The System Verification Breakdown Method

Mendonça, Cássio Henrique[a,a1]

[a]INPE – Instituto Nacional de Pesquisas Espaciais, Brazil.

Abstract. High integrated and complex systems are more and more scattered and common in people lives. Even without the exact feeling of this means, they hope for the best product. This desire implicate in system manufactures improve knowledgement and create solutions with more advanced technologies to satisfy consumers expectations. Verification have been done at the end of process development, but it have resulted in difficulties to manufacturers because is very expansive and hard to implement any required modifications at this point of development process. Thus, many manufactures have started verification process at begin of development, decreasing erroneous implementations. This paper is intended to show an intuitive method possible to apply in any cases, using block diagrams, that assists generate test cases procedures, since when development starts, making relationship among system interfaces, subsystems and functions, enabling tests traceability and tests coverage analysis. In cases where manufacturer develops same kind of products, the block diagram will be easily reused to a new one, including or removing systems, subsystems and functions, adapting it to new features and project requirements. The propose is starting developments doing the things right earlier as possible.

Keywords. Verification, System, Interfaces, Block Diagram, Reuse and Test Coverage.

1 Introduction

The verification and validation activities have been generated a lot of discussions where is applicable validation or verification. It is possible find many ways to understand and define these activities, even that they should executed together without clear and explicit separation in terms of time and stages of development. Many documents consider that validation is based in non implemented hardware and software while verification is based in a target implementation of hardware and software components. In spite of divergences about the best definition to validation and verification, there is a common

[1] INPE –Instituto Nacional de Pesquisas Espaciais; São José dos Campos – SP; Brazil; Av. dos Astronautas, 1758 – Jd. da Granja; Tel: +55 (12) 39471131; Email: chmendonca1977@yahoo.com.br;

consensus that validation means “Do the right things” and verification means “Do the things right”. Thus, both issues are supplementary during process to ensure the correctness products development and, at this paper, the term verification will be considered here disregarding stages of development.

2 Motivation

This work is motivated by the last modifications occurred in development process of high integrated and complex products forcing developers improve them knowledge about technologies, process and methods to speed up the development and ensure that the new products will be accepted by market and consumers will be satisfied.

The intention of this paper is propose a method to start verification even without formal requirements have been written. When a development is defined, developers have, at least, a minimal knowledgement of the system under construction, thus it enables the start of verification process, once that it is possible create a block diagram with systems and breakdown it in a small parts. This method allows and helps many levels of verification, traceability of tests and coverage. A brief example will be done later.

3 The System Verification Breakdown Method

The verification activities, usually, were executed from the middle to end of product development, when implementation in the target was started. It have caused a lot of difficulties because problems of erroneous creation and implementation of elements and components that when integrated in target were figured out and at this time is very complex and expansive any required modification to fix this mistakes.

Thinking in avoid this kind of problems, manufactures, using the new facilities allowed by grow up of computer sciences, starting modeling requirements and testing it to ensure that the desired functionalities were created and implemented. Only after test all, or at least, a major part of testable requirements, the hardware and software implementation is ready to be initiated. This model tests are proving a good initiative in order to reduce later problems finding. Developers, just after write requirements, were able to generate models and starting tests.

Unfortunatelly, the process to write requirements is not a short process and it is made during many stages of development, it causes changes due requirements correction and creation. Then it implicates in generate models with risk of hard modification after requirements revision or even create a new one. Due a divergences in accept verification while modeled systems and difficulties to create these models due a requirements modifications, a considerable number of problems are figured out in implementation in target phases.

If verification starts even no formal requirements have written, it will help developers to create requirements more reliable and correct. It could make easy the

identification of integration problems even if the systems and their interfaces are not detailed and totally defined.

Using blocks diagrams to depict systems is an easy way to visualize whole integrated system because they are more readable and intuitively understandable than formal languages. With the complete view of the system and its interfaces is possible investigate details not so simple to think when writing a requirement. Using a symbolic way to present ideas about system could be simplier than use words, spoken or written. This way to analize system is valid and applicable also to subsystems, elements, components, functionalities among others. In any moment is demonstrated the intention of change formal requirements by non-formal languages, it is only expected to aid developers to visualize the systems.

Usually a developer creates each system separately establishing interfaces between systems only by requirements, but it could not be well understand by parts causing problems later during integration. This example is only one case, but misunderstand could be generated inside of own workgroup, where subsystems do not agree with them interfaces. A device defined by one part of the system or subsystem could not met all features requested by other or a post requirements elaboration changes could not be transmited completely or clearly.

With this method is possible make integration of parts and start a process to develop a complete verification analyzes from system to components and functionalities (top down verification) or from components and functionalities to system (bottom up verification). The system verification responsible could use this method to perform analyses around the correction of system functions (top down) in normal or abnormal conditions. The components verification responsible could start the verification from componens and devices level (bottom up). Both, system verification responsible and components verification responsible are able to use the same block diagram breakdown, it only depends of fidelity and detail of block diagram breakdown.

Other advantage is facilitate the coverage of tests. Usually is generated a verification matrix to ensure that all requirements are verified. The verification matrix is used to make cross between each requirement with each verification modalities and during the initial stages of developments and based in written requirements that can change. Here is easy to split systems in subsystem or in minor parts and analyze each separate parts as well as integrations and interfaces.

Nowadays the reuse of tools, developments, process, methods is very commented, but the reuse is not so easy to achieve once that, there are improvements and different objectives from development to development. Other feature this method is reuse the block diagram, making modifications that are not so hard to implement. It only needs remove or add blocks and rearrange interfaces, inputs and outputs. It is very normal that an product have very closed features with other, for instance, in automotive development a automobile could have automatic transmission while other manual transmission, thus, the changes should be done in transmission diagram blocks where one receive car speed as input to shift gear while another wait for driver command among others signals.

Basically, method is based in imagine how create a system and its main features and interfaces and inputs/outputs. Following this, start creating a low level of system and linking interfaces. Continue breaking systems in subsystems and

putting interfaces and related signals. Repeating this until the system is totally broken and covered. Next a example to demonstrate the idea to understand better it, as described above, sometimes visual depict is more readable and intuitive. This method consider that who develops it have a minimal knowledement of the system.

4 A Electric Wheel Chair Example

To this first work with this method was chosen an electronic wheel chair as example. This is an interesting system, once that it has structural, electrical, electronical and mechanical subsystems. The basic features of electric wheel chair are:

- The Electric Wheel Chair shall move without a human pusher.
- The Electric Wheel Chair shall have a control to command chair by the desired way

It can be understand that these two basic features are requirements to start project. Based in these is possible start thinking in interfaces and necessities that system will require to satisfy them. After keep in mind the basic requirements, starts the phase where detailed requirements are written. But like we are intended to show that no requirements are necessary to start the assembly of this block diagram breakdown we will start drawing blocks.

First of all, it is necessary understand which are the main interfaces of electric wheel chair and them insert in a initial block its interfaces, like **Figure 1**.

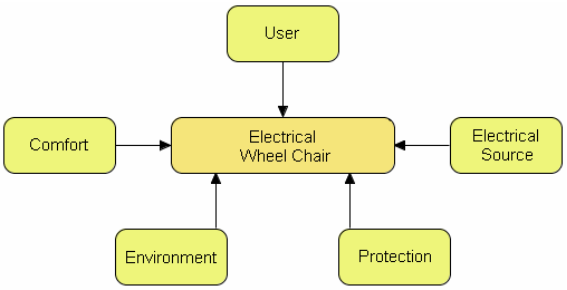

Figure 1 - Electric Wheel Chair Interfaces

After gathering all possible interfaces, in major view, break system in known technologies or parts and insert those interfaces internal and external. In this case was chosen break in technologies electrical system, mechanical system and structural assemblies (**Figure 2**).

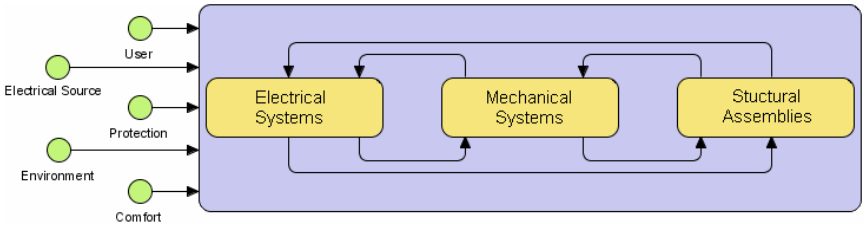

Figure 2 - System Major Breakdown View

Start to splitting technologies in separated blocks to make easier the connections of interfaces, but always thinking systematically and shall never forget other systems and technologies interfaces.

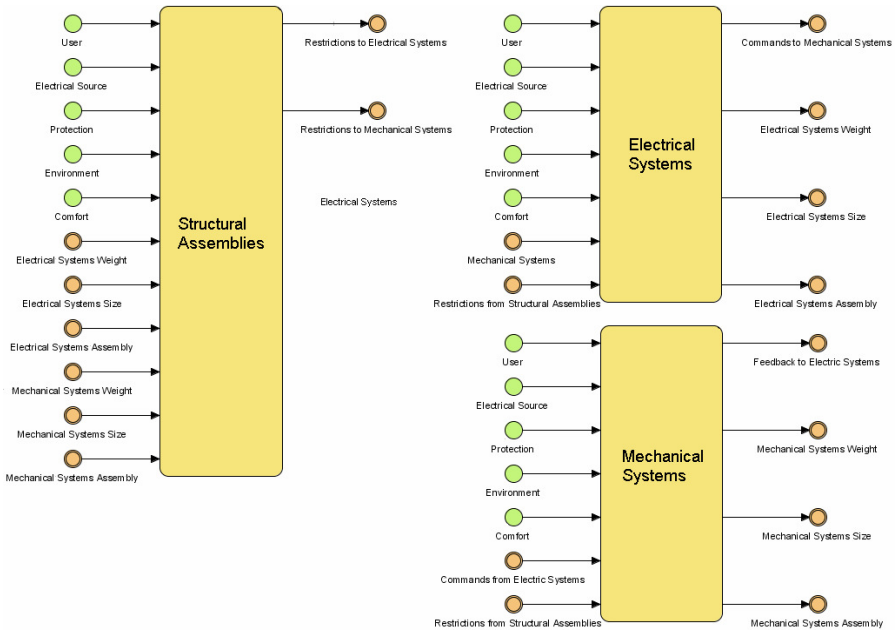

Figure 3 - Detailed Technologies View

It is possible to visualize in **Figure 3** that main interfaces and relationships among systems are kept allowing analyze connections and ensure that all parameters and signals were inserted.

To simplify this paper was chosen detail only blocks to electrical systems. As below, **Figure 4** depicts in more details electrical systems that was broken in electrical sources, electrical commands and torque system. At this step is possible insert in each subsystem their respective inputs and outputs parameters. An example is that all subsystems have a kind of protection, but comfort is a not necessary characteristic to battery, but necessary to other three subsystems.

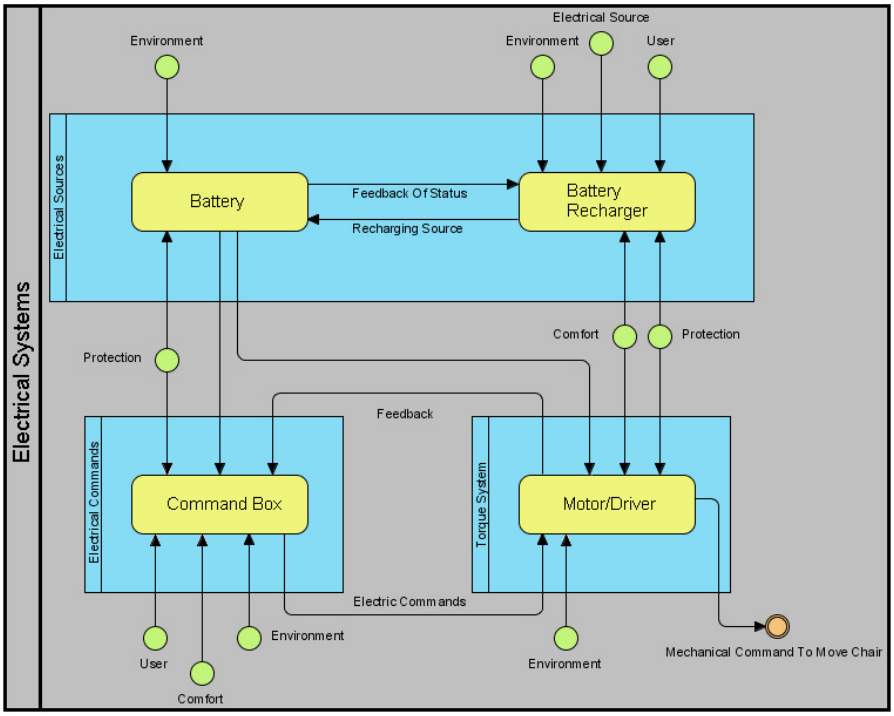

Figure 4 - Electrical Subsystems Relationships

In order to demonstrate reuse, it will be created new requirements to modify system and show how this method could be reused. These requirements are proposed, for instance, to create a higher level of product with more functionality and confort. Consider next requirements:

- The Electric Wheel Chair shall have a control to recline back and feet to increase comfort of user.

- The Electric Wheel Chair shall have a protection to no wheel motion available when it is reclined. Rationale: this requirement is necessary to avoid collision and damage to user and equipment when user is reclined and his visibility has no long range.

The new implementation could be done inserting a new box with a motor to recline back and feet like **Figure 5**. Note that there signals that are going from existent boxes (command box and battery) to new one and vice-versa.

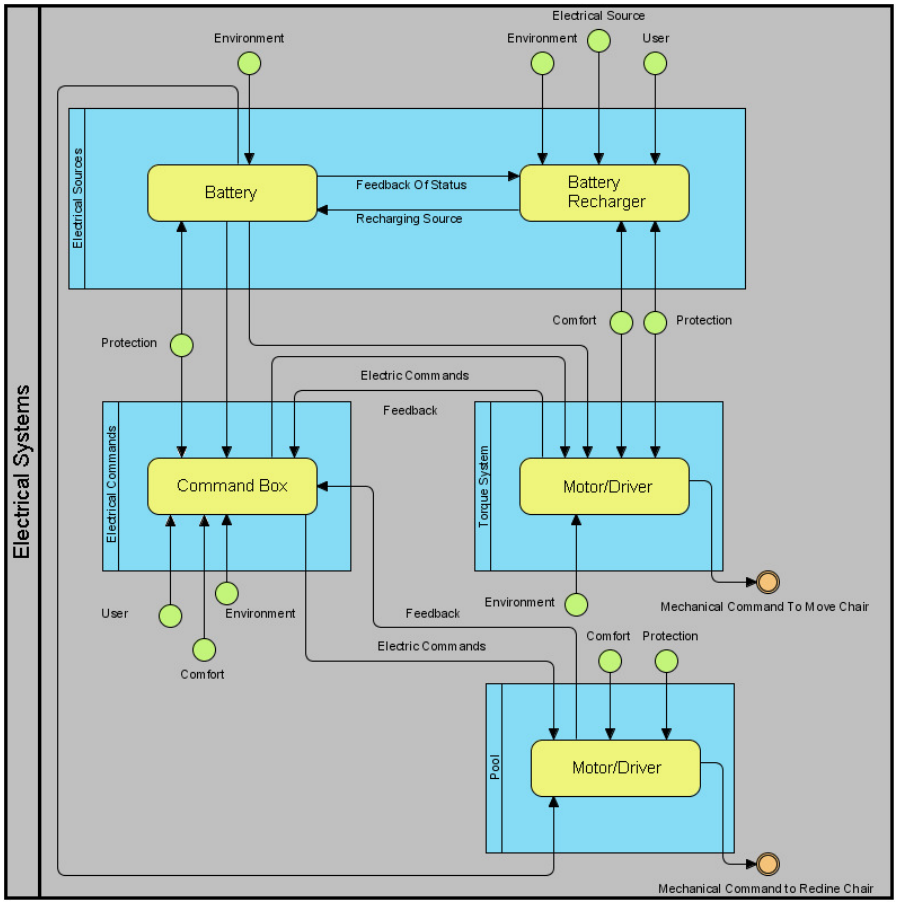

Figure 5 - Electrical Subsystems Relationships With Two Motors

This example is stopping at this point, but following the present idea it could be detailed even no requirements written, for instance, command box could be break in two functions, move chair or recline. Example of modification could be adding physical system like motors that could be break in motors and clutch and so on.

Verification procedures changes due system modifications are common once system is under development and improvements are able to be implemented, but there is a time where requirements and improvements will be frozen. Even after requirements frozen, verification procedures shall suffer modifications, when necessary, to keep them updated.

5 Conclusion

While requirements in creation, it is possible increase details of functions subsystems, components, interfaces parameters, etc. When desired details are

achieved, other activities could be initiated, the detailed verification procedures. The detailed verification procedures can be created from top to down or bottom to up in this system chain.

Using this method to perform verification, is possible keep control over traceability once that modifications can be easily met in block diagrams and interfaces parameters replaced as necessary. It is possible to analyze the impact over integrated systems in different levels and aid to accept changes. The integration is a good point to exercise in this method once it regards all interfaces with system and external factors.

Other point to highlight is the facility to reuse same block diagram to different improvements in systems only making modifications adding or removing blocks and interfaces.

New improvements in this method shall be implemented to become it more useful than now. Improvements should start by creating software to generate diagrams in order to facilitate to user, creating a user friendly interface to drag blocks and create links.

6 Reference List

[1] Chew, J., Sullivan C. Verification, Validation and Accreditation in the Life Cycle of Models and Simulation. Proceedings of the 2000 Winter Simulation Conference, 2000.

[2] Ladkin, P. B. Analysis of a Technical description of the airbus A320 Braking System. CRIN-CNRS & INRIA Lorraine, Bâtiment LORIA, BP 239, 54506 Vandoeuvre-Lès-Nancy, France, 1995

[3] Lutz, R. R., Mikulski, I. C. Requirements Discovery During the Testing of Safety-Critical Software. Jet Propulsion Laboratory of California Institute of Technology.

[4] Souza, M. L. De O. e, Trivelato, G. da C. Simulators and Simulations: Their Characteristics and Applications to the Simulation and Control of Aerospace Vehicles. Society of Automotive Engineers, 2003.

[5] Varró D., Varró, G., Pataricza, A. Automatic Graph Transformation In System Verification. Technical University of Budapest, Department of Measurement and Information Systems.

[6] VanDoren, V. J., Simulation Simplifies 'What-if' Analysis. Control Engineering, July, 1998; 38-45.

[7] System Engineering Fundamentals. Defense Acquisition University, 2001.

[8] System Engineering Handbook. International Council on Systems Engineering. INCOSE, 2004.

[9] System Engineering Handbook. National Aeronautics and Sace Administration. NASA, 1995.

[10] Design and Verification Strategies for Complex Systems: Part 1. Available at: <http://www.dspdesgnline.com/showArticle.jhtml>. Accessed on: Jan. 18th 2007.

[11] Design and Verification Strategies for Complex Systems: Part 2. Available at: <http://www.automotivedesignline.com/189401216>. Accessed on: Jan. 18th 2007.

[12] Electric Wheel Chair Characteristics Information. Available at: < http://www.medicalproductsdirect.com/inneutelser.html>. Accessed on: Jan. 27th 2007.

[13] Electric Wheel Chair Characteristics Information. Available at: <http://www.1800wheelchair.com/asp/view-product.asp?product_id=1786&s_cid=dlshpg_1786>. Accessed on: Jan. 27th 2007.