

Systems Engineering Excellence Through Design: An Integrated Approach Based on Failure Mode Avoidance

Ioan Felician Campean and Ed Henshall
University of Bradford

Brian Rutter
Ford Motor Company

ABSTRACT

Automotive Product Development organisations are challenged with ever increasing levels of systems complexity driven by the introduction of new technologies to address environmental concerns and enhance customer satisfaction within a highly competitive and cost conscious market. The technical difficulty associated with the engineering of complex automotive systems is compounded by the increase in sophistication of the control systems needed to manage the integration of technology packages. Most automotive systems have an electro-mechanical structure with control and software features embedded within the system. The conventional methods for design analysis and synthesis are engineering discipline focused (mechanical, electrical, electronic, control, software). Multi-disciplinary systems engineering design requires a more coherent approach to the concurrent development of systems and their integration on a functional basis, supporting the flawless delivery of customer required functions in a robust and reliable manner. This paper presents an integrated framework for multi-disciplinary systems engineering design underpinned by a Failure Mode Avoidance (FMA) process that ensures coherent information flow within the product creation process. The FMA process is based on the analysis and cascade of functional requirements and associated function failure modes, along with robust countermeasure development and effective design verification. A case study, based on the development of a diesel exhaust aftertreatment system, is used to provide an illustration of the application of this framework and the associated engineering tools that underpin the FMA process within a multi-disciplinary system design context. The case study demonstrates the integration of the FMA process with the systems engineering design and product development process flow, supporting the achievement of right first time through design. The paper also includes a reflection on the practical benefits and challenges associated with the implementation of this framework on a large scale within a PD organisation.

CITATION: Campean, I., Henshall, E. and Rutter, B., "Systems Engineering Excellence Through Design: An Integrated Approach Based on Failure Mode Avoidance," *SAE Int. J. Mater. Manf.* 6(3):2013, doi:10.4271/2013-01-0595.

INTRODUCTION

The pressure for automotive OEMs to address environmental concerns and enhance customer satisfaction within a highly competitive and cost conscious market environment, places significant technical and organisational challenges on the Product Development (PD) organisations. The technical difficulty associated with the engineering of complex automotive systems is compounded by the increased sophistication of control systems deployed to manage the integration of technology packages. Most, if not all, modern automotive systems have an electro-mechanical structure with control and software features embedded within the system. Nevertheless, the conventional methods and tools that support the design analysis and synthesis tend to be

discipline focused (mechanical, electrical, electronic, control, software) and largely structural / hardware based. Multi-disciplinary systems engineering design requires a more coherent approach to the concurrent development of systems and their integration on a functional basis. The business challenge to the PD organisations is related to the need to address the effectiveness of the product design and development processes. By analogy with product manufacturing, the lack of effectiveness in the PD organisation can be judged in terms of the amount of "rework" required. In this sense, the number of PD "engineering changes" could be taken as a measure of the amount of rework conducted in PD. Research evidence [1, 2] clearly illustrate that the number and cost implications of

engineering changes are highly significant. The argument has been made [3] that engineering changes are triggered by the discovery of a design failure mode (which could be due to business or procedural decisions as well as technical / engineering decisions), which requires a design countermeasure to be developed and implemented. On this basis, the Failure Mode Avoidance (FMA) paradigm and business strategy has been introduced as a practical approach to enhance the effectiveness of product development [3, 4].

Key to the deployment of the FMA strategy within engineering design is the effective use of structured tools and processes to facilitate the discovery of failure modes and the development of countermeasures early in the design process, i.e. achieving “right first time” through design. While failure avoidance tools such as FMEA, P-Diagram, Design Verification [5, 6], are commonplace in industry practice, research [5, 7] showed some common weaknesses with the practical deployment of these tools, in particular (i) the lack of integration between different tools and in their application across the boundaries of engineering disciplines, and (ii) the lack of alignment between the deployment of the failure avoidance tools and the product development (PD) timeline, with the tools being often applied late in the PD process

More recent work [8], building on the axiomatic design principles [9], has argued that the integration of the FMA process should be on a functional basis, and underpinned by structured tools (such as the system state flow diagram and enhanced interface analysis) to support functional requirements specification within a systems engineering design context. The reasoning behind this approach is that if engineering functional requirements are comprehensively identified (i.e. including both the main functions and the interface functions required for system integration) this supports both (i) the systems engineering design process by providing a comprehensive specification of functions that need to be achieved by the system, subsystems, and components; and (ii) the failure mode avoidance process by supporting the early identification and prioritisation of function failure modes, as well as the development and verification of robust countermeasures. A focus on functional requirements also facilitates the full integration of a design across engineering disciplines.

The aim of this paper is to present an integrated framework for conducting systems engineering design underpinned by an enhanced Failure Mode Avoidance process, which has been developed by the University of Bradford Engineering Quality Improvement Centre (BEQIC) based on collaborative work with the global automotive industry over the last 15 years. The FMA process as a whole and the subsidiary tools have been described elsewhere [5,8,10]. The focus in developing this framework was on achieving the following objectives:

1). To set up a function-driven FMA process, underpinned by structured support tools which facilitate the analysis and integration of multi-disciplinary automotive systems on a functional basis;

2). To provide a coherent framework for the information flow in the product development process based on the analysis and cascade of (i) functional requirements, (ii) function failure modes, (iii) effective design verification;

3). To ensure a synergistic integration of existing failure avoidance practices and tools with new or enhanced tools, integrated with the systems engineering cascade on a practical and straight forward basis.

The main contribution of this paper is on the integration of this FMA process with the systems engineering framework for a complex multi-disciplinary system. In order to both illustrate and validate the proposed framework a case study, based on the development of a diesel exhaust aftertreatment system, will be considered. The case study provides a comprehensive illustration of the application of the framework and the associated engineering tools that underpin the FMA process within a multi-disciplinary system design context, and demonstrates the integration of the process with the systems engineering design and product development process flow, supporting the achievement of right first time through design.

METHODOLOGY

FMA Process Outline

The FMA process consists of four steps, illustrated in Figure 1, and briefly discussed in the following sections.

Function Analysis

The cascade of functional requirements from customer to shop floor is the backbone of systems engineering design, supporting the delivery of the customer requirements in a robust and reliable manner by the engineered system. Within a *system-of-systems*, such as a vehicle, it is essential that the function analysis covers both (i) the *main* functions - which are directly required to deliver the customer required functionality, and (ii) the *interface* functions, which are required to manage the interactions between subsystems within the system and with external systems, in order to ensure the whole system integration (i.e. the vehicle). Evidence from previous studies [5] suggests that most problems in the field are due to systems interactions that have not been properly managed through the design process. In current system engineering practice function analysis is commonly underpinned by brainstorming and is generally biased towards the analysis of the main functions, which is seen as a weakness [8]. Therefore, a key deliverable from the function analysis FMA process step is a comprehensive list of functional requirements, including both main functions and interface functions, to be cascaded through the systems engineering “Vee” process [11].

Another important role of function analysis is to develop an optimal breakdown of the systems and the allocation of system design responsibilities. Historically, given the iterative / evolutionary nature of automotive design, the system design task decomposition is commonly done on a

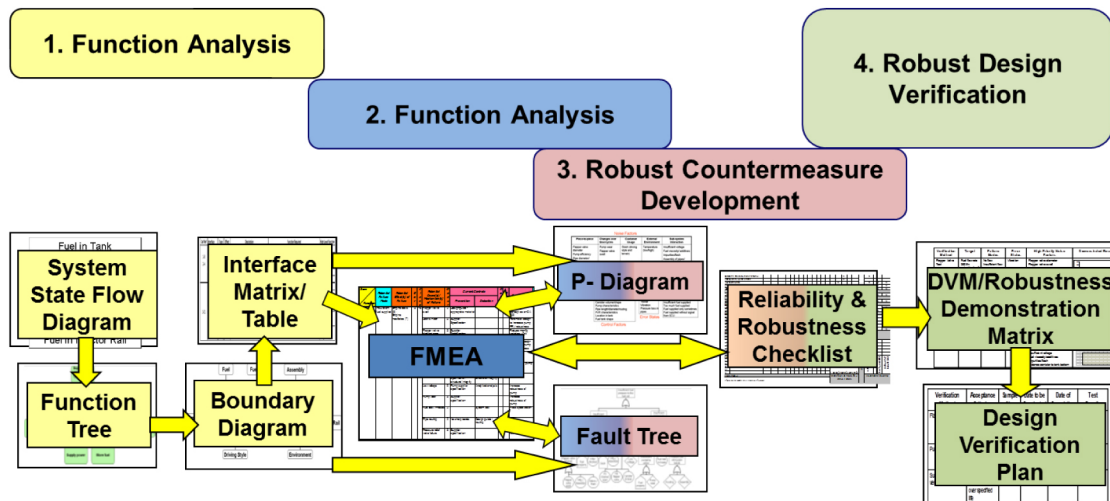


Figure 1. FMA Process Outline

structure / architecture basis, with design tasks being allocated to teams on an engineering “discipline” basis (i.e. mechanical, electrical, electronic, control, software). This approach does not facilitate system integration and it is not directly focused on customer required functions.

The weakness of current approach to system decomposition can be illustrated by taking the example of a typical exhaust system (considered as a case study later in this paper) for which the system design responsibility is split on a structural / disciplinary basis between 4 teams / units: (i) exhaust pipework, (ii) exhaust gas aftertreatment, (iii) sensors and (iv) control software, with no overall responsibility for the system integration on a functional basis. Experience shows that the lack of focus on system integration can lead both to un-necessary iterations within the product development process and suboptimal delivery of the system functions. Such issues can be avoided if the system decomposition analysis is pursued on a *functional* basis. This is promoted by the FMA process illustrated in Figure 1 by the introduction of the system state flow diagram (SSFD) [8, 12] as a structured diagrammatic tool that supports the high level function decomposition on an engineering basis. The SSFD follows the flows of energy, material and information through the system, and aims to identify *states* which can be described by measurable attributes. Within an SSFD diagram the states are represented in boxes while the arrows denote the engineering functions that need to be provided in order to ensure the transfer between states. Consistent with other generic approaches to system decomposition [13], at a given level of analysis (i.e. corresponding to a “level” within the systems engineering cascade, e.g. system, subsystem, component) decomposition granularity should result in 3 to 8 states and linking functions.

The main advantage of SSFD is that it promotes a structured decomposition analysis based on fundamental engineering analysis. This ensures that the functions that must be provided in order to meet the customer requirements

are well understood, and without reference to the structure (hardware) that provides the design solution. This latter point is important because it promotes solution independent thinking in systems engineering design, which fundamentally facilitates better design analysis and synthesis, coherent with the axiomatic design principles [9], as well as better allocation of the design responsibilities on a functional basis. Other tools commonly used in industry in conjunction with function analysis, such as function trees and boundary diagrams, can be derived from a SSFD, based on the same structured decomposition and engineering analysis of the system.

The FMA process also introduces an enhanced matrix based interface analysis tool, based on the interface analysis table (IAT) [8], which aims to systematically characterise the exchanges at an interface and document the functional requirements for managing the interface exchanges. The IAT is a very comprehensive, information rich document which not only provides a sound basis for the functional requirements specification and cascade, but also feeds into the other tools in the FMA process, in particular the FMEA, P-Diagram and Function Fault Tree. At any level of system engineering design analysis (i.e. system, subsystem, component), these functions should be part of the functional requirements specification, ensuring system integration, and should also be part of the systems engineering cascade to subsidiary systems and components.

Function Failure Analysis

Once the function requirements are identified, function failure mode analysis can be conducted to identify the potential function failure modes in the design, along with their effects and root causes. Failure Mode and Effects Analysis (FMEA) is a tool that it is widely used to guide, and document the results of function analysis and function failure analysis. At each level of analysis (i.e. system, subsystem, component) the FMEA focuses on the main functions of the

system, identified in the Function Tree and documented in the IAT. In taking a function failure mode approach to FMEA, the potential failure modes are (i) no function, (ii) partial function, (iii) intermittent function or (iv) function when not required (command failure). Interface functions, documented in the IAT, provide the possible root causes of these function failure modes, recognising that failure to manage an interface function is likely to cause failure in a main function. Completing the FMEA based on the IAT is a much more straightforward process compared to the conventional approach to FMEA [14] (which starts with “brainstorm functions” and continues with the identification of causes in a manner which tends to be based more on the previous experience of failure than a fundamental approach). The experience with the FMA process depicted in this paper is that the SSFD based functional decomposition and subsequent interface analysis based on this decomposition facilitates better (more concise and precise) FMEAs that are focused on a particular level in the systems hierarchy, compared to those developed in a more conventional manner.

Other tools that support the identification of root causes are Function Fault Tree Analysis (FFTA) [12] and the P-Diagram [5, 6]. FFTA is particularly useful with complex systems, where combinations of events and root causes can combine to initiate a significant failure mode. In particular combinations of partial function failure are likely to be missed with the more customary approach to FMEA analysis. The P-Diagram is particularly useful at lower level subsystems and component level, where identification of “soft” failure modes, causing robustness failures of the system, is important. Both FFTA and the P-Diagram build on the information from the IAT. Thus, higher level structure and content of the Function Fault Tree is based on the associated Function Tree and hence the functional decomposition achieved by the SSFD with the lower level structure and content of the Function Fault Tree being driven by information on function failure modes associated with the interface functions identified in the IAT. Since the external noise factors effects can only “come” through interfaces, the development of the P-Diagram is also based on information generated during the Function Analysis, in particular the information contained in the IAT. This approach to the development of both the Function Fault Tree and the P-Diagram can be compared to the more conventional approach to their development where, for example, in the case of the P-Diagrams noise factors and even the signal and response are typically generated through brainstorming [5].

Robust Countermeasure Development

The objective of this step in the FMA process is to ensure that all functional requirements are achieved reliably and robustly. High priority failure modes which indicate significant risks in design identified through FMEA criticality analysis are addressed through design countermeasures. The noise factors that cause failure modes are identified in the Robustness Checklist (RCL), which is strongly associated

with the P-Diagram. As discussed, much of the information required to develop the P-Diagram such as noise factors and function failure modes is already available from the functional analysis, and so the P-Diagram does not need to be developed in isolation of other key FMA tools. Function Fault Tree Analysis has also been discussed [12, 6] as a useful tool for developing countermeasures, in particular in relation to identification of opportunities for functional redundancy.

Design countermeasure development also relies on tools additional to those mentioned explicitly as a part of the FMA process illustrated in Figure 1, in particular analytical transfer function development and formal design optimisation methods, which are the core of the Design for Six Sigma (DfSS) process [15] and build on the FMA information flow.

Robust Design Verification

The role of design verification is to prove that the required functions are achieved robustly and reliably under customer usage conditions. An effective design verification process will need to consider all relevant failure modes and noise factors that can affect the functional performance of the product. The effect of each of the key noise factors identified in the RCL by function failure mode need to be included in at least one verification event; this inclusion is confirmed and documented in the RCL. Carefully thought out testing based on a consideration of noise factor effects and engineering experience allows design verification to be conducted primarily at the lower levels of the systems hierarchy and for any one verification event to include multiple noise factor effects [10]. This leads to an efficient design verification which can be summarised in the Design Verification Plan (DVP) document.

FMA Process Integration within a Systems Engineering Design Framework

The Vee model [11], illustrated in Figure 2, is a common conceptual model for systems engineering commonly used in the automotive industry. This model illustrates succinctly two important principles of systems engineering design:

1. Engineering design analysis and synthesis is carried out at a number of successive levels, for example:

- the vehicle level, as a system-of-systems;
- system level - e.g. powertrain;
- subsystem level - e.g. exhaust aftertreatment system;
- sub-subsystem - e.g. soot load sensing assembly
- component level - e.g. soot load sensor;

2. Systems engineering design is fundamentally based on the hierarchical cascade of functional requirements from the top level to the component level, where design optimisation ultimately takes place, and confirmation of successful integration of the manufactured system through design