Engineering of Solid Rocket Motors

Engineering of Solid Rocket Motors:

From Production to Firing

Edited by

José A. F. F. Rocco, Rene F. B. Gonçalves and Marcela G. Domingues

Cambridge Scholars Publishing

Engineering of Solid Rocket Motors: From Production to Firing

Edited by José A. F. F. Rocco, Rene F. B. Gonçalves
and Marcela G. Domingues

This book first published 2023

Cambridge Scholars Publishing

Lady Stephenson Library, Newcastle upon Tyne, NE6 2PA, UK

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN (10): 1-5275-0997-4
ISBN (13): 978-1-5275-0997-9

TABLE OF CONTENTS

PREFACE

This book concentrates on the subject of solid rocket propulsion engineering, focusing on manufacturing, performance and common issues observed in the industry and the operation of solid rocket motors. The aim is to present an introduction to rocket engineering, describing the project needs, integration, manufacturing methods, new developments and possible problems that occur in the rocket or its components. Solid rocket motors are used on air-to-air and air-to-ground missiles, on model rockets, and as boosters for satellite launchers. In a solid rocket, the fuel and oxidizer are mixed together into a solid propellant grain configuration which is packed into a solid case. A hole through the case serves as a combustion chamber. When the grain is ignited, a complex combustion process takes place on the surface of the solid propellant. Special attention is devoted to the production of rocket motors based on solid composite propellant formulations. Solid composite propellants are made of a polymeric matrix, loaded with a solid powder oxidizer and a metal powder that plays the role of a secondary fuel component. A certain number of properties, such ballistics as burning rate and specific impulse and mechanical behavior as stress and strain, are directly related to this composite formulation grain character. Finally, this book is intended to give an overview of chemical propulsion applied to solid rocket motors from their production process to firing. These systems generate a lot of thrust and can be stored for long periods of time in a "ready-to-go" state, but generally lack the controllability of turning them off and on when thrust is no longer needed or when a series of thrust pulses is needed, with some exceptions.

The first chapter provides an introduction to system engineering and the most common methods for developing rockets. As a rocket contains multidisciplinary systems, an integration model is of utmost importance for combining all areas and to provide a smooth development process. The next three chapters focus on the manufacturing of the solid propellant and the rocket case. This information is not commonly found in the literature, as usually the reports are on small scale systems, and most industries prefer to maintain their process as a "trade secret". The following chapters present information about internal ballistics, motor operation and common issues, like accidental ignition, transition phenomena and burning issues of the propellant during the motor operation. It is necessary to know and control these parameters to guarantee a perfect functioning of the system and completion of the mission.

Most of the knowledge contained in this book was obtained from experience, production of large-scale rockets and extensive research. A balance is presented between theory and practical engineering, rocket motors and their key components and also the analysis and performance of real operating systems. Although rocket propulsion has grown a lot in the last decades, there are still many empirical aspects, procedures and methods used in the manufacturing. As an example, depending on the formulation of solid propellants, there is a preferred order for adding the components to the batches, which increases the reproducibility of the final grains.

Solid rocket propulsion is still the most employed method of chemical propulsion, due to its long history, effectiveness, simplicity when compared to liquid and hybrid systems, high specific impulse, and reliability. However, there is a large variety of undesired issues that can happen during the solid propellant burn, which can be avoided or at least controlled by using special additives or modifying some ballistic parameters. The scale-up of rockets is not trivial, as many of these issues only happen in large-scale rocket chambers, so most of the rocket labs cannot report them as the test benches are usually much smaller. It is not by chance that no new solid propellants have been introduced and used in large scale over some decades, although there have been many reports and developments of materials, additives, and fuels over the years. Therefore, we aimed to offer deeper information on the most used and reliable systems through the chapters. Some information on new simulation methods is also mentioned, like reactive molecular dynamics simulations, which are very useful in understanding the combustion, stability and even aging of solid propellant formulations, as well as providing useful thermodynamic, kinetic and ballistic properties of the systems, and to elucidate the decomposition/combustion mechanisms.

Rocket propellants are hazardous materials. The authors and the publisher recommend that the readers do not work with them or handle them without an exhaustive study of the hazards, the behavior, and the properties of each propellant, and rigorous safety training, including becoming familiar with protective equipment. Safety training is routinely given to employees by organizations in this business. Neither the authors nor the publisher assumes any responsibility for actions on rocket propulsion taken by readers, either directly or indirectly. The information presented in this book is insufficient and inadequate for conducting rocket propulsion experiments or operations.

This book was directed at engineering students, rocket scientists and people who love space and aerospace systems. We hope to make a positive contribution with new information, and that this book becomes a complement to the study, projection, manufacturing and testing of solid rockets.

We gratefully acknowledge the help and contributions we have received in preparing this edition. A special thanks is given to Professor Luigi T. DeLuca, for the constant support, opportunities and valuable information provided. Without his help, this book would never exist, and a large amount of information would not be available for students and engineers and for the overall scientific community.

José Rocco
Rene Gonçalves
Marcela Domingues

CHAPTER ONE

SYSTEMS ENGINEERING PROCESS AND ROCKET DEVELOPMENT

LUÍS EDUARDO NUNES DE ALMEIDA[1],
JOSÉ A. F. F. ROCCO[2]
AND RENE F. B. GONÇALVES[2]

[1]Avibrás Indústria Aeroespacial Ltda., São José dos Campos/SP, Brazil
[2] Instituto Tecnológico de Aeronáutica, São José dos Campos/SP, Brazil

Systems Engineering Fundamentals

There is no one theory that completely defines systems engineering (SE).

SE is not simply a planning process to define and execute the job at hand. As the title of this book suggests, systems engineering is one piece of the total topic to be discussed. SE, however, is considered only in a loose sense, and the focus of the book is on the analysis and trade space associated with producing a balanced air and missile defense system (AMDS). SE will be treated first as an outline to accomplish the true purpose of the book. This is not a theoretical treatment of SE nor is it an exhaustive practical treatment. Accordingly, this book is not advocating for, nor arguing against, any specific SE formula. [1]

Systems Engineering Overview

The main purpose of this chapter is to offer a brief overview of the concepts of systems engineering. It is important to begin with some definitions, an abbreviated survey of the origins of the discipline and discussions of the value of applying systems engineering. Another objective of this chapter is to fortify the concept that when we design any spatial system or subsystem, we will always be designing systems that interact with other systems. The concepts defined by systems engineering are always a good guideline for a designer towards an integrated vision with a high success rate in complex projects.

Definition of systems engineering

Systems engineering is a profession, a process, and a perspective as illustrated by these three representative definitions [2]:

- Systems engineering is a discipline that concentrates on the design and application of the whole (system) as distinct from the parts. It involves looking at a problem in its entirety, taking into account all the variables and relating the social to the technical aspect. [3]
- Systems engineering is an iterative process of top-down synthesis, development, and operation of a real-world system that satisfies, in a near optimal manner, the full range of requirements for the system. [4]
- Systems engineering is an interdisciplinary approach and means to enable the realization of successful systems. [5]

Certain keywords emerge from this sampling – interdisciplinary, iterative, socio-technical, and wholeness.

The systems engineering perspective is based on systems thinking. Systems thinking is obtained through discovery, learning, diagnosis, and interlocution that lead to sensing, modeling, and talking about the real world to better understand, define, and work with systems.

The systems engineering process has an iterative nature that supports learning and continuous improvement. Complexity can lead to the unexpected and unpredictable behavior of systems, hence, one of the objectives of systems engineering is to minimize undesirable consequences.

Since systems engineering has a horizontal orientation, the discipline includes both technical and management processes. Both types of processes depend upon good decision-making. Decisions made early in the life cycle of a system, whose consequences are not clearly understood, can have enormous implications later in the life of a system. It is the task of the systems engineer to explore and predict these issues and make critical decisions in the best way and in a timely manner.

The role of the systems engineer is varied and according to Sheard, it is one source for a description of these variations [6].

There are many examples [7–14] to choose from. It is advocated that you simply need to have one and that you try to keep it as simple as possible while not letting the details of the program slip into the cracks. Certainly, if you are spending more money on planning and executing the program than on designing, testing, deploying, and sustaining it, you have some issues to address. You should, however, be allocating about 30% of your resources to the development of planning and requirements.

To avoid adding superfluous material, the definition that is most compatible with the topic of air and missile defense is adopted with some added descriptors. Systems engineering can be defined as "a process that is comprised of a number of activities that will assist in the definition of the requirements for a system, transform these requirements into a system through design and development efforts, and provide for the operations and sustainment of the system in its operational environment" [7].

The systems engineer is the one who is responsible for the program definition and who puts the plan of action into motion. The systems engineer has three roles: the technical roles of an architect, designer, integrator, and tester; the role of a systems or technical manager; and the role of a production engineer. To achieve success, the systems engineer is required to employ both artistic and scientific engineering skills.

An experienced systems engineer develops instincts for identifying and focusing a team's efforts on activities that will ultimately achieve an optimized or balanced design while accounting for lifecycle considerations. The art of systems engineering takes the form of developing the right set of design alternatives and options and then developing the necessary trade studies that will help the systems engineer to eliminate all but the best sets and combinations of alternatives from which an investment decision(s) can be made.

The purpose of SE is to establish a repeatable, traceable, and verifiable methodology to produce systems and products to facilitate verification, validation, and accreditation (VV&A) with an improvement in the cost and schedule while minimizing the risks associated with engineering endeavors. SE includes configuration control management and lifecycle sustainability and maintainability. Systems engineering starts by defining a standard framework within a common lifecycle process that can be applied to any system regardless of the scope or scale of the project. Numerous system engineering frameworks have been proposed.

Hall [7, 8] is widely quoted either implicitly or explicitly throughout the systems engineering literature [9–12]. Hall proposed that systems engineering has three major dimensions that make up the Hall morphological box of systems engineering: time, logic, and professional disciplines. Practically, this decomposition is incomplete and premature.

NASA [13] proposes the morphological framework to be a three-component model. Here we believe there is a fourth component and a slighted modified third component.

This systems engineering framework recommended for practice is shown in Figure 1, which shows that systems engineering can be thought of as vectors toward achieving one's goals and objectives. A program requires four components to succeed. The first component is a well-thought-out organizational structure or integrated product team (IPT) structure. This component is the most important.

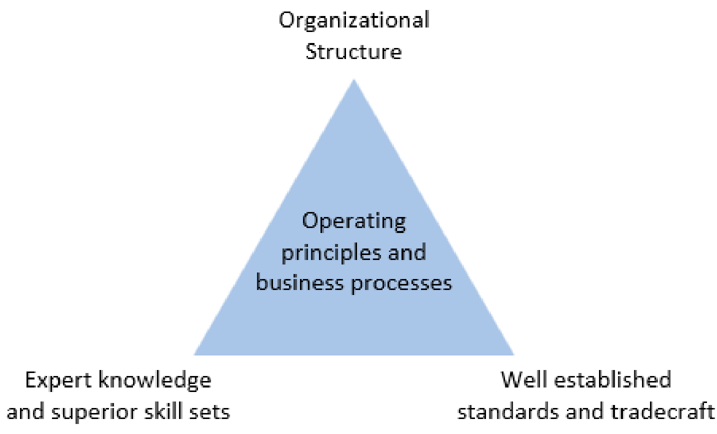

Figure 1. Three dimensions of systems engineering

Without the right organizational structure or set of structures, the program has little chance of succeeding. The second component is to populate the lead positions in the IPT (organizational structure) with experts in their field and having superior skill sets demonstrated by achievement. Third, the program needs to have established engineering standards and tradecraft practices communicated and understood throughout the team. The fourth component is to have well-established operating principles and business practices through the program. The operating principles and business processes are at the center of the pyramid to communicate that the alignment of the other three components relies on standardized operating principles and business practices.

Rockets are very complex machines. Developing a rocket and its subsequent test programs and operational lifecycle is an even larger and more complex challenge than the machine itself.

NASA has a great background in how to develop a system of systems and according to the NASA *Systems Engineering Handbook* (2007), there is an important concept adopted and described as follows:

> *Systems engineering is a methodical, disciplined approach for the design, realization, technical management, operations, and retirement of a system. A 'system' is a construct or collection of different elements that together produce results not obtainable by the elements alone. The elements, or parts, can include people, hardware, software, facilities, policies, and documents; that is, all things required to produce system-level results.*

In other words, a system is a complex thing made up of many pieces and functions that systems engineers often refer to as "elements." It is sometimes these millions of elements that make up the overall development effort that includes the hardware and software of the functioning device, all of the support infrastructure, the lifecycle elements, from conception to the end of life, and any other aspect involved with the project. NASA is an organization that follows this philosophy and the Department of Defense (DoD) also uses it as well as dozens of companies that develop complex systems.

Figure 2 shows the NASA program lifecycle. The first one is (1) formulation; and the second one is (2) implementation.

The formulation phase is the study and development of the ideas, while the implementation phase is the actual performance of the concept and achieving the end result.

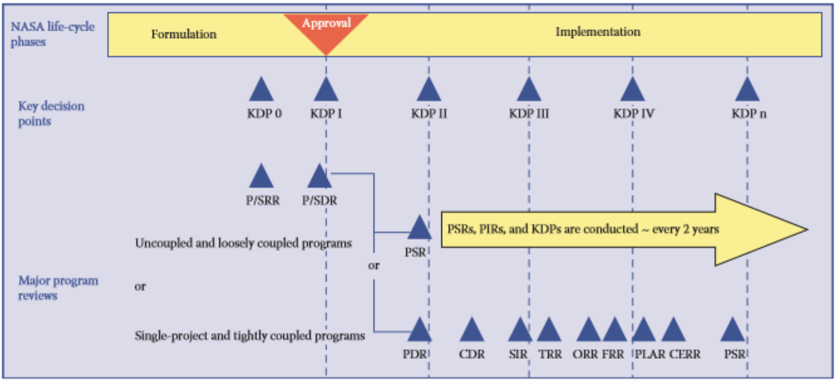

Figure 2. NASA program lifecycle shows the steps of a large-scale development space program (Courtesy of NASA)

Program Formulation

The program formulation phase establishes a cost-effective program that is demonstrably capable of meeting goals and objectives. The program formulation authorization document (FAD) authorizes a program manager (PM) to initiate the planning of a new program and to perform the analyses required to formulate a sound program plan. Major reviews leading to approval at KDP I are the P/SRR, P/SDR, and PLAR, and the governing Program Management Council (PMC) review.

A summary of the required gate products for the program formulation phase can be found in NPR 7120.5. Formulation for all program types is the same, involving one or more program reviews followed by KDP I where a decision is made approving a program to begin implementation.

Phase		Purpose	Typical Output
Formulation	**Pre-Phase A** Concept Studies	To produce a broad spectrum of ideas and alternatives for missions from which new programs/projects can be selected. Determine the feasibility of the desired system, develop mission concepts, draft system-level requirements, and identify potential technology needs.	Feasible system concepts in the form of simulations, analysis, study reports, models, and mockups
	Phase A Concept and Technology Development	To determine the feasibility and desirability of a suggested new major system and establish an initial baseline compatibility with NASA's strategic plans. Develop the final mission concept, system-level requirements, and needed system structure technology developments.	System concept definition in the form of simulations, analysis, engineering models, and mockups and trade study definition
	Phase B Preliminary Design and Technology, and Completion	To define the project in enough detail to establish an initial baseline capable of meeting mission needs. Develop the system structure end product (and enabling product) requirements and generate a preliminary design for each system structure end product.	End products in the form of mockups, trade study results, specification and interface documents, and prototypes
Implementation	**Phase C** Final Design and Fabrication	To complete the detailed design of the system (and its associated subsystems, including its operations systems), fabricate hardware, and code software. Generate final designs for each system structure end product.	End product detailed designs, end product component fabrication, and software development
	Phase D System Assembly, Integration and Test, Launch	To assemble and integrate the products to create the system, meanwhile developing confidence that it will be able to meet the system requirements. Launch and prepare for operations. Perform system end product implementation, assembly, integration and test, and transition to use.	Operations-ready system end product with supporting related enabling products

	Phase E Operations and Sustainment	To conduct the mission and meet the initially identified need and maintain support for that need. Implement the mission operations plan.	Desired system
	Phase F Closeout	To implement the systems decommissioning/disposal plan developed in Phase E and perform analyses of the returned data and any returned samples.	Product closeout

Figure 3 shows the NASA project lifecycle. It is this lifecycle that is most relevant to the development of a rocket system. The program and project lifecycles enable the rocket scientists and engineers to categorize all the element goals of the mission program and the subsets of rocket development efforts that must be reached in order to reach a successful conclusion.

The cycles include "key decision points" (KDPs): government speak for "go or no go." The project lifecycle includes phases A through E, which are defined as follows:

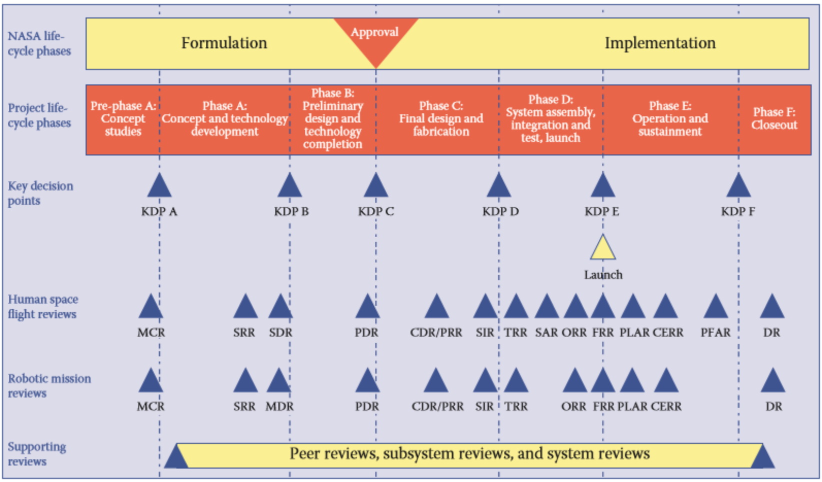

Figure 3. NASA program lifecycle shows the steps of a spacecraft development space program (Courtesy of NASA)

Pre-Phase A

A notional pre-phase A management plan is shown in Table 1. Each of the program strategy teams is put into action starting in pre-phase A. The management team manages this matrix, the internal processes, documentation, costs, constraints, and risks. The other teams execute their jobs as described earlier. This phase typically consists of the government team and laboratories, and should include prime contractor involvement. The prime contractors are truly the only ones who know how to engineer, manufacture, and deploy products. The prime contractor and laboratory support teams should have already been established for planning purposes.

Table 1. Pre-Phase A Management Plan

Entry Criteria/Inputs	**Pre-Phase A**		
	Develop and Implement the Program Plan		
	Project Readiness Review		
	Activity	**Documentation**	**Timeline**
Mission need	Translate MNS into program objectives	Top-level requirements document	Months
Authorizations/ funding	Define program objective measures of effectiveness		Months
	Define top-level requirements		Months
	ID/Define constraints and risks	Systems Engineering Master Plan (SEMP)	Months
	ID/Define SE processes/methods		Months
	ID/Define knowledge, skills, abilities, and requirements		Months
	Define work breakdown structure		Months
	Develop requests for proposals	RFPs	Months to a year or more

The objective of this phase is to develop and implement the program plan. It cannot be expected to accomplish practical objectives without their involvement upfront. Pre-phase A will include defining the mission need in terms of realizable goals and objectives, concept systems and architectures, draft measures of effectiveness/performance, and systems' top-level requirements (TLRs). In addition, stakeholders and their expectations are clearly identified; technology development needs are identified; and trade studies are identified and defined in detail. Iteration between these activities is expected to fully and accurately complete pre-phase A. Verifying and validating results are necessary in each phase.

The final pre-phase A output is likely to be a project readiness review (PRR).

The SEMP is essentially a road map to navigate through the program, which will be updated in the next two phases. The prime contractor and laboratory support teams should have already been established for planning purposes. Prior to developing a WBS and completing the SEMP, the program IPT structure must be in place. However, executing the program may require additional or different contractors. This fact will require the sending out of a round of requests for proposals (RFPs) at the completion of pre-phase A in an effort to hire the right contractor teams for the effort. The RFPs will usually be answered within a 30- to 45-day period, and the technical evaluation team will need to recommend a selection to the management team. The selection process may take months.

Phase A

A notional phase A management plan is shown in Table 2. This phase typically consists of the government team and laboratories and should, as in pre-phase A, include prime contractor involvement. The objective of this phase is to fully develop a baseline mission concept and assemble the systems requirements document (SRD). The final output is the systems requirements review (SRR).

Phase A develops and refines feasible concepts and finalizes goals and objectives. Some concepts may be eliminated while others may be added. Systems and architectures, and measures of effectiveness/performance are refined. The TLR document is updated and approved. At this point, technology development requirements and a risk assessment are developed for each viable concept that, in turn, becomes part of the set of trade studies. Trade studies are executed with the purpose of eliminating bad concepts and ranking good concepts. Trade studies focus on evaluating technical, schedule, and cost objectives.

The result of the trade studies includes a system and architectural baseline; functional allocations to hardware, software, and other resources; and new plans are developed. The SEMP now contains more details of the associated management plans and will be updated again in the next phase. It will be a requirement to send out the final RFP sets.

Table 2. Phase A Management Plan

Entry Criteria/Inputs	Systems Requirements Document		
	Systems Requirements Review		
	Activity	Documentation	Timeline
Authorizations/funding	CONOPS	Systems requirements document	Months
SEMP	Top-level systems architecture		Months
TLR	Requirements flow down		Months
SER	Configuration control management plan	Update SEMP/WBS	Weeks to a month
	Validation, verification, and accreditation plan		Weeks to a month
	Data development plan		Weeks to a month
	EMD plan		Months
	Request for proposals	RFPs	Months

The objective here is to expeditiously send out the RFPs and then to receive, evaluate, and select contractors from the proposal responses. In some cases, the government will use the PDR in phase B to down-select the final contract awards. In this case, the competing contractors down-selected from the pre-phase-A RFP process are funded to produce a preliminary design for review based on the documentation prepared in pre-phase A and phase A.

This process allows the government to actually see the contractors in action. It provides the government with multiple design choices, which is subsequently a risk reduction activity in itself. Moreover, in the event that the selected contractor slips, the government has a fallback position with the runner-up contractor. The added cost for funding multiple PDR phase competitors essentially buys down program risk while allowing multiple contractors to develop new capabilities that would benefit them and the government in follow-on competition.

Phase B

A notional phase B management plan is shown in Table 3. This phase typically consists of the government team and laboratories, and the focus is on the prime contractor developing a preliminary design for review. The objective of this phase is to produce a system definition with enough detail to baseline a design for EMD and capable of meeting the mission need. The final output is the preliminary design review (PDR). Phase B includes the flow down of design and performance requirements to subordinate systems and subsystems within the architecture. Interface requirements are added to the SRD. Trade studies defined in phase A continue as required to refine concepts and are input to newly defined design studies aimed at allocating capability and performance to systems and subsystems.

The design studies include interfaces, which include hardware, software, and communication within the architecture, systems, and subsystems. Verification and validation plans are developed. Finally, all of the products developed in phase A are updated and reapproved. Verification and validation accompany the results of trade studies. An updated SRD, system design documents (SDD), a verification assessment, and an updated EMD plan are presented for approval at the PDR. The SRD is now in lockdown. However, requirement changes beyond this point are a normal and expected occurrence.

The difference is that a formal procedure and approval by configuration control management (CCM) must take place. Any proposed changes to the requirements will usually cause a cost and schedule impact to the program. If requirement changes are inevitable (requirement creep), it usually means that cost and schedule increases should be expected. Technical interchange meetings (TIMs) between all or subset IPT members are a requirement during this phase. TIMs are also highly encouraged in the earlier stages, but are shown specifically in the phase B management plan because of the critical nature of early and continuous communication and resolution of issues between IPT members when the preliminary design is being developed.

Table 3. Phase B management plan

	Phase B		
	Produce System Definition Detail Necessary to Establish a Baseline Design of Meeting Mission Need		
Entry Criteria/Inputs	**Preliminary Design Review**		
	Activity	**Documentation**	**Timeline**
Authorizations/funding	Update SRD, flow down requirements to subsystems	Updated SRD	Months
SEMP	Place requirements under CCM		Weeks
SRD	Develop design solutions and ICDs	Systems design document	3-6 months to a year
TLR	Produce performance predictions		Months
	Develop drawings, design to specifications		Months
	Validate systems design solution against requirements	Validation assessment	Months
	Technical interchange meetings	Memoranda	Routine
	Update EMD plan	Updated EMD plan	Weeks

Phases C–F

Management plans for phases C–F are assembled in the same manner as shown earlier, but the details are different. These program phases are, however, beyond the scope of this book and are not presented. The important lesson here is that although macro schedules and plans are required, breaking the program down into small chunks is more manageable. The larger and more complex the program, the more essential it becomes to break it into smaller pieces. In turn, as the system is subdivided into smaller pieces to manage interfaces, interoperability solutions become an increasingly important design consideration for attention.

Air and missile defense systems (AMDS) are complex engineering undertakings composed of many systems within systems and subsystems within systems. AMD subsystems have many components and those components interact in complex ways with each other and the environments surrounding them. The interactions are complex and can lead to unpredictable and sometimes unexpected results.

Essentially, AMD systems engineering is rocket science and needs to be treated accordingly. Following good systems engineering practices will enhance the likelihood of achieving optimized performance and ultimate success in satisfying program objectives within cost and schedule. Systems engineering employed properly will aid the production of a balanced design despite, at times, being faced with opposing and conflicting physical, fiscal, and time constraints. Each phase discussed earlier has three common and primary focused activities that are solved iteratively.

These are requirements development and management, concept development, and architectural design solution development. These activities are tied together with rigorous verification and validation activities and occur in a cyclical fashion between the activities. The iterative spiral moves from one phase to the next progressively improving the system performance.

The NASA Systems Engineering Handbook, SP-6105 [13], states, "The objective of systems engineering is to see to it that the system is designed, built, and operated so that it accomplishes its purpose in the most cost-effective way possible, considering performance, cost, schedule, and risk." Figure 4 provides a process to follow to help to achieve a successful SRR that will enable these objectives to be met.

Objectives and constraints are derived from the mission need and stakeholder expectations and documented. Objectives and constraints are met during each phase by defining and executing trade studies, conducting design and performance analysis, and verifying and validating results in an appropriate way for each phase.

The iteration process is executed until an "optimal solution" is achieved. Optimal is the point at which the objectives and constraints are satisfied such that there is diminishing improvement in cost, schedule, and risk with additional iterations. The three major functions on which the book focuses in terms of air and missile defense systems engineering include concept-of-operations development, architecture and design development, and requirements development and analysis.

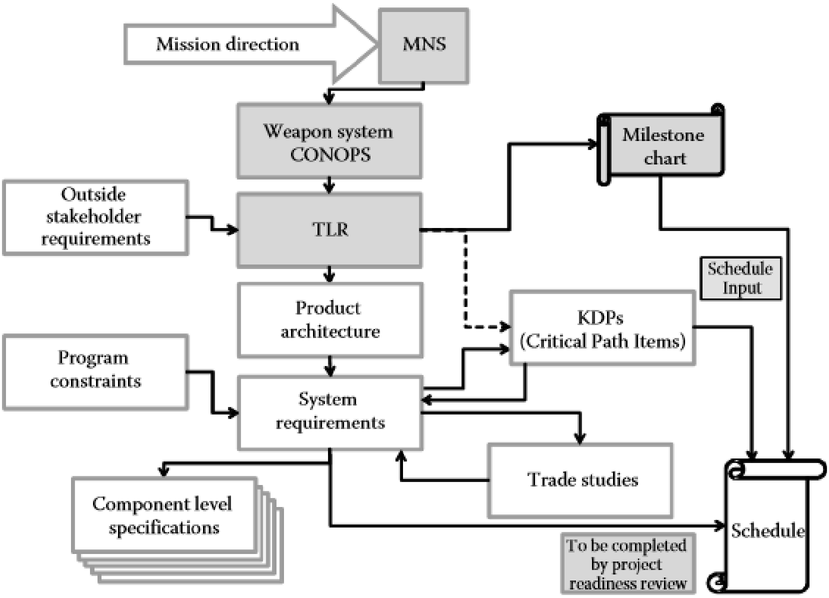

Figure 4. Systems Engineering process to the SRR

The program and project lifecycles really do offer an outline or a template for any large-scale technology development effort.

In order to implement the lifecycles, we must follow the systems engineering process (SEP). The SEP is the process for describing the path for mitigating program and project risks. The risks can be cost, technical, managerial, safety, part availability, logistics, and a lot of other things. The ISS development and construction continued to spiral out of control with never-ending budget overruns.

The overall ISS program, including its implementation, is now at risk because the space shuttles are being grounded immediately following the ISS construction finalization. Therefore, there will be no way to get crew and supplies up to and down from the station without relying on the Russian launch vehicles. Using the SEP has led NASA to the Ares I rocket development and the Orion space capsule to fill the void that will be left when the space shuttle program is grounded.

Systems Engineering Models

Figure 5 is the "standard V model" of systems engineering. It starts at the top of the left side of the V with a "top-down" view and this is where the "big picture" is generated. Here is where the idea of the overall architecture for the system begins to take shape. System-level design requirements are defined but at a very top level in the system functional review (SFR). Then, the path of the SEP flows down the leg of the V where individual components' design requirements are developed in the preliminary design review (PDR).

Once the design requirements of the complete system down to the component level are developed, then a critical design review (CDR) is held to make final adjustments to the blueprints before components are built and tested. The nomenclature here is important as any modern rocket scientist or engineer will often be working hard to meet the PDR or CDR deadlines. Afterwards, the CDR fabrication of components begins. The components are integrated together into a larger system of subsystems, and testing begins following the test readiness review. Following rigorous testing, the system goes through the system verification review where the analysis of all the data of the SEP to date is conducted to determine if the rocket is ready to move forward into operational status. If the analysis suggests that more development is needed, then the process starts over again at the top of the left side of the V.

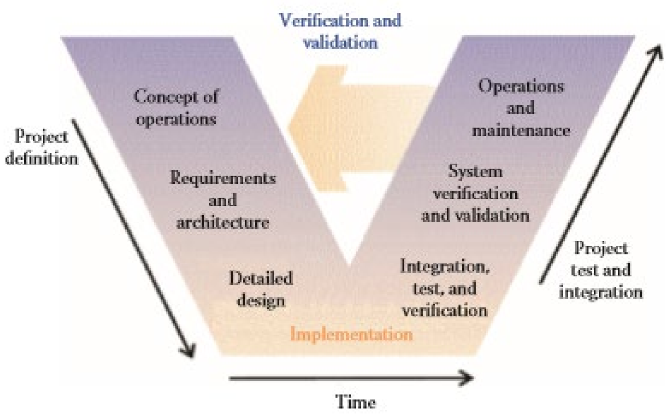

Figure 5. Standard systems engineering V is the template used by many programs to maintain best systems engineering practices (Courtesy of the US Air Force)

We should also note here that NASA is taking a step away from the V model and implementing a systems engineering engine (SE engine), as shown in Figure 6.

In the 1995 version of the NASA Systems Engineering Handbook, the V model was quite prevalent. In the 2007 version of the handbook, there is no mention of the V, and it is replaced by the SE engine. Even though the SE engine is not totally unlike the V, it is tailored more to NASA-type programs and projects. After all, the SEP is meant to be a living and updateable process and is not set in stone as the only way; rather, it is a template for a process.

The DoD still uses the V model, and so do many other organizations. The point of this section isn't to debate which one is better, but merely to show that the two methods exist. There are other SEPs, such as "spiral development," which is again possibly just another way to display the SEP. Figure 7 shows the typical spiral development process. These SEP tools should be implemented to aid the rocket scientists and engineers in the rocket development efforts. One or all three or even others might be implemented, but, in reality, it is the fact that a SEP is put in place for rocket development that is most important.

Verification and validation are accomplished during each cycle of design and development and for each segment, system, subsystem, element, and component to ensure the system meets the required mission objectives. Essentially, verification addresses whether or not the design satisfies the requirements. Each of the major system elements, the CONOPS, and the architectural design are verified against the requirements and must be consistent in solving the objective problem. Verification continues throughout the design and development process in a sequential manner. The CONOPS is verified against the mission objective; and the architectural design is verified against the CONOPS. Validation is usually defined by ensuring that the objective system is built correctly. This includes analyzing, inspecting and testing, and simulating the system prototype against real-world data to accomplish validation. In the case of an AMDS, it should be tested and simulated in a manner that represents the way it is intended to fight. This requires end-to-end testing and simulating, including all input items, interfaces, and performance requirement emulations. Expected and unexpected variations in the environment and intended target sets need to be explored.

The Verification and Validation Process

The SEMP should provide the systems' vision for verification and validation (V&V). V&V are interrelated and accomplished throughout the systems engineering process and include lifecycle support and sustainment. Together, V&V demonstrate that the AMD system meets the mission need, design goals, and stakeholder expectations, referred to as the mission objectives.

Verification

The SEMP should describe how verification is used to ensure that the AMD design satisfies the mission objectives. This is a continuing process that encompasses the verification of the CONOPS, the architectural design, and the requirements. The CONOPS, requirements, and architectural design verification process should ensure that mutual consistency between them is maintained throughout the program. The written process should emphasize that phase A and B verification activities strive to show that the right system design has been chosen before the detailed design proceeds in phase C, minimizing the chance that the wrong system will be designed. Verification also occurs later in the lifecycle when mission simulations, end-to-end tests, and other activities show that the AMD system has been designed correctly and meets the mission objectives.

Validation

Validation is an important risk reduction function that attempts to uncover issues before they become operational problems. Validation includes those functions, which ensure that the team builds the system correctly, by validating all requirements and verifying the architectural design against the requirements. The validation process includes the identification of the item and the method (analysis, inspection, or test) for validation. The process for the review and approval of the validation results needs to be explained. The validation activities of phases C and D show that the correct system is designed.

Often, it is not possible to accomplish such extensive testing in pure hardware. Simulations become more important when the system and the problem become more sophisticated. In many cases, simulation may be the only means to validate the design that leads to an entirely new program of modeling and simulation that should be handled in a parallel manner to the systems development and testing against real-world data whenever possible. Building and testing complex and sophisticated simulations have now become an inherent requirement for AMDS engineering. This not only includes a digital representation of the system but a commensurately complex digital representation of the intended targets and the associated environments. This is no small undertaking, and it will be a necessary part of the budget.

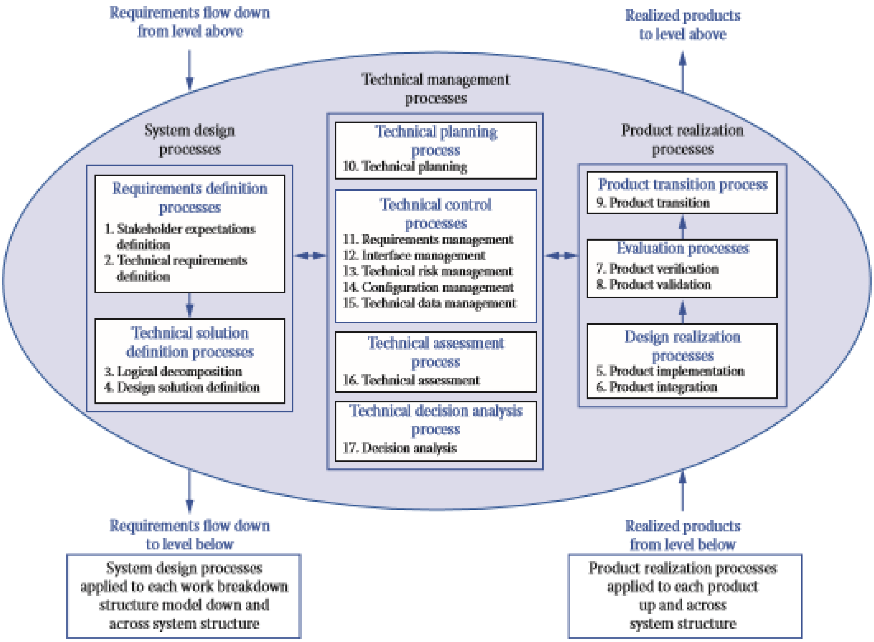

Figure 6. NASA systems engineering engine (Courtesy of NASA)

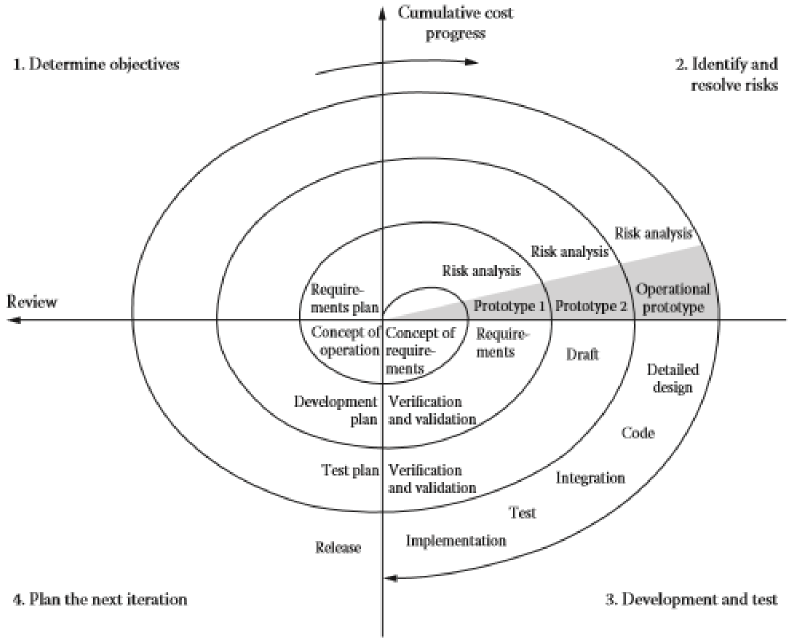

Figure 7. Spiral development systems engineering model (GNU free document license image)

The concept and process essentially include the layout of the SE plan that lends itself to VV&A and this is intended to maintain consistency, repeatability, and traceability throughout the program's lifecycle. Tools and methods apply to defining tradecraft and will subsequently contribute to VV&A, configuration control, traceability, and repeatability of results. The knowledge and skills of the workforce call for the placement of highly trained and experienced engineers with appropriate backgrounds in leadership positions and, most notably, skills in running integrated product teams (IPTs). A balance between a solid government team, a contractor, and laboratory team members must also be maintained.

The program should be structured such that the manager is able to matrix talent in/out of the IPTs as required. The components of a standard systems engineering process that can be applied to any project regardless of scope and scale include the following: