System Robustness Achieved with System Engineering Methodologies

David G. Mabee
Heidelberg Digital L.L.C.
Rochester, New York

Abstract

System engineering processes and methodologies are employed in the development of printing system robustness. Under the overarching umbrella of six sigma concepts, system robustness can be defined, developed, tested, measured, and verified. Designing robustness into a product at the systems level requires that the important system attributes are focused upon and made robust beginning in the subsystem development stage. Robustness and critical customer requirements are translated into physical functions, linked to contributing system and subsystem functions, and traced to critical dimensions on the parts. Six-sigma methodology as applied to parts manufacturing is not sufficient by itself. The application of the six sigma methodology at the system design level requires great discipline which is achieved by seemingly disconnected processes becoming linked under the umbrella of six sigma. These processes include require-ments management, FMECA's, problem identification and corrective action, DFx, Taguchi experiments applied at the system level, configuration management, additional systems test processes, noise maps, and limits definition. These elements are examined in a linked process that, applied with discipline and tenacity, results in a well-defined system, a well designed system, and a system that is verified as robust.

Introduction

The development of any complex technological product always requires very specific technology expertise and focus. Even with expert knowledge and focus on product requirements, solid product ideas targeted at very valid customer needs are often plagued in the development cycle. This is not due to poor insight from the technology experts, but rather from the lack of broad-based system-oriented personnel and processes that can transcend and link the complex product and organizational interactions. Technology and subsystem experts have the tendency to build a product brick by brick, layer by layer without knowledge or concern of interactions with other technologies or subsystems. This results in project delays, costly overruns, and often times products introduced which do not function well or deliver the value expected by the customer. Such projects are frustrating for all involved, stress levels are high, and the size and complexity of the problems are largely unknown.

The processes and methodologies discussed are a solution that provides a top down development approach that complements the brick by brick approach. As with subsystems development, these methodologies must be linked in order to benefit the commercialization process. A project using any of these methodologies independently in an unlinked fashion will not achieve the full potential and benefits of system engineering.

Often times it is difficult for technology and subsystem experts to support and buy into these processes because they are viewed as additional work, interference from outsiders, exposure of their weaknesses and problems, etc. The benefit of these linked processes are much like compound interest, daily investments in the form of small non-complex contributions that add up over time and deliver more than the original investment. Early in the project the benefits won't be realized or be visible and yet the focus of the processes is early in the development cycle. The return is in the later stages of the development cycle, and more importantly in the production of the product and beyond. Each one of the processes presented could be a paper in itself. Instead, a broad brush of each process is presented and the discussion will focus on the feeding and linking of the processes across the product being developed and the organization developing it.

The processes in themselves are not complex, thus their value is often overlooked. What is complex is the behavioral changes which must occur in the organization, and the tenacity and patience required to see through to the benefits. As a result, any organization attempting such an approach must have a systems expert and team who are supported by the upper management levels. This group will need to do many things that are not popular or understood by the subsystems experts. Their efforts will be resisted and occasionally personal rejection will occur. Without upper management support and backing, success is not possible.

Threads

The thread tying these processes together is six sigma. Six sigma is as much a mindset and behavior as it is a tool to develop and understand how well a product functions. Six sigma is not only about parts. It is about making sure that the output of the process, in this case a printed page, satisfies customer requirements for image quality almost all the time, as shown in Figure 1.

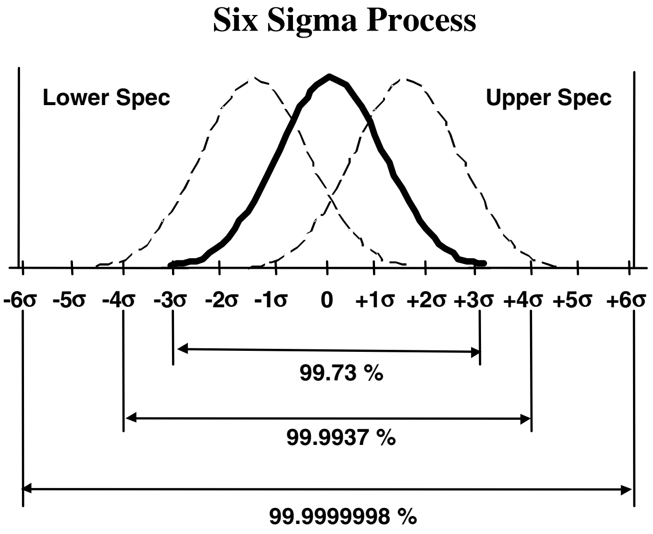

Figure 1. In a six sigma process, variation is controlled so that the output almost never falls outside the spec limits.

Six sigma also provides a structure for key project metrics. Intertwined and enabled by the mindset is a ruthless system of documenting every problem found in development and testing no matter how minor it may seem. The mindset of six sigma is to strive for perfection.

Specific Tools and Practices

The specific tools and practices supported by the overarching umbrella of a six sigma mentality include requirements management, FMECA's, noise mapping, stress testing, limit definition and performance evaluation, corrective action methodology, and configuration management. There are certainly other processes and methodologies which tie into this umbrella, or could tie into it. The processes used are going to be somewhat organization dependent as well as project dependent. A short, simple project leveraged from a successful existing design may not require all these processes and probably can be achieved with less rigor. A from-the-ground-up project must include all these processes and probably some not discussed.

Even though six sigma tends toward perfection, physical reality is that no product is ever perfect. To complement these processes, a set of measurements such as process capability growth and reliability growth rate must be used by upper management to judge the progress, the risks, and to determine when the product is good enough.

Project planning, although not discussed in this paper, is also critical to the timely and cost effective commercialization of any complex system. The planning must be thorough, detailed, and linked from the upper program plan right down to every subsystem and system plan. Metrics for conformance to plan must also be used.

Linking Key Processes

The evolution of product requirements during the development process is reflected in the phases shown in Figure 2. Phase 1 is concerned with subsystem testing and establishing an initial system design. Phase 2 focuses on the integration of subsystems into a larger system robustness activity. Phase 3 is testing of engineering models, (EM's) that are near the production configuration. Phase 4 is verification of the final product design, production-like parts, and manufacturing methods.

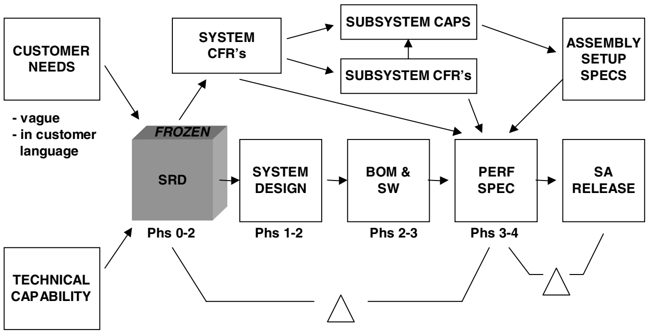

Figure 2. Phases of product development from the System Requirements Document to Shipping Approval.

The system requirements document (SRD) is the result of reducing customer needs to critical function requirements (CFR's). The CFR's are mapped down to the level of critical to function (CTF) part dimensions and setpoints. Process capability for each CFR is determined using experimental design and noises. Each CFR must be determined as robust by testing at limits of the contributing subsystem responses, adjustments, and part dimensions. Data is reported in six sigma terms as a Cpk.

Initiating problem capture, a corrective action methodology, and strict configuration management in the first phase and rigorously following these disciplines throughout phases 1-3 requires a large initial investment in time, discipline, and resources. This up-front effort eliminates problems that are much more expensive to fix in latter stages of product development as integration occurs and tooling dollars are been committed, as shown in Figure 3.

The problem reporting process that initiates corrective action is shown in Figure 4. It is very effective to use a database to record problems and manage the corrective actions. Metrics evolving from a thorough problem documenting process provide many insights into the project's areas of risks, resource shortages, time delays, and cost overruns.

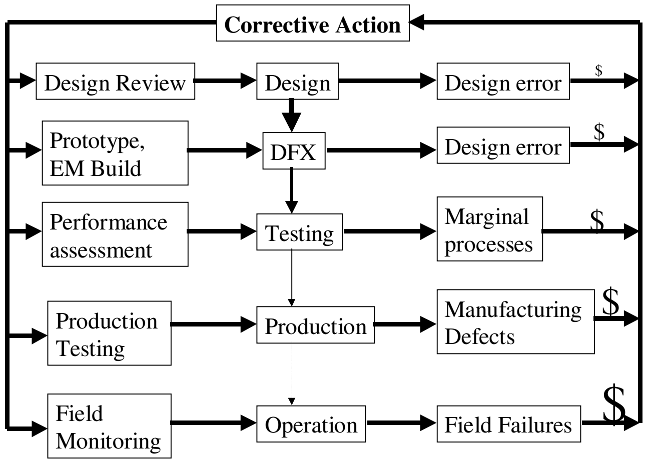

Figure 3. Corrective action is much less expensive during the initial stages of product development than during production of in the field.

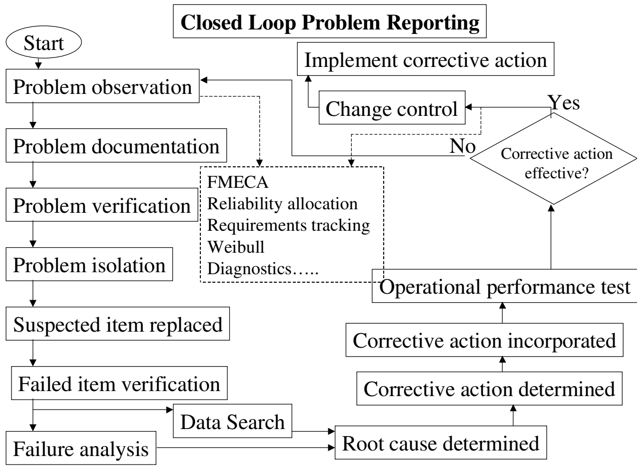

Figure 4. A well-established problem reporting process based on product requirements is needed to implement corrective actions. Problems are reported by email to the relevant personnel and can trigger corrective actions. The problem and its resolution are recorded in a database.

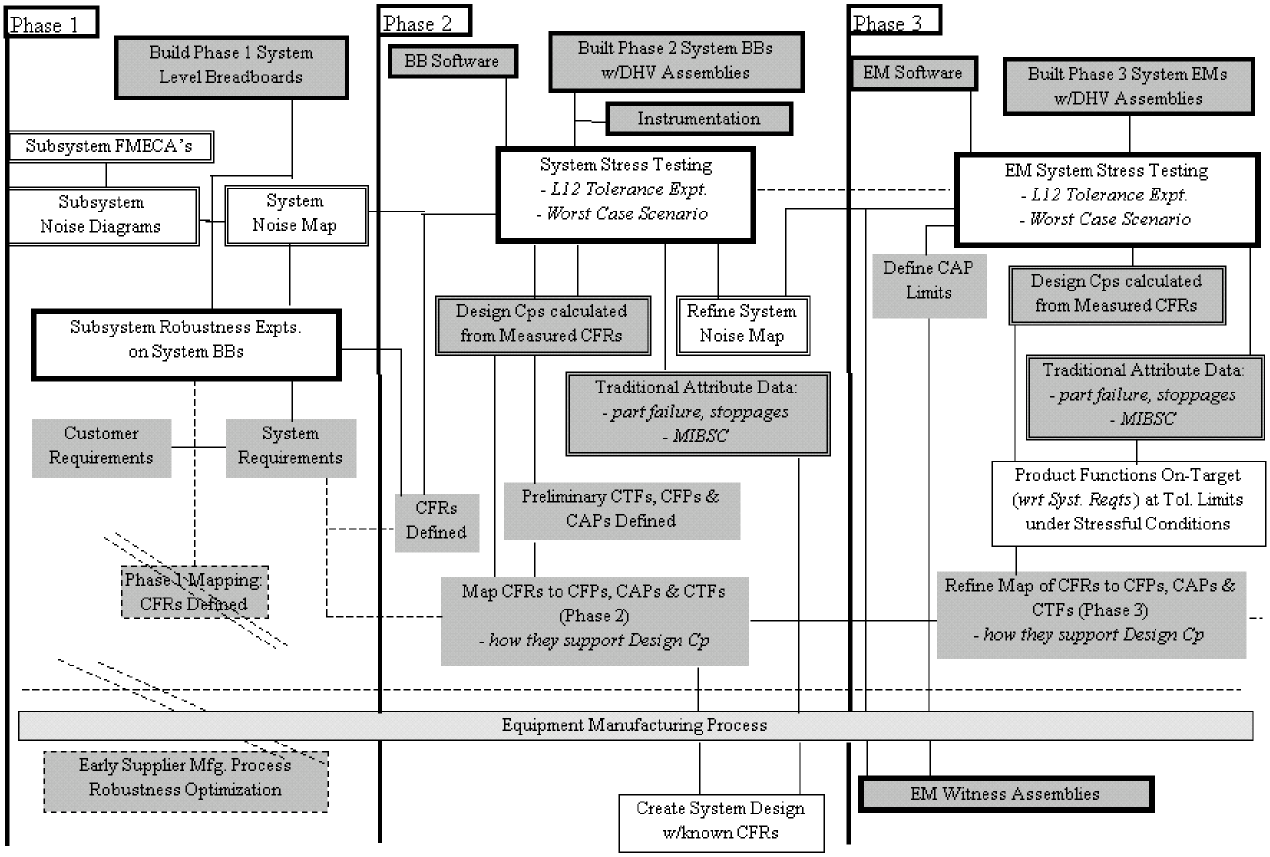

Figure 5. The first three phases of the product development cycle. One of the major components of system testing is the system noise map. This is derived from the noise map for each subsystem by identifying where the noises go. This information is used to robustize the affected subsystems. Phases 1 and 2 are used to determine limits for the subsystems and the systems. Phase 3 is used for testing the system as a whole under stress conditions near failure limits rather than at nominal. This controlled introduction of "noises" allows system reliability to be measured effectively with relatively few units under test.

Figure 5 is a more detailed look at phases 1-3 that shows the transition from subsystem design activities to system design activities. Phase 1 contains failure mode analysis (FMECA) and noise characterization at the subsystem level from subsystem robustness experiments.

Noise mapping from the subsystems to the system level is a crucial function for system design. Noise maps play a critical role in systems level experimentation. The subsystem noise diagrams (Figure 6) primarily identify the source of noises. Each subsystem noise map is used to identify the prominent output noises and these are mapped in the system noise map to where they are initially believed to be going. We then verify through subsystem experiments and noise experiments what the prominent noises are in the system and what subsystems they affect.

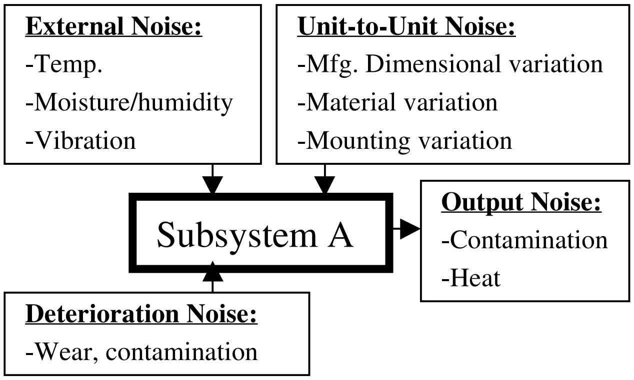

Figure 6. A generic subsystem noise map. Noises are identified by source.

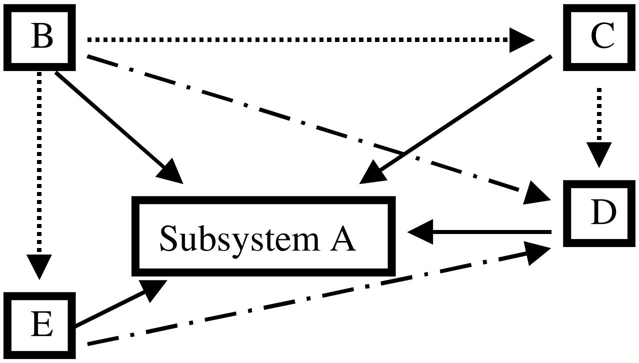

Figure 7. A generic system noise map for subsystems A-E.. Status is usually indicated by colored arrows. Solid or green arrows= acceptability. Dotted or red arrows= performance does not meet criteria and that robustness issues exist. Alternating dots and dashes or yellow arrows = nominal performance meets criteria but that robustness issues remain to be resolved.

In the system noise map (Figure 7) all the noises from each of the subsystems and the operating environment are mapped and the status of each interaction is indicated as acceptable, marginal, or problematic by color code. In order for a subsystem to be green it must be robust to the noises that are coming into it and it cannot send out harmful noises to the rest of the system. Corrective actions are implemented into the system under strict configuration control. Configuration control during the system design stage must be as rigourous as it is in a production environment. using DFx principles, meaning design for anything: reliability, manufacturing, service, and remanufacturing. Under DFx principles, for example, a part that functioned perfectly would be redesigned if it could be installed incorrectly.

After noises are identified and limits are established, robustness is verified by testing a small number of units at stress levels in noisy conditions to force failure modes and measure both design capability and assembly capability. Successful performance evaluation of a few product-like models, built and adjusted to the 1.5 sigma level to represent natural production variation, completes the system robustness and evaluation process and leads to the introduction of a robust, well-understood and well-documented product.

Conclusion

A stable product development process starts by establishing product requirements and an initial configuration at an early stage with understanding of subsystem output noises, robustness, and control factors. For system development, it is essential to implement a rigorous problem reporting process and a corrective action process with strict configuration control. This must be followed during the entire product development process. Management support is essential because significant commitment is needed to implement these measures early in the program long before the advantages or even the problems become apparent. Mapping subsystem noises onto the system components that they affect, measuring these noises, and establishing realistic noise surrogates and countermeasures allows the final design to be verified as robust by testing critical parameters at normal limits of design and manufacturing variability.

Acknowledgements

My thanks to Eric Stelter, Carla VanDeRostyn, and George Walgrove for aiding in the completion of this paper.

References

1. Karl Ulrich and Steven Eppinger, Product Design and Development, McGraw Hill, New York, NY, 2000.
2. Don Clausing, Total Quality Development, ASME Press, New York, NY, 1994.

3. Preston Smith, Developing Products in Half the Time: New Rules, New Tools, Van Nostrand Reinhold, New York, NY, 1998.
4. Seymour Morris and David Nicholis, Reliability Toolkit: Commercial Practices Edition
5. Clyde M. Creveling and William Y. Fowlkes, Engineering Methods for Robust Product Design, Addison Wesley Publishing Company, Reading, Massachusetts, 1995.
6. Benjamin S. Blanchard and Wolter J. Fabrycky, Systems Engineering and Analysis, Prentice Hall, Inc. Englewood Cliffs, NJ, 1990.

Biography

David G Mabee, Heidelberg Digital L.L.C., is the Systems Engineering Expertise Manager and a Project Development Manager for digital printers. Dave has had broad product commercialization experience including assignments in QA Product and Software Testing, Operations, Sales and Service support, Product Design, Production Engineering, and Assembly Supervision. BS in engineering (US Military Academy at West Point), and MS in Engineering and Manufacturing Management (Clarkson University).