.com/in/ron-carson-phd-esep-573549b

Improving Reliability Throughout the Product Life Cycle

Christopher Laplante

Within the stages of a product lifecycle, reliability tasks are performed to maintain and improve product quality while also helping engineer customer trust in the brand. This paper discusses the integration and importance of accelerated life techniques into different phases of the life cycle, with a focus on Highly Accelerated Life Testing (HALT) and Highly Accelerated Stress Screens (HASS).

The idea of accelerated life testing is to apply stresses to a product in order to precipitate and identify product weakness. This provides an opportunity to eliminate design weaknesses in development and prevent process-related weaknesses from reaching the field.

Definition of HALT and HASS

HALT is not a pass or fail test, but a developmental tool used to understand weaknesses and limitations of a product in a short period of time. This is done by applying accelerated stress techniques such as extreme temperatures, rapid thermal transitions, six degrees of freedom repetitive shock vibration, and a combined environment of these stresses. The goal is to stress until failures are identified, indicating operating and destruct limits. The nature of these failures can be recoverable, non-recoverable, or intermittent failures. With this information, failure modes should be analyzed for root cause and eventually lead to corrective actions that improve the product by eliminating identified weaknesses.[4]

Environmental Stress Screening (ESS) has been an essential part of production processes for many years. The goal is to re-

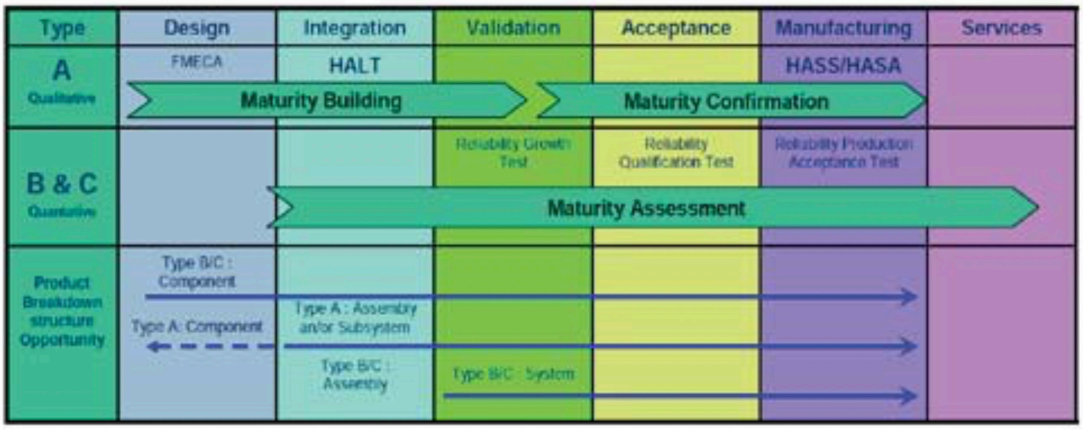

Figure 1. IEC 62506 Methods for Product Accelerated Testing

move latent defects from the product prior to field delivery. Within this concept, it is understood that the greater the stress that is applied, the more efficient the process is in identifying the failures.[5] Though in traditional ESS, the stress applied is typically much lower than known limitations of the product and is therefore a conservative approach. HASS is an accelerated ESS, applying higher stresses in order to more efficiently discover the same latent defects. HASS is performed on 100% of units, though a selection-based process known as Highly Accelerated Stress Audit (HASA) can also be applied. The goals for either remain the same, to identify individual units that may be weaker due to process variability. The stresses applied during HASS are based on the results previously identified in HALT. With this process, units that are marginal and will potentially fail early in life will be identified and prevented from reaching the field, reducing warranty cost and improving product quality. [4]

Accelerated Testing Integration Timeline

On a basic level, HALT is identified as a process used in design and HASS is a process used in production. Figure 1 clearly shows HALT occurring as soon as possible in the design phase (as early as valid prototypes are available) and HASS as part of the manufacturing process.[6] This paper takes this concept and expands on it, using HALT and HASS as not just one-time events, but tools that can be applied throughout product life to help with product reliability and quality.

In order to effectively integrate HALT and HASS into different phases of a product's life cycle, it is first important to understand each individual phase. This paper breaks out the life cycle process into eight main phases. They include:

- **Design.** The first phase of a product's life in which the product is developed to meet specific requirements.
- **Pilot Build.** The first physical build of the product providing an opportunity to test processes.
- **Production.** Products are built specifically for customers to purchase and use in the field.
- **Contract Manufacture.** Outside source contracted to produce and deliver product.
- **Material Receiving.** Parts used in product that are not designed in-house, but purchased from outside vendors.
- **Quality Audit.** An audit of existing or new vendors in order to approve of a vendor or validate existing process.
- **Field Return/Depot Repair.** Products that are returned due to field failure.
- **Engineering Change Order.** A design change that may be the result of field failures or obsolescence.

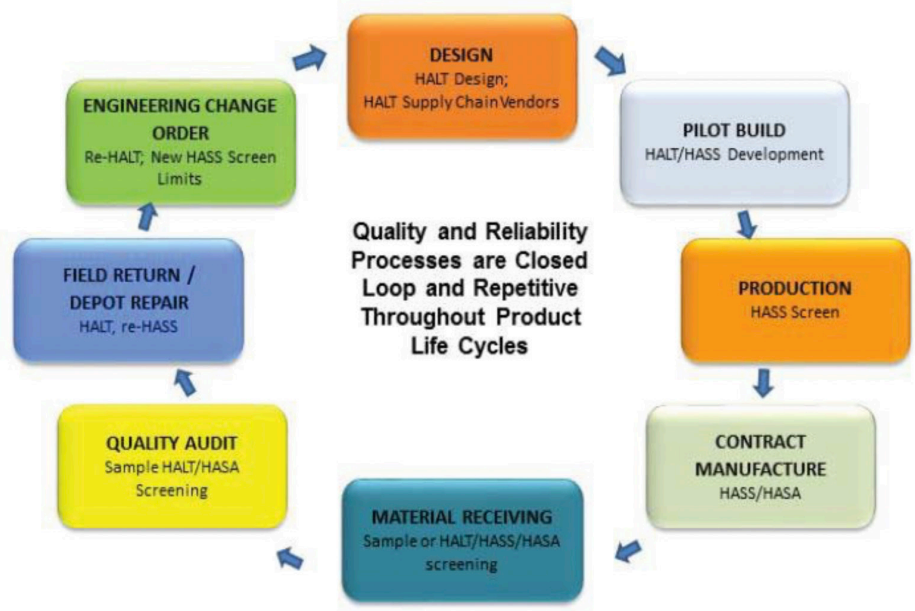

Figure 2. Product Life Cycle

Interactions of HALT/HASS and Reliability in the Product Life Cycle

HALT is an exploratory process, providing qualitative information about a product. It does not provide information that can be directly interpreted in terms of product life predictions.[7] Consequently, it is sometimes difficult for a Reliability Engineer to see what value it can bring to the highly quantitative tasks they are faced with in product development. However, the failure mode and limit information provided by HALT can be used to improve the results from a Reliability Program.

The integration of HALT and HASS into the life cycle offers a give and take process in which the standard reliability tasks and the HALT/HASS tasks provide useful information for each other. The limits identified in HALT will help define more effective testing profiles, while HASS will consistently validate that there is no drift in the product or processes. HALT is not intended to replace other reliability tasks, however HASS can be seen as an accelerated form of ESS which may already exist in the production process.

Design

Typically, a reliability prediction will be completed in the design phase in order to determine a theoretical Mean Time Between Failure (MTBF). Based on the designs and reliability prediction, a Failure Mode Effects Analysis (FMEA) is completed to identify possible failures and prioritize them in order of importance to the product. These, among other tasks, provide an opportunity in design for reliability engineering to take place.[8] With the addition of HALT, these processes can be streamlined and the data collected (failure modes, operating and destruct limits) can become valuable assets towards improving the product.

Pilot Build

Once the product has gone through a majority of its research and development, the product is built for the first time. This build can then be used for additional reliability testing as well as testing manufacturing processes. Limits found in HALT can help define limits to be used in qualification tests such as Accelerated Life Testing (ALT), resulting in a more efficient testing process.[9] HALT results before and after corrective actions can affect infant mortality estimates

and improve warranty predictions. Parameter estimates can be used to determine effectiveness of corrective actions. Evaluation of results from Reliability Growth Testing (RGT) or ALT can determine if HALT should be redone or modified to capture missed failure modes.

Production

By the time a product reaches the production phase, the major issues found through development should have been addressed in order to improve the design. Now HASS is fully implemented on 100% of production units to monitor for manufacturing errors due to process variability. With this implementation, the use of more conservative screening techniques such as burn-in, thermal cycling or other ESS isn't necessary.[10]

Contract Manufacture

Not all production is performed in-house. Manufacturing is sometimes outsourced to a contract manufacturer (CM) who provides all the resources necessary to implement the production process of a product. When CMs are utilized, the owner of the product controls the design, but not the manufacturing process. This can offer challenges such as maintaining quality control of the production process, as well as process variability. These challenges can be addressed by implementing a HASS process at the CM level. The HASS is developed by the product owner, but is performed by the CM at the production facility.

Material Receiving

In order to manufacture a new product, update a design or simply repair a failed product, materials are required. Often materials are purchased from vendors and therefore are not manufactured in house. It is important upon receiving new materials to be aware of the durability of the product. Counterfeit components, which have become a growing problem across industries, can also compromise the integrity and quality of products.[11] All it takes is one component or subassembly not meeting standards to take down an entire system. Data from incoming sampling can identify the need for a HALT to quickly evaluate a possible deviation. Failure information from HALT can be used to identify likely failure modes to focus inspection. Margin information from HALT can add confidence to decisions to reduce screening levels.

Quality Audit

A quality audit can validate existing or new vendors in order to approve of a vendor and its products. HALT provides another data point in the decision process on the need for an audit of an existing vendor, or the approval of a new vendor. Statistical Process Control (SPC) data can trigger change in sample size or screen strength. A shift in HASS failure rates can trigger a need to audit an existing vendor or validate a new vendor. Data from HALT can give early input to aid in reliability analysis or audit data. Parameter drift in HASS can influence SPC data choices and limits.

Field Return/Depot Repair

Changes in warranty returns or deviations from predicted field reliability should be debugged, and HALT speeds up that process.[12] HASS performed on returned units may force intermittent failures to fail hard and help in failure analysis. Similarly, if HASS is done on returned units that are NFF (No Failures Found) and the units pass, likely they are returned in error. By determining this quickly and efficiently, these units can be removed from reliability calculations in order to improve accuracy.

Engineering Change Order

Field failures and obsolescence will often initiate a change in the product's design. If change results in a significant shift in calculated reliability or parameter estimates show significant change, this can trigger a requirement for a new HALT before the Engineering Change Order (ECO) is approved. A HALT can also be brought upon if the FMEA indicates possible significant failure effects of change. If a change results in a significant shift in calculated reliability, a HALT can validate changes and determine if new calculations are warranted.

Conclusion

HALT, HASS, and all other reliability engineering tasks share a common goal of reducing warranty expenses by improving and maintaining quality of a product. Product weakness information gathered from any of these sources should be shared to meet this common goal. These techniques, applied throughout a product's life, are an effective method for improving reliability, maintaining quality, and streamlining reliability processes.

The following work has been abridged to meet copyright requirements. The work was first published in the 2018 IEEE Reliability and Maintainability Symposium (RAMS). It is used here with permission from the IEEE: http://ieeexplore.ieee.org/document/7889654/
References have been removed to meet copyright issues but are available on the IEEE website: https://ieeexplore.ieee.org/document/8463120/

About the Authors

A System Design Method to Reduce Cable Failure Propagation Probability in Cable Bundles

Dr. Douglas L. Van Bossuyt is currently an Assistant Professor in the Systems Engineering Department at the Naval Postgraduate School (NPS). He holds a PhD in Mechanical Engineering, a Master's of Science in Mechanical Engineering, an Honors Bachelors of Science in Mechanical Engineering, and an Honors Bachelors of Arts in International Studies from Oregon State University. His research interests lay at the intersection of risk and failure analysis, systems engineering and design, manufacturing, and operation of complex systems such as defense systems, nuclear reactors, and aerospace systems. Dr. Van Bossuyt is a member of the American Society of Mechanical Engineers (ASME) and the Prognostics and Health Management Society (PHM Society), and regularly attends the International Design Engineering Technical Conference (IDETC), the PHM Society Conference, and the Reliability and Maintainability Symposium (RAMS).

Dr. Bryan O'Halloran is currently an Assistant Professor in the Systems Engineering (SE) department at the Naval Postgraduate School (NPS). Prior to joining NPS, he was a Senior Reliability and Systems Safety Engineer at Raytheon Missile Systems and the Lead Reliability and Safety Engineer for hypersonic missile programs. He holds a Bachelor of Science degree in Engineering Physics and a Master of Science and Doctorate of Philosophy in Mechanical Engineering from Oregon State University. His current research interests include risk, reliability, safety, and failure modeling in the early design of

Complex, Cyber-Physical Systems (CCPSs). He is a member of the American Society of Mechanical Engineers (ASME) and the Institute of Electrical and Electronics Engineers (IEEE) and regularly attends the International Design Engineering Technical Conference (IDETC), the International Mechanical Engineering Congress and Exposition (IMECE), and the Reliability and Maintainability Symposium (RAMS).

Dr. Nikolaos Papakonstantinou has a diploma in Electrical & Computer Engineering from the University of Patras (Greece) and a doctorate degree in Information Technology in Automation from Aalto University (Finland). Currently he works as a senior scientist at VTT Technical Research Centre of Finland in the area of system modeling and simulations. He focuses on simulation, model and data driven approaches to system design, operation and safety assessment. Even before moving to VTT, as a post-doctoral researcher at Aalto University, he focused on simulation based safety assessment of complex systems using case studies from the nuclear power production industry. He managed the IFAPROBE project, part of the Finnish Research Programme on Nuclear Power Plant Safety and was the responsible teacher for the "Managing the product life cycle" master level course. His earlier research was in the area of automation software design, mainly targeting IEC61131 and IEC61499 based controllers, with applications on machine, batch and continuous process automation control.

Reliability is More Art Than Science

Dev Raheja, MS, CSP (Certified Safety Professional) has been a Quality Management, System Reliability Engineering and System Safety consultant for over 25 years, His range of consulting and training encompasses automotive industry, defense systems, transportation systems, electric power systems, high tech industry, aerospace, nuclear , medical systems, and consumer products. He has conducted training in several countries including Sweden, Australia, Japan, UK, Belgium, Finland,Turkey, Germany, Poland, Singapore, Brazil, South Africa, and Canada. He has done training and consulting work with NASA, U.S. Army, U.S. Navy, U.S. Air Force, GM, Ford, Boeing, Eaton, Nissan Aerospace, Litton, General Dynamics, ITT, BAE Systems, Lockheed-Martin, Harley-Davidson, and United Technologies.

Prior to consulting, Dev Raheja worked at General Electric (Manager of Manufacturing), Cooper Industries (Chief Engineer), and at Booz-Allen & Hamilton (Senior Consultant). He is the author of several books including Assurance Technologies Principles and Practices (Second Edition, Wiley 2006), Zen and the Art of Breakthrough Quality Management, and Design for Reliability (Wiley, 2012). He has taught several courses for the American Society for Quality and was their representative judge on the selection of the first National Malcolm Baldridge Award. He has a received Scientific Achievement Award and the Educator-of the-Year Award from the International System Safety Society.

He has conducted training courses at several universities including University of Maryland and at Florida Tech University. He is a member of SAE G48 Committee and the Fellow of the American Society for Quality.

Misperceptions of Systems Engineering: Availability & Reliability

Dr. Ron Carson is an Adjunct Professor of Engineering at Seattle Pacific University, an Affiliate Assistant Professor in Industrial and Systems Engineering at the University of Washington, a Fellow of the International Council on Systems Engineering and a certified Expert Systems Engineering Professional (ESEP®).

He retired in 2015 as a Technical Fellow in Systems Engineering after 27 years at The Boeing Company. He is the author of numerous articles regarding requirements analysis and systems engineering measurement, and is the developer of numerous industry systems engineering training courses. He has been issued six US patents in satellite communications, and two patents regarding "Structured Requirements Generation and Assessment". His current interests are in quantitatively incorporating sustainability considerations in systems engineering methodologies and education. Dr. Carson has a PhD from the University of Washington in Nuclear Engineering (Experimental Plasma Physics), and a BS from the California Institute of Technology in Applied Physics.

Improving Reliability Throughout the Product Life Cycle

Christopher Laplante is a Reliability Engineer at Qualmark Corporation in Denver, Colorado. He has been working as a Reliability Engineer for over ten years, first in analysis then most recently in applications, technical support and training of HALT and HASS. He holds a BSEE from Western New England College in Springfield, Massachusetts.

Colophon

The Journal of Reliability, Maintainability, & Supportability in Systems Engineering

Editor-in-Chief: John E. Blyler
Managing Editor: Russell A. Vacante, Ph.D.
Production Editor: Phillip S. Hess

Office of Publication: 9461 Shevlin Court, Nokesville, VA 20181
ISSN: 1931-681x

Instructions for Potential Authors

The Journal of Reliability, Maintainability and Supportability in Systems Engineering is an electronic publication provided under the auspices of the RMS Partnership, Inc. on a semi-annual basis. It is a refereed journal dedicated to providing an early-on, holistic perspective regarding the role that reliability, maintainability, and supportability (logistics) provide during the total life cycle of equipment and systems. All articles are reviewed by representative experts from industry, academia, and government whose primary interest is applied engineering and technology. The editorial board of the RMS Partnership has exclusive authority to publish or not publish an article submitted by one or more authors. Payment for articles by the RMS Partnership, the editors, or the staff is prohibited. Advertising in the journals is not accepted; however, advertising on the RMS Partnership web site, when appropriate, is acceptable.

All articles and accompanying material submitted to the RMS Partnership for consideration become the property of the RMS Partnership and will not be returned. The RMS Partnership reserves the rights to edit articles for clarity, style, and length. The edited copy is cleared with the author before publication. The technical merit and accuracy of the articles contained in this journal are not attributable to the RMS Partnership and should be evaluated independently by each reader.

Articles should be submitted as Microsoft Word files. Articles should be 2,000 to 3,000 words in length. Please use ONE space after periods for ease of formating for the final publication. Article photos and graphics should be submitted as individual files (not embedded into the article or all into the same file) with references provided in the article to their location. Charts and graphics should be submitted as PowerPoint files or in JPEG, TIFF, or GIF format. Photos should be submitted in JPEG, TIFF, or GIF format. All captions should be clearly labeled and all material, photos included, used from other than the original source should be provided with a release statement. All JPEG, TIFF, or GIF files must be sized to print at approximately 3 inches x 5 inches with a minimum resolution of 300 pixels per inch. Please also submit a 100-125 word author biography and a portrait if available. Contact the editor-in-chief, John Blyler, at j.blyler@ieee.org for additional guidance.

Please submit proposed articles by October 1 for the Spring/Summer issue of the following year and April 1 for the Fall/Winter issue of the same year.

Permission to reproduce by photocopy or other means is at the discretion of the RMS Partnership. Requests to copy a particular article are to be addressed to the Managing Editor, Russell Vacante at president@rmspartnership.org.

The Journal of Reliability, Maintainability, and Supportability in Systems Engineering

Winter 2018–19