DISSERTATION

LINKING SYSTEM COST MODEL TO SYSTEM OPTIMIZATION USING A COST SENSITIVITY ALGORITHM

Submitted by

Danny Israel Polidi

Department of Systems Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Summer 2022

Doctoral Committee:

Advisor: V. Chandrasekar

Mike Borky
Thomas Bradley
Ketul Popat

ABSTRACT

LINKING SYSTEM COST MODEL TO SYSTEM OPTIMIZATION USING A COST SENSITIVITY ALGORITHM

Lack of adequate cost analysis tools early in the design life cycle of a system contributes to non-optimal system design choices both in performance and cost. Modern software packages exist that perform complex physics-based simulations. Physics based simulations alone typically do not consider cost as a factor or input variable. Modern software packages exist which calculate cost and can aid in determining the cost sensitivity to a chosen design solution. It should be possible to combine the system sensitivity to cost with the system sensitivity to performance. Methods and algorithms are needed to determine which components in a system would most significantly contribute towards the impact to the overall cost and which design alternatives provide the best value to the system. These methods and algorithms are needed during concept development to aid in system scoping and cost estimation.

In the bidding phase of a system design, most of the time is typically spent determining cost. System design trades are either seldomly done or abbreviated. This has not been preferable because the system design becomes locked into place long before significant trades have been performed. And the solution may not be optimal for either cost or performance.

This paper reviews the research performed and includes work in creating a cost model based on a set of questions & answers to drive system design, electronic design work applicable to the specific subsystem element FLO (Frequency Locked Oscillator), development of a standardized modular diagram and Work Breakdown Structure (WBS) for a RADAR System applied to military aerospace applications in the aerospace industry, and the development of a cost sensitivity algorithm. The goal of the research and cost sensitivity algorithm was to allow the system designer the ability to optimize for both cost and performance early in the system design cycle.

ACKNOWLEDGMENTS

My experience at Colorado State University has been very rewarding and full of great people who helped to encourage and guide me every step of the way.

First, I would like to express my gratitude to my PhD advisor Dr. Chandrasekar for his support, guidance, and coaching. I also thank my committee for brilliant comments, suggestions, and never-ending encouragement: Dr. Mike Borky, Dr. Thomas Bradley, and Dr. Ketul Popat.

I would also like to thank the Team at Galorath who shared tools and resources.

I also would like to thank my colleagues at work who understood the simultaneous challenges of work and school. They provided a shoulder to lean on, an ear for issues, and confidence when it waned: Rahul Dixit, Mike Crist, Monica Thomas, Bret Lowell, and many more.

A special thanks to Ingrid Bridge who never failed to answer the phone and kept me on top of all requirements and deadlines.

My deepest and most sincere gratitude goes to my family and closest friends without whom this goal could not have been reached. So often other activities took a back seat. Their unwavering understanding through all the challenges helped to convert my dreams into reality: Israel Polidi, Lydia Polidi, Ari Polidi, and Sylvia Goldhammer.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1 - Introduction

For the purposes of bidding new hardware and products, it is essential to understand cost of the proposed design as well as the cost of the alternatives. Commonly, customers work with a manufacturer to determine how a need can be solved with hardware. Then the effort becomes to provide cost for the identified solution. Commonly, as in the case of an airborne RADAR system for military aerospace applications, a costing exercise may take a month and potentially $1M for large systems. However, the most optimal solution may not actually be costed. Smaller trades occur later during the design phase. But the biggest trades at the front end, system design solutions, often are not performed due to a lack of time.

Limited available resources to create a solution is commonly what prevents the solution or adds risk during the life cycle of a Program. It is a common situation to have a need but with limited resources (time, money, personnel, etc.) to address that need. Certainly, with unlimited resources, solutions could be created which have been optimized for time, money, personnel, etc. With only limited available resources, it is desirable to have a means to determine where to employ those resources. Typically, there is no elegant way to direct with the most advantageous application of those resources. Research is being done and conducted in this area which attempts to formalize the process by which resources are utilized and are supported by some sort of return-on-investment calculation. In chapter 2, an example is explored where a large amount of existing data will be used to create an algorithm which will suggest to a Department Manager, responsible for a broad variety of Programs, an area where to focus resources for the most impact towards the success of a business.

In the bidding process, lack of time results in a less than optimal design solution and less than optimal cost. If tools were available which were based on the cost sensitivity for each cost driving element, then a system designer could make the best use of time including trades early in the system design life cycle and determine optimal performance and cost. The presentation here attempts to capture the research conducted in the areas of system modeling and system cost using a standardized block diagram and work breakdown structure as well as the generation of a cost sensitivity algorithm and tools to assist a system designer in determining solutions which are optimized both in terms of performance and cost.

Problem Statement

Lack of adequate cost analysis tools early in the design life cycle of a system contributes to non-optimal system design choices both in performance and cost. The goal is to develop cost algorithms for an automated tool/approach utilizing cost element sensitivity to enable a system designer the ability to understand the relative cost impacts of various decision/choices which affect system design early in the design cycle for an airborne based RADAR System for military aerospace applications

In the bidding phase of a system design, most of the time is typically spent determining cost. System design trades are either seldomly done or abbreviated. This has not been preferable because the system design becomes locked into place long before significant trades have been performed. And the solution may not be optimal for either cost or performance.

The common process is that Business Development representatives work closely with the customer to understand the needs and requirements. It is commonly during this interaction that a system design is determined. That system design is then flowed to Engineering for bidding purposes. Engineering takes the designated system design and develops cost based on historical similar-to programs. Commonly, this bidding process takes on the order of a month, including various approvals and signature cycles. In a sense, the system design has already been locked by the time the bidding, or costing, process has started.

Modern software packages exist that perform complex physics-based simulations. Physics based simulations alone typically do not consider cost as a factor or input variable. Modern software packages exist which calculate cost given a defined system design. These tools are used during bidding. But what has been lacking is the ability to understand the driving factors which most significantly impact cost and to be able to offer system design trades which may yield acceptable performance with respect to the requirements at an optimized cost.

Research Objectives

This dissertation aimed to summarize the research performed to facilitate in the optimized costing of a system. The goal was to highlight the key elements of research and publication which were applicable to the topic and then to develop a cost sensitivity algorithm to aid in optimized system design.

Commonly, cost model applications require a system designer to already have in mind a system design. Given the design, a cost model can provide solutions for cost. An example cost model is presented which demonstrates the ability to utilize mission objectives combined with a set of questions and answers to drive system design. This was a novel approach in that commonly cost model applications begin with a fixed design and then provide cost based on the given design. This paper demonstrates that research has been performed in this area and that it is possible to begin with mission objectives and through a small set of questions and answers, to determine a system design which can then be costed.

Commonly, for a new system design, as in the case for an airborne RADAR for military aerospace applications, the block diagram (which yields architecture) needs to be created. A common higher level block diagram which is applicable to any scenario within the limits of a military aerospace airborne RADAR would be preferable as a starting point. And, for any given case, the higher level, more general version, could be tailored for the specific application. This paper demonstrates that research has been performed in this area and that the more general block diagram has been created. This allows a system designer the opportunity to begin with the more general case and tailor the system for a specific application. And further, it was demonstrated that the same structure could simultaneously be used for both system modeling as well as for system cost calculations.

The key points that will be addressed in this research will be:

- An algorithm which suggests where to focus time as a resource for the most impact towards the success of a business
- A cost model based on a set of questions and answers to drive system design
- A cost model which includes an auto-generated Basis of Estimate (BOE) to aid in rapid costing
- A design approach for a Frequency Locked Oscillator (FLO) which serves as a subsystem block for an airborne based RADAR system for military aerospace applications to which can be applied all the same costing sensitivity algorithms as for the full system
- A standardized block diagram and corresponding Work Breakdown Structure (WBS) for an airborne based RADAR system for military aerospace applications
- Development of a cost sensitivity algorithm to aid in optimizing a system design and applied to a sample system cost model
- Application of the cost sensitivity algorithm on an airborne RADAR system for military aerospace applications using the standardized Work Breakdown Structure

Dissertation Overview

Research has been done in various areas of cost and system design. Those research elements contributed toward the goal of developing a cost sensitivity algorithm and tools which enable a system designer to better optimize for performance and cost.

Research advancements in cost for a system as well as in system design was explored. This research has focused on methods for collecting cost which were novel in that they illustrated how early questions and answers could yield design choices, as opposed to the more traditional method of defining a design and then costing it. Research has been performed regarding design alternatives for a Frequency Locked Oscillator (FLO) utilizing a discriminator method. This work was novel in that it incorporated multidisciplinary technology in analog, digital and software engineering. Referring to Figure 1, a subsystem block within the Exciter subsystem has been labeled FLO, for Frequency Locked Oscillator. This is, of course, very specific and would not necessarily be applicable for every airborne RADAR application but is none-the-less included here, again for illustration of a larger concept, to be explained as part of further research.

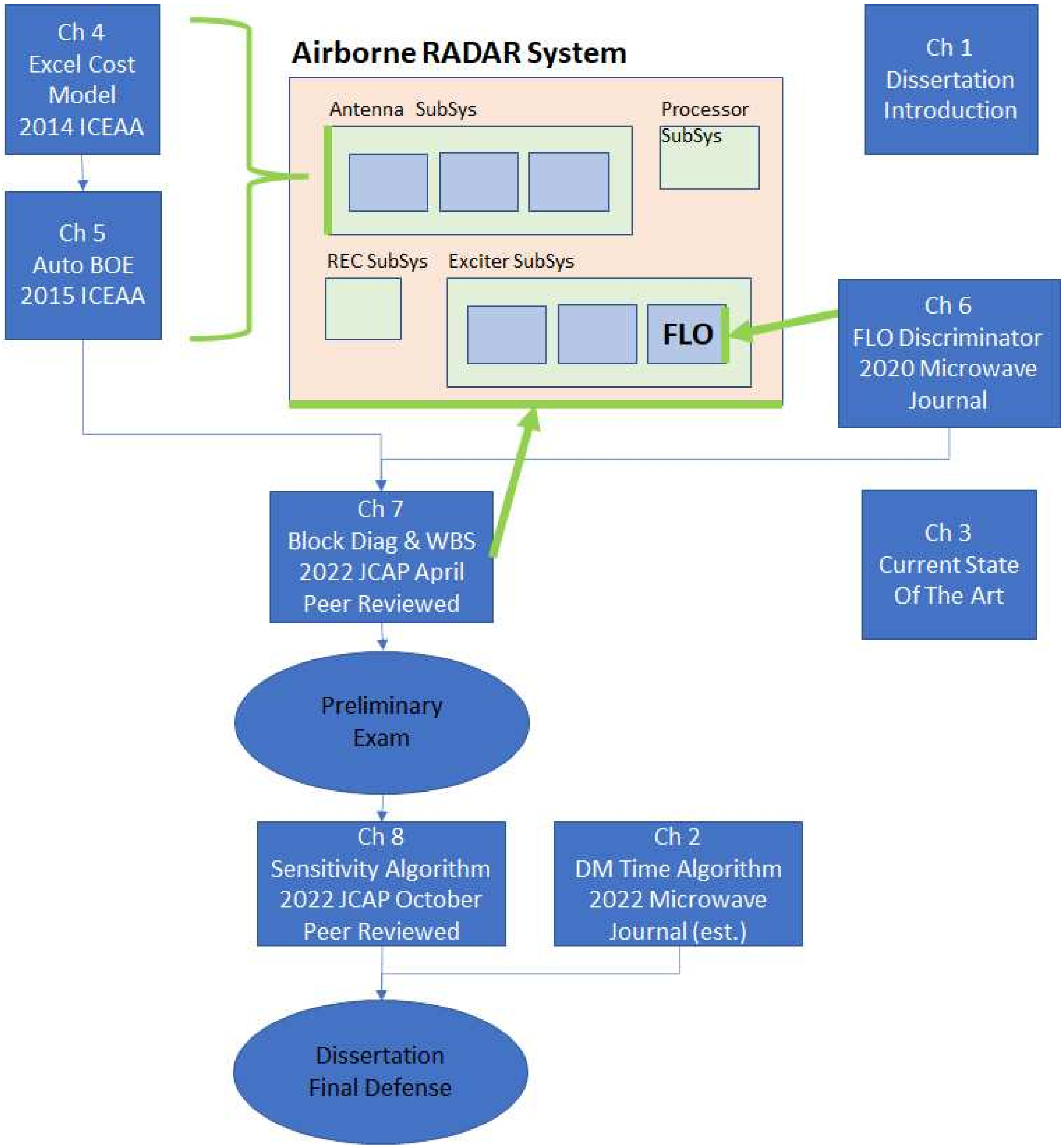


Figure 1 - Overview Of Dissertation.

The figure illustrates the intersection of those various research efforts. An explanation of the elements in Figure 1 is provided.

The airborne RADAR is a system comprised of various subsystems. It can be seen in this simplified illustration that it is comprised of four main subsystem elements: Antenna subsystem, Receiver subsystem, Processor subsystem, and Exciter subsystem. And both the Antenna and Exciter subsystems are further decomposed into various sub-blocks. This is true for all the subsystems and further details will be provided, but for purposes of illustration in Figure 1, those sub-blocks are only indicated for two of the subsystems.

In chapter 2, the discussion focuses on time as a resource **[1]**. It is a common situation to attempt to achieve goals but there is a limit to the available resources. If unlimited resources were available, then certainly any goal could be achieved. It is the limitation of those resources which cause risk to achieve a set of goals. Take, for example, the resource Time as it is applied to the design of a System. For example, if it typically takes 18 months to design a particular System, but the need is to have the System designed in 12 months, it may be possible. But certainly, there will be some risk in achieving a design with a limited amount of time. The real-life example that will be explored throughout this chapter will be that of a Department Manager responsible for a product line which is used across a broad variety of programs, or contracts.

In chapter 3, the discussion focuses on the current approach within industry where after a given design solution has been determined a cost analyst would then provide a cost estimate based on that given design. It can be considered a unidirectional sequence of events lacking a feedback mechanism. Some tools do offer some rudimentary ability to visualize sensitivity of the cost parameters. With those cost tools, after a design has been selected, the system in terms of cost can then be modeled. And, after being modeled, a cost analysis can indicate something about sensitivity. The research referenced in Chapter 3 **[2]** refers to a cost tool which is organized as a set of questions and answers. The questions have to do with the system mission, and based on the answers, a design is selected. This, in of itself, is a novel approach. There are no commercially available cost model tools which produce cost based on operational objectives. In addition, because the approach incorporates answers to the questions, there cannot be an infinite number of possible outcomes. The answers could be organized in order of impact to the overall system.

In Chapter 4, the discussion focuses on the research entitled "Building A Complex Hardware Cost Model for Antennas" **[2]** which was presented at the 2014 annual Symposium for ICEAA, the International Cost Estimating and Analysis Association in Denver, CO. The research was again presented at the 2014 Southern California chapter of ICEEA for the bi-yearly workshop. This work was limited to an Antenna subsystem. The research included a parametric cost model. This work was novel in that it illustrates the ability to begin an antenna system design with mission objectives, and based on those objectives, to utilize a small set of questions and answers which drives design, and subsequently cost for the overall system. Included in that presentation was a brief reference to cost sensitivity as it relates to each cost driving element. This also was novel because it allows a system designer to better understand the system elements which drive cost and could then be used to influence system design.

In chapter 5, the discussion focuses on the research entitled "Development of a "Similar-To" Basis Of Estimate (BOE) Generation Tool Used in Conjunction with a Complex Parametric Antenna Cost Model" **[3]** which was presented at the 2015 annual Symposium for ICEAA, San Diego, CA. This work was limited to an Antenna subsystem. This work built upon prior work by expanding the earlier cost model to include a feature for the auto generation of a BOE, or Basis Of Estimate. This effort demonstrated a novel approach to provide a BOE with an exportable single similar-to program data set while the cost calculation was performed utilizing a large proprietary data set with a parametric model.

In chapter 6, the discussion focuses on the research article entitled "Digital Control and a Delay Line to Frequency Lock an Oscillator" **[4]** which was published by the Microwave Journal magazine, March 2020. The novelty presented here was a multidisciplinary approach to a solution AND an approach whereby a signal was stabilized using its own frequency, delayed in time, to correct for frequency drift rather than with the commonly required additional frequency source. In context, the FLO is a subsystem block, within an Exciter subsystem, within a RADAR system. This subsystem block is required to demonstrate that the cost sensitivity algorithms apply at both the macro and micro scale in a generic system. This work is fundamental to the proposed research in that it may be used to demonstrate the cost sensitivity algorithm's applicability both at a micro and macro level.

In chapter 7, the discussion focuses on the research article entitled "Foundation of Structured Architecture, System & Cost Modeling" with early work first introduced at ICEAA Symposium 2020 **[5]** and subsequently modified, peer reviewed and published by JCAP (Journal of Cost Analysis and Parametrics) April 2022 **[6]**. The research describes the literature search performed to identify a standardized block diagram and WBS, or Work Breakdown Structure, to be used as a foundation for an airborne RADAR system for military aerospace applications. The research discovered no generic block diagram or structure had previously been identified. In the absence of a standard, the available literature was used to construct a suitable solution which could function for all examples and where the available examples could be considered a tailored version of the more general form. The paper contributes to the body of knowledge by providing a general form for a block diagram for an airborne RADAR for military aerospace applications, a general form for a WBS, a system model using Rhapsody (including guidance on how to create it), and a cost model using SEER (including guidance on how to create it).

In chapter 8, the discussion focuses on the final phase of research. The article "A System Engineering Approach Using Sensitivity Analysis For Reducing System Cost" **[7]** has been peer reviewed and accepted for publication by JCAP for October 2022. The effort was to tie the pieces together and make advancements regarding the sensitivity of cost variables to influence the design choices early in the design cycle of a Program. To do this, it was necessary to finish the work of creating a cost model using a Commercial Off The Shelf (COTS) cost package based on the generic block diagram presented, consistent with the WBS and system model. Then, to use the cost model to generate sensitivity data for each cost element of the model. The data demonstrated that it was possible to create an algorithm which allows a system designer to make trades on design parameters which will most significantly impact cost. In other words, identify where a designer can get the greatest impact in design trades to influence cost most significantly.

Chapter 2 – Maximization Of Available Time: Department Manager Example

Introduction

It is a common situation to attempt to achieve goals but there is a limit to the available resources. If unlimited resources were available, then certainly any goal could be achieved. It is the limitation of those resources which cause risk to achieve a set of goals. Take, for example, the resource Time as it is applied to the design of a System. For example, if it typically takes 18 months to design a particular System, but the need is to have the System designed in 12 months, it may be possible. But certainly, there will be some risk in achieving a design with a limited amount of time. The real-life example that will be explored throughout this chapter will be that of a Department Manager responsible for a product line which is used across a broad variety of programs, or contracts.

In this example, the DM is responsible for the Receiver Product. Within the Organization, if a Receiver Subsystem is required, then it falls to the Receiver DM for execution. The DM is the Product Owner. In this context, the Department Manager is responsible for the full life cycle of a product including all aspects of bidding, design, development, and production. In addition, the Department Manager is responsible for all resources including personnel, facilities, capital equipment, etc.

Division of Authority

The hierarchy of a company may vary, but in general the structure is consistent while the titles may vary. For purposes of this chapter, the hierarchy is representative of a specific example. Therefore, the hierarchy is defined such that the reader has context. The entire organization can be divided into three main categories: Program Manager, Department Manager, and Team.

Program Manager:

The Program Manager is involved early in the life cycle of a Program. The Program Manager is the main interface between functional line management and the customer. The PM understands the needs of the customer and relays those needs to functional line management for bidding and scope purposes. During execution of the Program, the PM continues to interface between the customer and functional line management to ensure the Program is executed successfully and that the needs of the customer are being met continuously.

Department Manager:

The Department Manager is involved early in the life cycle of a Program. The Department Manager represents all functional line management. The DM receives requirements from the PM. The DM assigns personnel to bid a Program. The DM approves all bids. During execution, the DM assures that the appropriate personnel are assigned for the various tasks. The DM monitors earned value and applies assistance to guide the Program towards success. This may take the form of assigning additional resources, ensuring training and processes are being followed, monitoring work products, reporting Program status to Management, etc. Ultimately, the success of the Program lies within the purview of the DM.

Team:

The Team consists of the personnel charged with the day-to-day execution of the Program. The Team is formed after the Program has been awarded. The Team prepares the schedule, manages the budget, performs the operations, and drives the Program from award to successful completion. If the Team identifies shortcomings in personnel or training, the Team enlists the help of the DM to resolve those shortcomings. The Team takes ownership of the Program execution while at the same time understands that ultimately the responsibility lies upon the DM. The Team works with the DM providing feedback and status.

Resources: Time, Money, and Personnel

The DM is responsible to assign resources to a Program with the goal of successfully executing the Program. Although the practical day-to-day responsibilities fall upon the Team, the DM has the higher level of responsibility, making sure things are set up for a successful execution. As is generally the case, resources are limited. As a result, the DM is faced with the responsibility to allocate the limited set of resources to the Program. For purposes of this chapter, it will be considered that all resources fall under three categories: Time, Money, and Personnel.

Money:

When a Program is bid, the DM is tightly coupled into the bidding process. Ideally, a current bid is significantly similar-to a legacy Program. And, when determining the cost for the current bid, the actual dollars spent on the legacy bid can be either used directly or scaled using appropriate Key Size Metrics, or scaling factors. In **[7]** the author addresses the need to create KSMs to vary a scaling factor as part of a cost analysis. In the case of a bid for a new Program, the same concept applies. In the event a Program is captured, money would then be allocated to the Team to execute the Program. Typically, the money allocated to the Program for execution is not equal to the money bid. This is a result of the bidding process which includes cost negotiations. In addition, a portion of the funding is held by the Program Manager in the form of Management Reserve, or MR. It then becomes the responsibility of the DM to understand the difference between what was bid and what was allocated or received. As part of the personnel responsibilities, the DM assigns a Cost Account Manager, or CAM, to administer the funds. Generally speaking, the CAM ensures the Program stays on track in terms of cost.

Personnel:

The DM has responsibilities for personnel. This includes hiring, training, assigning mentors, verifying applicable certifications, etc. The simplest of situations would be when there is a long-term project and there exists insufficient personnel to execute the Program. The solution could be to hire more people. Frequently, however, is the situation where there is a need for a certain number of people and the approximate number of people are available, however, the available skill set is not aligned with the required skill set. This would require either retraining of personnel or replacement. In any case, the responsibilities for aligning the needs of the Program with the appropriate personnel falls upon the Department Manager.

Time:

There are two significant aspects of Time worth discussing. The first to discuss is the Time allocated for Program execution. When a Program is bid, part of the legacy information includes the actual time it took to execute the legacy program. This duration is used along with KSMs to determine the required time to execute the current Program. Frequently the allocated time to execute is less than the ideal duration. This situation requires the identification of risks to the program. The DM is responsible to identify and quantify the risk. Typically, a risk register is used.

Time can refer not only to time for execution of a Program, but also can refer to how will the DM spend their time. A DM is not in the trenches running a simulation or summarizing test results. Instead, they are more like a firefighter looking for how to solve problems at a higher level. So how should it be decided where to put that time?

The second aspect of time to discuss, and is the primary focus for this chapter, is the time for the DM. The DM arrives at work on Monday morning and considers, “There are many programs, many phases, many dollar values, lots of performance metrics, where do I spend my time?” Frequently, the Program with the loudest voice commands the most attention. This concept, limited time (as a resource) and where to apply that resource is a common issue. It is a difficult issue to address. And there are few effective means to address the issue.

Current Method

The Department Manager has responsibility for a broad variety of Programs. Some Programs in terms of dollar value are large while others are small, with every variation in between. Some Programs in terms of phase are nearly done, some are just beginning, with every variation in between. The Program types can be classified in several varieties. Some Programs are research and development, in other words very early on in the lifecycle of a Program. Other Programs are production Programs with deliverables, in other words very mature and developed Programs. The Programs have a variety of performance with some Programs performing within schedule and budget and other Programs which are in serious trouble from a perspective of earned value. With limited resources, such as the Department Manager's time, there exists a need to understand where to focus that resource, or time.

As an example, Monday morning the Department Manager gets to work and has an 8-hour day in front of him or her. With such a large variety of programs, with different values, at different phases, and different earned values, etc., it is no simple matter for the Department Manager to understand where to spend those 8 hours. Without tools, the Department Manager has no means by which to determine where to focus those resources.

There are external forces which complicate the decision-making process. For example, Program A has a Chief Engineer with considerable motivation. The Chief Engineer for Program A may constantly be on the phone with the Department Manager attempting to get the Department Manager to focus on Program A. In the absence of context, it most certainly may Seem that Program A needs immediate attention. However, when all things are taken into consideration, it may be that Program A is performing quite well and should be left unattended by the Department Manager. While at the same time, Program B has a Chief Engineer who for whatever reason handles all the problems and issues internally. And might need some assistance. It may be that the Department Manager ought to spend the 8-hour day focused on Program B. But without some method to determine where to focus resources the Department Manager typically assigns resources in a somewhat random fashion.

Above the Department Manager is the Center Director. On a monthly basis the Center Director requests from all the Department Managers a status report. Because the book of business for each Department Manager is so large it is impractical to present status on every Program so the Department Manager hand selects several programs for which to report status. The process to determine which of those Programs to select is also somewhat random. The Department Manager selects three Programs (Program1, Program2, Program3), prepares status charts and presents them. The Center Director has no way to understand how the Department Manager has determined which Programs to report on. Meanwhile the Center Director has heard that Program4 is in trouble and was expecting that the Department Manager was going to report some status on Program4.

Faced with this real-life scenario it was considered that there must be a way for a Department Manager to determine where to apply resources and where to focus attention such that there would be a maximum impact to the performance of the business. There must be a transparent means to collect a large amount of data across many Programs, generate a performance algorithm which implies where to focus attention, and to do it at the push of a button in real time with the most recent data available. At the same time, the Center Director should have the same transparent visibility to understand where within that Department are the problem areas. And when the Department Manager comes to present at a monthly status review, it would be understood from both sides which Programs will be presented. There should already be an agreement about what will be presented prior to presenting even the first status slide.

Future Vision Of Solution

There is a large amount of program data available in a company. The data could be collected manually every time there is a need to perform the performance algorithm. But for this real situation, there exists a company database and it is full of very valuable information related to the Programs. It becomes a matter of determining which elements of that database are relevant and pull those pieces of information. For this discussion it will be assumed that the database exists and all references to available data will assume that the data was obtained from there.

Given that there exists a database of information, it should be possible, given access to a large quantity of real-time Program data, to be able to pull relevant Program information, manipulate the data with a performance algorithm, format the data and then present a summary of Program data in such a way as to offer to the DM the visibility where to engage with the various Teams for successful execution. It should answer the question: If the DM has 8 hours, what is the most effective use of those 8 hours?

At-A-Glance Cartoon:

To begin, a cartoon is created which describes the end goal. The cartoon will represent all the Programs and will be organized in such a way as to offer a recommendation on where the DM should engage.

A DM is responsible for a set of Programs. It should be possible to pull from the database a simple list of those Programs, see Figure 2 - **List Of Programs**. Of course, there are a variety of Programs. Some Programs are IRAD, some are production, some are development, etc. But for purposes of this illustration, the nature of the Programs will be considered homogeneous. In other words, it will be considered that every Program falls under the same category, they are all the same type of Program.

Program1
Program2
Program3
Program4
Program5

Figure 2 - List Of Programs

In Figure 2, five Programs are listed. For the moment, consider this represents the full book of business for the DM.

Every Program has a dollar amount associated with it. It is the value of the Program. It should be possible to collect, in addition to each Program name, the Program value. For purposes of a graphical representation, it is unnecessary to display the numerical value of the Program. To be useful, only the relative relationship between Programs is of importance. In addition, once the Program values are understood, it should be possible to sort them from greatest to least value, see Figure 3

$	$	$	$	$	$	Program1
$	$	$	$	$	$	Program2
$	$	$	$	$	$	Program3
	$	$	$	$	$	Program4
	$	$	$	$	$	Program5

Figure 3 - Programs Sorted By Value

It can be seen in Figure 3 that the symbol “$” is used to represent relative value and that Program1 represents more value than Program5. This is typically the extent to which a DM might organize Program priority. Because Program1 has more value than Program5, Program1 will receive the most attention. This, of course, lacks any consideration of the other factors which will now be capitalized upon.

The next attribute to consider is the health of the Program. The Program definition of Good vs. Bad is critical and will be addressed in a subsequent section of this chapter. But for the current discussion, it is sufficient to accept that a Program falls within either of two categories, Good or Bad. Using information from the database, an algorithm will be applied to the Program to evaluate the data and determine if the Program is Good or Bad. Once again, those definitions (Good/Bad) will be reserved and developed in detail in a later section of the current chapter. The Program health data could then be graphically displayed along with the Program names and value, see Figure 4.

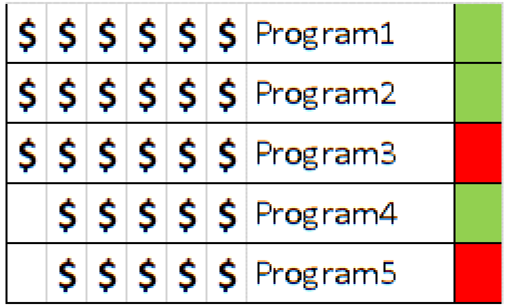


Figure 4 - Program Health Score

In this case, in Figure 4, regardless of the definition for Good vs. Bad, Program3 & Program5 are designated as Bad and color coded red while the others are designated as Good and color coded green.

Another aspect to consider regarding the determination of Program health is the interval of time over which the Program health is being evaluated. In other words, to code a Program as green and Good represent the health of the Program over some interval of time. This evaluation may very well be the Program's health today, or this week, or this month. It is important to establish the period over which the rating period applies. In this example, the most meaningful period over which to evaluate a Program is monthly. This is due to the rate at which the data is being collected. For this hypothetical company, earned value is typically collected weekly and reported monthly. As such, the rating period will be considered per month.

The health of a Program for a given interval of time should be considered in context of an overall performance. It is important to continuously consider how a Program is performing. However, to only consider how a Program is performing within the current interval of time would necessarily require a DM to jump erratically between Programs without fully understanding the larger view. This is the point of a performance algorithm, to offer a thoughtful view of where to apply focus. In short, it is the difference between a reactive and proactive response to Program health.

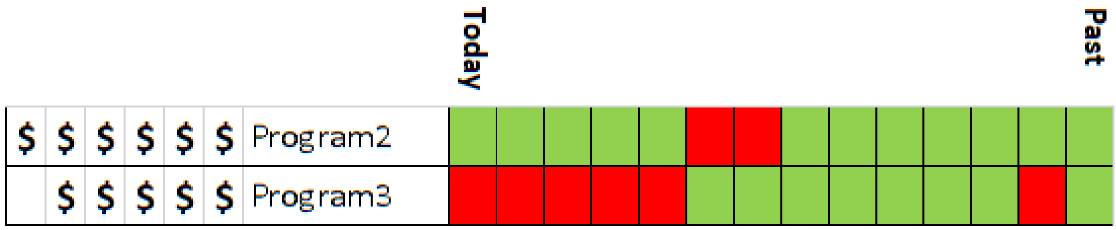


Figure 5 - Yearly View.

Figure 5 indicates a full year of Performance for two Programs. The current rating period is on the left and the historical performance proceeds backwards in time as the graphic moves to the right. The figure indicates that Program2 has more value than Program3. Program3 has had Bad performance for the most recent five months of recorded data. Basing a decision simply on the value of the Programs would have resulted in attention being focused on Program2 rather than Program3. However, it is clear from the figure that Program3, based on performance, is certainly in greater need of attention than Program2.

One final aspect to consider for the at-a-glance graphic is the phase of the Program. In other words, it is useful to understand if the Program has just started, is about to finish, or is in some state somewhere in between. Consider the graphic represented in Figure 6.

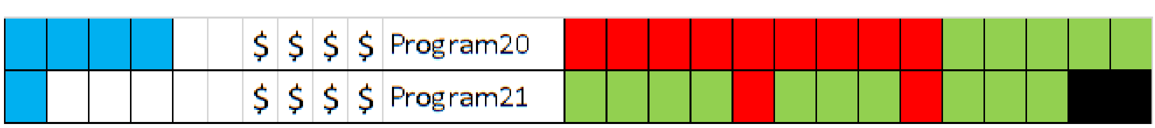


Figure 6 - Phase Of Program.

In Figure 6, the blue shaded portion represents completed work. In this case, Program20 is in the final phase and will be completed relatively soon while Program21 is in the early stages. Value is roughly the same with Program20 ranked higher indicating slightly more value. Program20 has had many months of Bad performance including the current month.

From this perspective, it can be argued that Program20 should receive focus because of its consistently Bad performance. However, it could also be argued that because Program20 is nearly completed, there may be other Programs which are in an earlier phase and would benefit more. In other words, Program20 is almost done. It could be argued that no amount of focus at this point could alter its trajectory. And effort focused here might not be as well spent as effort focused somewhere else.

Finally, all the elements are combined into a full at-a-glance view, see Figure 7. Initially, a DM might consider focusing attention on Program1 because it is the highest value. However, based on its performance, it consistently demonstrates Good health. There can readily be seen that four Programs consistently perform poorly: Program3, Program8, Program20, and Program28. However, by observing the phase of the Program (the blue shaded area), of the four poorly performing Programs, three are nearly done: Program3, Program20, and Program28. It could be argued that these Programs are so mature, that any amount of focus at this time is too late. What remains is Program8. Program8 is a relatively young Program (blue shaded area), is relatively large in value (resides to the left of the chart) and has been performing poorly (four months of red). With this information, the best recommendation on where a DM should apply focus would be on Program8.

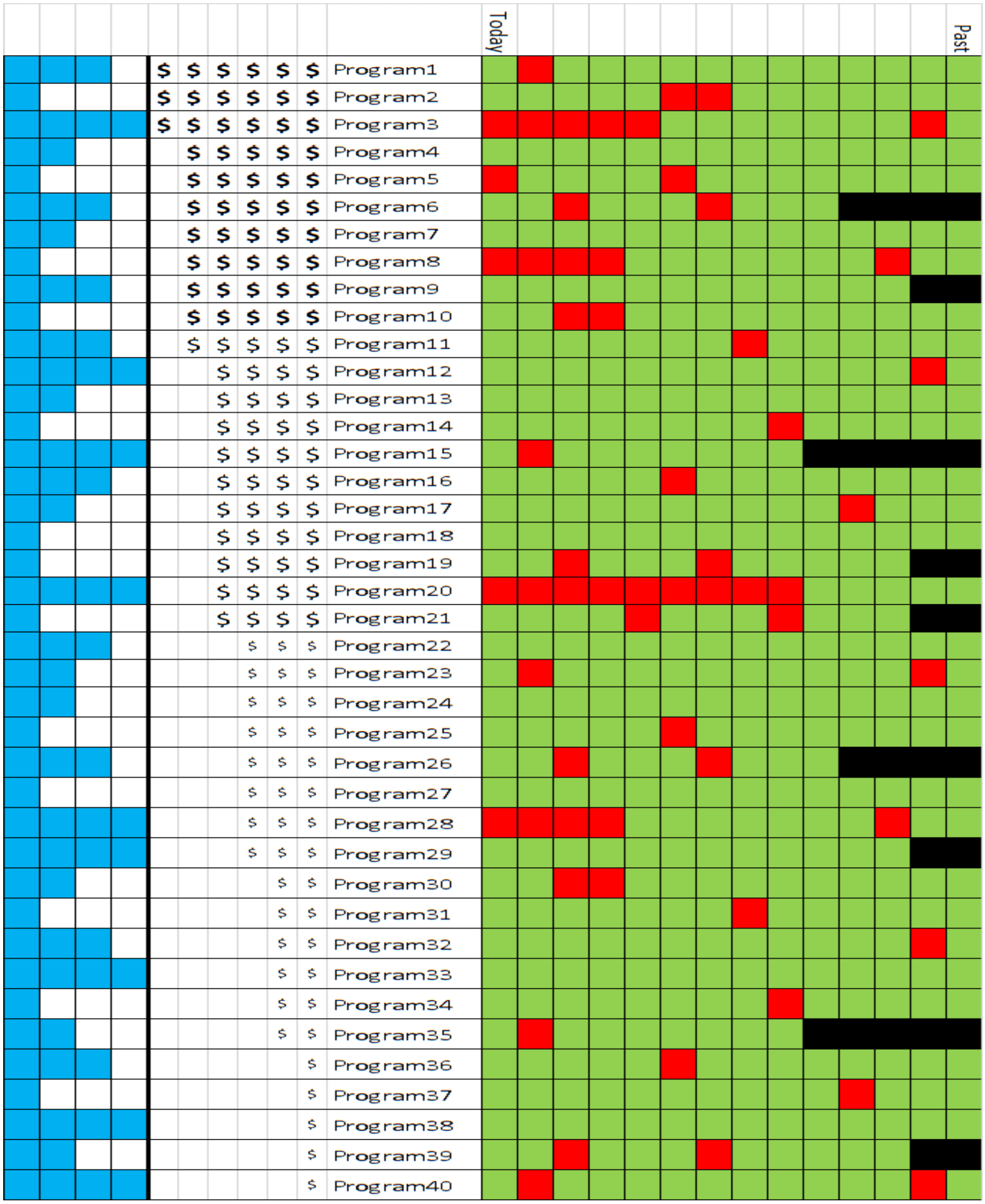


Figure 7 - Full At-A-Glance Concept

This is the value of the performance algorithm. The DM's time is a limited resource. To impact the overall business most effectively, an understanding of the broad landscape of the business is required, rather than to arbitrarily decide "focus here is better than focus there". The Program Office for Program4, for example, may demand attention. And it may be very justifiable to deny that attention in favor of another Program, Program8 for example. With metrics such as these, a DM would have the ability to make better decisions regarding Time as a resource.

Definition of a Healthy Program, Good vs. Bad:

The definition of Good vs. Bad health is critical. For this application, it is a binary determination. In other words, the health is either Good or Bad with no grey area in between. This should be a reasonable requirement because the goal is to determine if a Program either needs immediate attention or does not. To have a grey area in between becomes meaningless.

The database will provide a set of data used to determine the Program health. The items from the database are referred to as Criteria. Examples of Criteria may be CPI, SPI, TCPI, etc. It is recognized that the suggested criteria are consistent with ANSI EIA 748, the Standard for Earned Value Management Systems. For the moment, the specific criteria are not required. Only an understanding that Criteria exists and can be pulled when needed, see Figure 8. In this case the Criteria are listed in no particular order.

Criteria
Criteria1
Criteria2
Criteria3
Criteria4
Criteria5

Figure 8 - Set Of Health Criteria

The next activity is to apply a weighting factor. For example, Criteria1 may be associated with budget while Criteria2 may be associated with schedule. Both Criteria are important. But in terms of running the business, a determination will need to be made as to which Criteria is more important. And, based on the relative importance, weights can be assigned.

Criteria	Weighted Factor
Criteria1	5
Criteria2	15
Criteria3	20
Criteria4	35
Criteria5	25
	100

Figure 9 - Weighted Criteria

In Figure 9, the Criteria have been assigned Weights. The Weights would necessarily sum to 100%. Again, because the Criteria themselves are for illustration, the Weights are arbitrary and used for purposes of algorithm development.

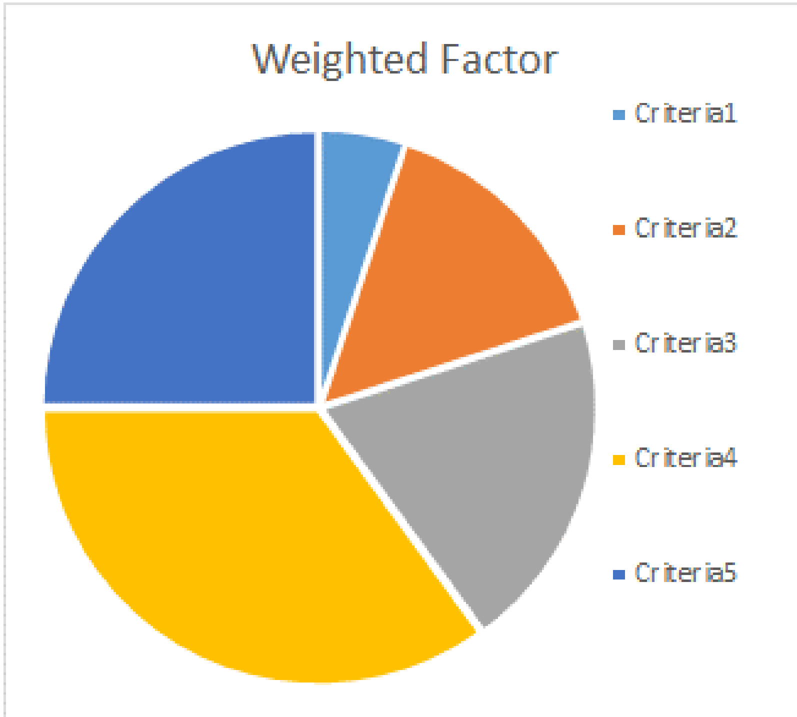


Figure 10 - Weighted Factor Graphical Display

Because the Weights have been assigned, they can be graphed, see Figure 10. It is easy to see from the graphic in Figure 10 the relative impact of the various Criteria. The greater the Weight, the larger the angle of the pie slice.

The goal now is to determine a Score for each Criteria for this Particular Program. To understand the concept, the discussion will proceed graphically and then transition to a numerical representation. To illustrate a score graphically, it is necessary to assign a radius for a particular pie slice which corresponds to a Criteria Score. A radius of zero would indicate a perfect score. A radius of one would indicate a worst-case scenario. In this way, the shaded area can be controlled. As an example, the graphic which appears in Figure 10 has all pie slices with a maximum radius of one. This corresponds to every Criteria having a worst-case scenario. Clearly an undesirable condition.

By contrast, Figure 11 shows an example of one Program with various Criteria Scores. Criteria3, with a small radius, is indicating a relatively Good Score while Criteria1, with the largest radius, is indicating a relatively Bad Score. The angle of the slices indicates the Weight of the Criteria. In this way Criteria4 is identifiable as the most Weight and is performing better than Criteria1 which has very little Weight. In general, about 25% of the circle is shaded which indicates overall that the Program performance is not perfect, but certainly is leaning much closer to Good than Bad.

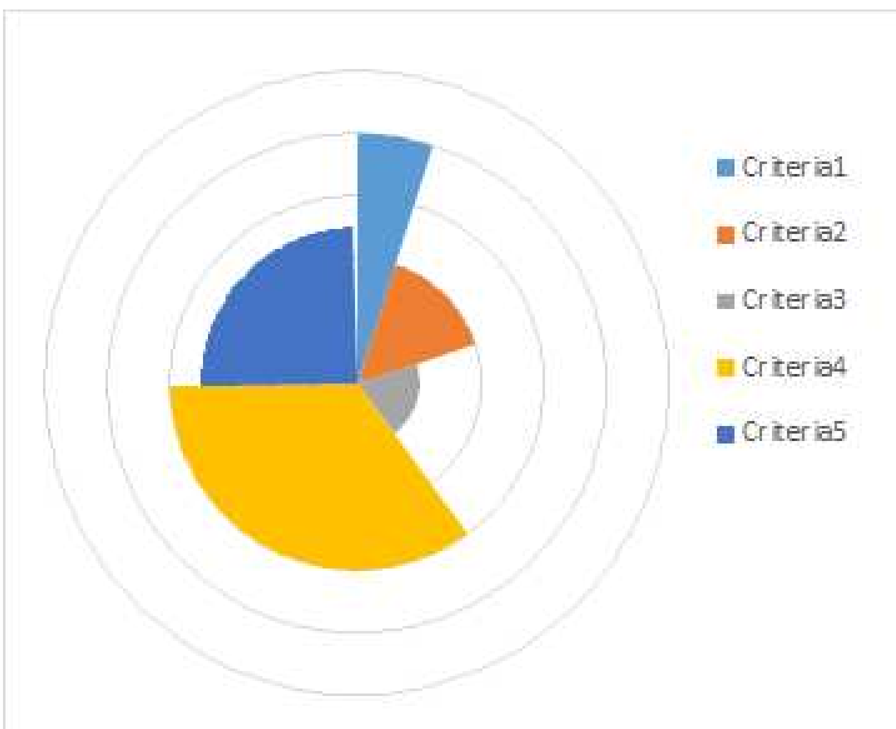


Figure 11 - Example Criteria Scores for One Program

With the graphical understanding of Figure 11 , the numerical calculation of the score is presented. As an example, consider that Criteria1 represents a Cost Performance Index, or CPI. As is generally understood, CPI can have any value above zero and is generally within the range of 0.5 to 1.2. In terms of a color code, a CPI score usually has three ranges:

0.97 – 1.00 is Green

0.94 – 0.96 is Yellow

0 – 0.93 is Red

A raw CPI value would need to be translated to a corresponding Criteria Score. To do this, consider Figure 12. Highlighted in yellow is the raw CPI value, in this case 0.8. Although CPI could be 0.5 or less, it certainly serves to indicate a problem long before it drops that low. In this case, the trip point to indicate a problematic situation has been set arbitrarily at 0.75, as indicated in the figure. The ideal raw value for CPI is set for 1.0. This serves as the range of raw CPI values. At the same time, the worst-case Score is set at 100 and the best-case score is set at 0. With these limits, and using a simple linear extrapolation, the CPI value of 0.8 has been converted to a Score of 80.

Criteria1 (CPI)	
worst raw value or Trip Point	0.75
worst corresponding score, or Trip Point Score	100
Best Raw value for this Criteria	1
Best corresponding score for this best raw value	0
This Criteria raw value	0.8
Therefore this Criteria Score	80

Figure 12 - Translation of CPI value to Criteria Score

As can be seen in Figure 12, what is needed for translation is the raw Criteria value, limits for the raw value (which would vary depending on the specific Criteria), limits for the Score (which would be the same for all Criteria), and then a simple linear extrapolation. The same raw value to Score translation must be performed for each Criteria.

One such set of data, as an example, might be for the values indicated in Figure 13. In this case, the Program health is described by five Criteria. Those Criteria had some raw value which was then translated to a health Score. Once again, the larger the score means more shaded area in the circle graphic which corresponds to Bad performance. The Scores indicated in Figure 13 are also represented graphically in Figure 11.

Criteria	Score
Criteria1	80
Criteria2	40
Criteria3	20
Criteria4	60
Criteria5	50

Figure 13 - Example Scores

The next step is to use the Criteria Scores to calculate the area in the circle for the unshaded region. As discussed earlier, the unshaded region corresponds to Good health. So, it is desirable to have as small a number as possible. In this case, the unshaded region of Figure 11 was calculated to be 0.7475. In other words, approximately 75% of the circle is unshaded.

The question remains as to what value would trip the indicator between Good and Bad, as was indicated in Figure 4. Here it is important to understand that the trip point is arbitrary. Ideally, all the calculations for all the Programs should be performed with the Good vs. Bad trip point remaining as a variable. Then, the variable should be adjusted to create a graphic which is meaningful. To put it another way, if the trip point was set too low, then all the data in Figure 6 would be coded Bad, or red. If the trip point is set too high, the reverse would be true. It really depends upon the full view of all Programs over a full range of time periods.

Just as an example, if the trip point between Good and Bad were to be determined to be 0.70, then a set of example Programs could be like those indicated in Figure 14.

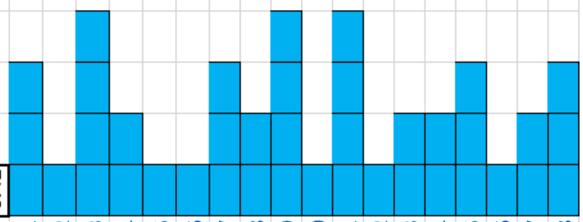
Figure 14 - Program Score Ranges

And finally, Figure 7 represents a sample of all Programs. From this figure all Programs are arranged by value, four Programs are consistently performing Badly and three of those Programs are nearly at completion. Therefore, Program8 rises to the top of the Programs which should receive the immediate attention of the DM to impact the overall business performance most effectively.

Data

To be meaningful, data must be collected over a long period of time. The performance algorithm suggested here is for a years' worth of data. Historical data may be available. However, real-time data most certainly is available. The Excel based tool to analyze the data has been created but is currently populated with sample data which has been created using a random number generator, see Figure 15.

this number is my trip point: 0.42



							0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-10	-11	-12
Program1	$	$	$	$	$	$	0.65	0.64	0.72	0.65	0.76	0.70	0.70	0.77	0.52	0.56	0.59	0.40	0.71
Program2	$	$	$	$	$	$	0.34	0.71	0.52	0.55	0.68	0.50	0.56	0.61	0.37	0.73	0.37	0.54	0.79
Program3	$	$	$	$	$	$	0.80	0.58	0.96	0.42	0.29	0.52	0.75	0.56	0.49	0.61	0.61	0.39	0.73
Program4		$	$	$	$	$	0.68	0.61	0.55	0.45	0.64	0.67	0.44	0.32	0.34	0.80	0.43	0.68	0.57
Program5		$	$	$	$	$	0.37	0.33	0.62	0.58	0.86	0.81	0.48	0.53	0.54	0.24	0.68	0.50	0.49
Program6		$	$	$	$	$	0.73	0.63	0.64	0.46	0.56	0.52	0.73	0.89	0.35	0.79	0.61	0.80	0.73
Program7		$	$	$	$	$	0.64	0.69	0.71	0.74	0.61	0.43	0.41	0.76	0.38	0.55	0.54	0.29	0.32
Program8			$	$	$	$	0.71	0.46	0.66	0.51	0.58	0.46	0.77	0.36	0.79	0.72	0.68	0.65	0.62
Program9			$	$	$	$	0.74	0.48	0.40	0.70	0.80	0.64	0.79	0.74	0.66	0.69	0.55	0.49	0.83
Program10			$	$	$	$	0.64	0.58	0.57	0.49	0.67	0.48	0.51	0.73	0.56	0.77	0.77	0.24	0.75
Program11			$	$	$	$	0.63	0.80	0.65	0.65	0.61	0.53	0.53	0.20	0.74	0.34	0.65	0.77	0.38
Program12				$	$	$	0.62	0.59	0.66	0.27	0.38	0.86	0.79	0.60	0.46	0.62	0.58	0.44	0.54
Program13				$	$	$	0.61	0.68	0.71	0.97	0.46	0.22	0.41	0.63	0.53	0.75	0.87	0.42	0.72
Program14				$	$	$	0.77	0.43	0.65	0.70	0.67	0.86	0.30	0.44	0.69	0.63	0.72	0.71	0.54
Program15					$	$	0.61	0.53	0.33	0.63	0.78	0.50	0.53	0.62	0.31	0.60	0.69	0.60	0.84
Program16					$	$	0.82	0.41	0.83	0.31	0.67	0.47	0.62	0.63	0.62	0.63	0.63	0.68	0.64
Program17						$	0.90	0.77	0.65	0.58	0.95	0.59	0.41	0.60	0.68	0.40	0.72	0.63	0.79
Program18						$	0.66	0.78	0.62	0.75	0.34	0.54	0.55	0.69	0.81	0.60	0.47	0.47	0.65

Figure 15 - At-A-Glance Example, Random Data Generator

Data Wish List:

To assess the health of the Program, five Criteria will be used: CPI, SPI, TCPI, BEI, EAC/BAC.

A definition of the Criteria is presented.

CPI: Cost Performance Index. This is a measure of past performance. This Criteria indicates if the Program is executing following its cost profile. A value of 1.0 is perfect performance, values less than 1.0 indicate overspending, and values greater that 1.0 indicate under spending.

SPI: Schedule Performance Index. This is a measure of past performance. This Criteria indicates if the Program is executing following its schedule profile. A value of 1.0 is perfect performance, values less than 1.0 indicate a behind schedule condition, and values greater that 1.0 indicate ahead of schedule.

TCPI: To-Complete Performance Index. This is a measure based on past performance to indicate how should a Program execute going forward. If a Program has executed poorly, to end on target, it would need to begin to perform better than initially planned. If a Program has executed better than initially planned, to end on target, it may underperform for the remainder of the Program. A score of 1.0 indicates the Program is right on plan and should continue right on plan.

BEI: Baseline Execution Index. This is a measure of past performance and indicates the efficiency with which a Program has been executing. A value of 1.0 indicates perfect efficiency. Values less than 1.0 indicate a less than perfect performance efficiency.

EAC/BAC: This is a measure of expected performance. If the Estimate At Complete matches the Budget At Complete then the value is 1.0. A value greater than 1.0 indicates the Program will end over budget. A value less than 1.0 indicates the Program will end under budget.

Available data:

The data for the Excel based tool is populated using information from the company database. Although the database contains a great deal of information, not all the information is applicable to the current analysis. Some fields, for example, are populated with descriptions or special codes which have no bearing on the performance or health of a Program. As such, only a small subset of information is required for meaningful analysis. The items, or Criteria, mentioned in the previous section were all readily available from the database and are included in the analysis and Program health calculations.

Pulling Data:

As mentioned previously, to be meaningful, data must be collected at standard intervals over a long period of time. The performance algorithm suggested here is organized to view and make recommendations given a full years' worth of data. The database associated with the data may, or may not, adequately retain historical data. In other words, past data may no longer be available. However, real-time data most certainly is available and could be pulled from the database at any time. The Excel based tool to collect, organize and analyze the data has been created. Because the tool is relatively new, the initial data to populate the tool is not readily accessible. To overcome this condition, the tool is initially populated with sample data which has been created using a random number generator. The tool output is captured in Figure 15.

Over time, the database will be consulted to obtain the five Criteria for each Program. The cadence should be monthly. As it is done, the resulting tool output will become more useful.

Manipulating Data:

Each month, as data is pulled from the database, a table such as that indicated in Figure 16 will be generated. Using the techniques described in the previous sections, weights will be assigned, Criteria values will be translated to applicable Scores, the scores will be normalized to represent areas of a circle, and then summed across for each Program yielding a final Program health number. The Program health numbers will be amended to the information in Figure 15 for the current month and the remaining data will shift to the right but will be retained as past data.

This analysis has been performed, however, as mentioned, it is currently primarily consisting of sample data created from a random number generator.

Program name	1	2	3	4	5
	(CPI)	(SPI)	TCPI	BEI	EAC/BAC
Program 1	0.93	0.5	0.63	1.43	0.69
Program 2	0.96	0.68	1.1	1.19	0.77
Program 3	0.81	0.77	1.31	0.75	0.87
Program 4	0.96	0.99	1.3	0.67	0.53
Program 5	0.88	0.81	0.57	1.37	1.42
Program 6	0.68	0.64	0.95	0.71	0.62
Program 7	0.5	0.88	0.81	0.53	1.38
Program 8	0.55	0.88	0.93	1.05	0.95
Program 9	0.61	0.53	1.2	0.69	0.99
Program 10	0.55	0.83	0.81	0.88	1.01
Program 11	0.8	0.66	0.5	1.07	1.39
Program 12	0.96	0.99	1.22	1.13	0.85
Program 13	0.84	0.62	1.33	0.96	0.95
Program 14	0.54	0.72	0.56	0.62	0.67
Program 15	0.69	0.99	0.71	0.93	0.89
Program 16	0.57	0.76	1.48	1.35	1.03
Program 17	0.88	0.7	0.72	0.83	1.37
Program 18	0.5	0.67	1.16	0.53	1.16

Figure 16 - Sample Program Data

Formatted Output

Observing the results in Figure 15, only one Program requires immediate attention. That Program is clearly Program5. Program5 is the only Program which, including the current month, has had two consecutive months of Bad health. Clearly something is going on there which could benefit from DM attention.

It is worth noting that the trip point between Good and Bad health was somewhat arbitrarily defined as 0.42. As it turns out, this value yields a graphic which displays an 85/15% ratio of Good to Bad. In the future, it would probably be worthwhile to arbitrarily adjust the trip point each month and record the trip value. After some amount of time and collected data, the trip point value could be automated. For example, if the trip value is always arbitrarily adjusted to approximately 85%, then the full set of data could be used to determine the 85% point and use that as the trip value providing for additional automated analysis.

Summary

Limited available resources to create a solution is commonly what prevents the solution or adds risk during the life cycle of a Program. It is a common situation to have some type of need but with limited resources (time, money, personnel, etc.) to address that need. In this chapter, a specific example of limited resources was explored. In particular, the limited time available to a Department Manager responsible for a broad variety of Programs and how best to determine where to apply focus for the most effective positive impact to the business.

This chapter attempts to formalize the process by which the DM, as a resource, can best be utilized. A solution was explored where Program data from a database can be pulled, weights were assigned, Criteria values were translated to applicable Scores, the Scores were normalized to represent areas of a circle, and then summed across for each Program yielding a final Program health number. The Program health numbers were then amended to a table of sample data for the current month and the remaining data was shifted to the right but retained as past data. The full set of results was graphically displayed and clearly indicated a need for attention on one specific Program.

Future work will focus on developing an automated algorithm for determining the trip point between Good and Bad Program health.

Conclusion

It is possible using a large amount of Program data, to analyze the data and direct the efforts of a Department Manager responsible for a broad variety of Programs toward a specific Program to apply focus for the most effective positive impact to the business.

Chapter 3 – Current State Of The Art

The current approach with industry cost models is to start with a given design and then provide a cost estimate based on that given design. It can be considered a unidirectional sequence of events lacking a feedback mechanism. Some tools do offer some rudimentary ability to visualize sensitivity of the cost parameters. With those cost tools, after a design has been selected, the system in terms of cost can then be modeled. And, after being modeled, a cost analysis can indicate something about sensitivity.

The research referenced in Chapter 3 **[2]** refers to a cost tool which is organized as a set of questions and answers. In Figure 17, the set of questions and answers appears on the left of the tool while the impact to design appears on the right of the tool interface. The questions have to do with the system mission, and based on the answers, a design is selected. For example, a mission question might be "What is the environment for the system?" The possible answers might be "1) Space, 2) Air, 3) Sea or 4) Land". In this example, with one question and one answer, the field of possible solutions has considerably narrowed. And the design of the system is significantly defined. The cost model referenced takes this approach. This, in of itself, is a novel approach. There are no commercially available cost model tools which produce cost based on operational objectives. In addition, because the approach incorporates answers to the questions, there cannot be an infinite number of possible outcomes. The answers could be organized in order of impact to the overall system.

After the system is costed, it was possible for the referenced cost model to revisit all the questions and answers and make a rudimentary calculation for overall cost if any of the variables was modified to the next selection. In other words, the original cost might have been to select "2) Air" as the expected environment. Afterwards, it is possible to select "1) Space" and "3) Sea" and determine by how much the overall cost varied. In this way, each Q&A can have a relative factor for impact to overall cost. This was a novel approach which allows a designer to reconsider each question in order of cost impact to determine if the design could stand modification. Tools organized for this operation are not currently available in industry from a COTS cost modeling tool.

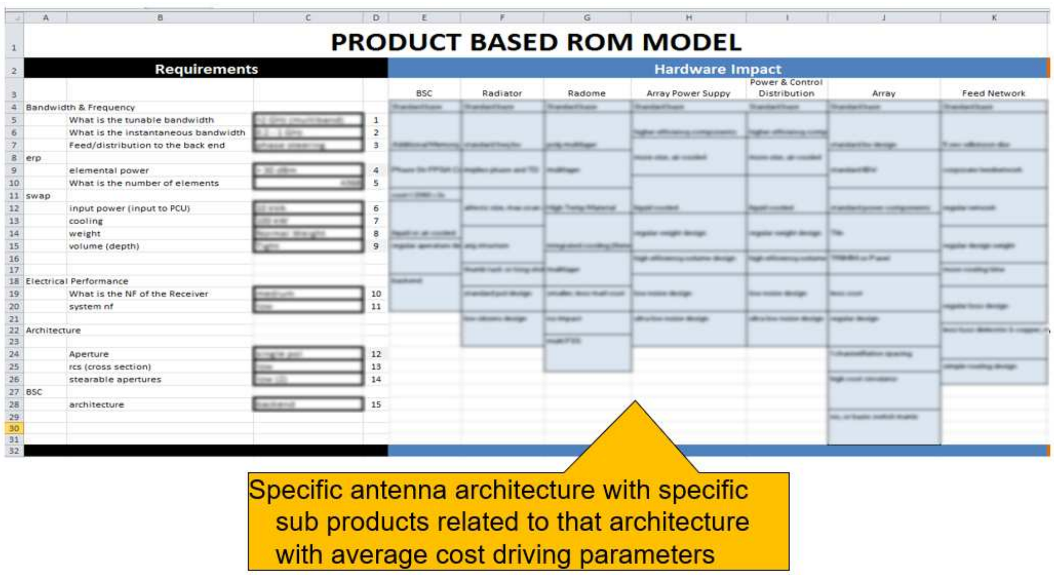


Figure 17 - Product Based ROM Model

The research referenced in chapter 7 **[6]** refers to the creation of a generic block diagram applicable to all airborne RADAR systems for military aerospace applications. Prior to the referenced research, no such generic industry standard block diagram exists which could be tailored for a specific application. Instead, many block diagrams were available for the various articles and publications within the same mission area but were customized for each application.

In the absence of a generic block diagram, the article details the creation of the generic block diagram which appears in Figure 18. To create a more universal block diagram, many examples as were necessary were pulled together until the subsequent examples failed to yield new information. Then, one version, a more general form, was pieced together which contained all the various elements. The created example was then compared to the discovered examples to verify that those examples constituted a tailored version of the more general form. The generic Block Diagram is presented here in Figure 18 for consideration and represents a novel example upon which any system designer could begin their work.

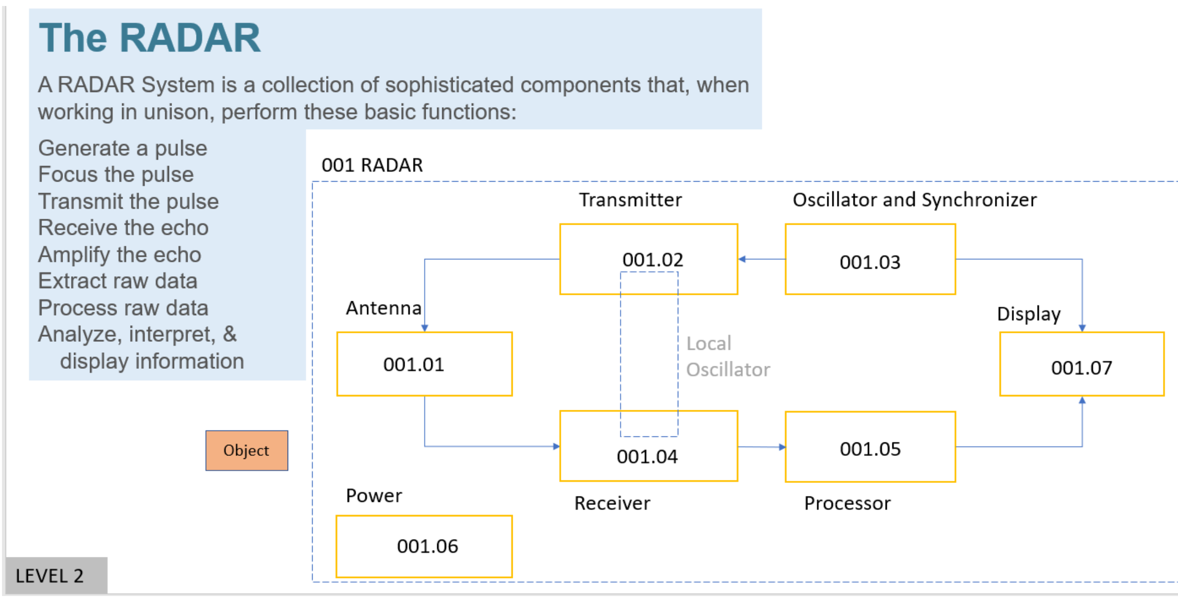


Figure 18 - RADAR Block Diagram

As was the case for the Block Diagram, the research referenced in chapter 7 documents an effort to discover a generic, industry standard WBS applicable to all airborne RADAR systems which could be tailored for any specific application. The research revealed that no such generic WBS exists. Instead, as was the case for the block diagrams, many WBSs were available for the various articles and publications but were customized for each application. Following the standard set with the generic block diagram, a standardized WBS for all airborne RADARs for a military aerospace application was created. Then, all the available examples discovered was compared to the generic WBS to verify that those examples were a sub-set of the more generalized form. It is included here for consideration, Table 1, and is a novel example upon which any system designer could begin their work.

Table 1 - RADAR WBS Structure

Level 1	Level 2	Level 3	Block Name
001			Radar
	001.01		Antenna
		001.01.01	Radiator
		001.01.02	TR Product
		001.01.03	Duplexer
	001.02		Transmitter
		001.02.01	Power Amplifier
		001.02.02	Up Converter
		001.02.03	Local Oscillator
	001.03		Synchronizer
		001.03.01	Synchronizer
	001.04		Receiver
		001.04.01	Low Noise Amplifier
		001.04.02	Down Converter
		001.04.03	Local Oscillator
		001.04.04	IF Amplifier
		001.04.05	Filters
		001.04.06	2nd Down Converter
		001.04.07	2nd Local Oscillator
		001.04.08	Detector
		001.04.09	Analog to Digital Converter
	001.05		Processor
		001.05.01	Processor
	001.06		Power
		001.06.01	Transformer
		001.06.02	Rectifier
		001.06.03	Filter
		001.06.04	Regulator
	001.07		Display
		001.07.01	Video Amplifier
		001.07.02	Display

As part of future research, it will be required to have cost models for multiple hardware levels. One of the subsystems as seen in Figure 1 may be a Frequency Locked Oscillator, or FLO. The available literature was searched for information regarding the design and development of an FLO. Although information is available, the information referenced in this document is novel in that it draws input from multiple disciplines including analog hardware, digital hardware, and software, and offers a novel process by which a system designer can create an FLO using a discriminator method. The resulting block diagram is included as Figure 19.

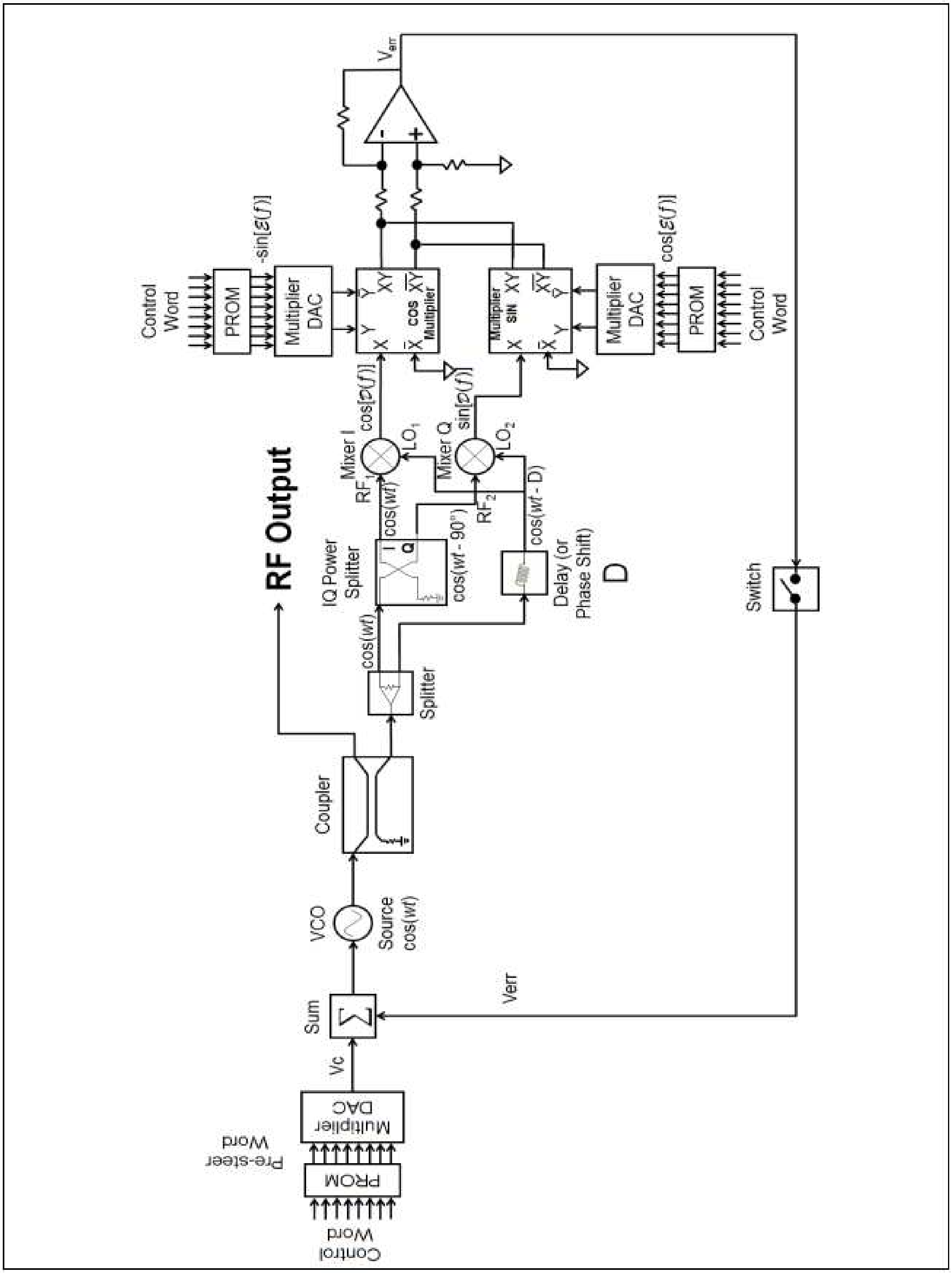


Figure 19 - FLO Block Diagram

Chapter 4 - Excel Based Cost Model: Q&A Front End with Sensitivity Calculations

The research referenced **[2]** discusses an excel based cost model which included two very novel features: A Q&A front end and a cost factor sensitivity calculation.

The research utilized two analogies to illustrate the development of the cost model, see Figure 1. First, it gave the example of hiring an electrician to mount recessed lighting in a kitchen. For such an activity, an electrician might observe the size of a kitchen, suggest four recessed cans, calculate a price per can, add an additional amount for labor and time, and provide an estimate almost immediately. This rapid ability to provide cost illustrated the ability to provide a cost estimate quickly and beyond a rough order of accuracy. This suggested that it may be possible for even a complex system to determine cost quickly and accurately based on limited information. In the case of the analogy, cost estimation does not require an expert in cost estimation. It is enough to have knowledge of the mission, provide a solution, and provide cost for that solution.

The second analogy to illustrate the development of the cost model was the idea of purchasing or pricing a car. It is apparent, there are countless types, styles, and models for a car. The price can vary anywhere from free to multimillions of dollars as in the case of a moon vehicle. To understand cost there are many variables to consider. A cost estimator would have to interface with multiple disciplines and request cost for various sub-components, providing specifications for each, and then roll all the costs up together for a total product cost. This process may take a month with a team of people. Approaching the problem from the perspective of a mission, it may be possible to determine cost quickly and accurately. For this second analogy, the mission maybe to obtain groceries. It should be apparent that a simple question and answer regarding mission objective can reduce an enormous variety of choices to a very narrow set of options. With a second question the range of choices can be limited even more. It was this concept which drove exploration into the concept of mission questions and answers to help define a design, and subsequently a cost, rapidly and within a rough order of accuracy.

To begin, a series of interviews with subject matter experts was held to understand and document a block diagram for an airborne Antenna Subsystem, see Figure 1. Because a block diagram had not yet been created, one was created utilizing various products which were combined into a general block diagram to satisfy all the known diagrams. After which, further interviews verified that all the various products could be considered tailored versions of the more general form.

Next, a series of interviews with subject matter experts was held to identify what mission questions need to be answered to influence design. The attempt was to understand a set of most significant questions which define mission parameters for an antenna subsystem. It was determined that there were 15 questions to bound the mission parameters, and for each question, there were 2, 3 or 4 possible multiple-choice answers. This was sufficient to provide enough details and boundary to define design.

The next step was to interview subsystem block subject matter experts to discuss how the various multiple-choice answers would influence the design of the respective subsystem blocks. At the conclusion of this round of interviews, it was demonstrated that it was possible to go from a desired mission to a subsystem block design which would satisfy the mission.

The next step was to collect cost data for several antenna systems. Collecting data is commonly an easy step. The difficulty comes in interpreting the cost data. Two sets of cost data for similar products may have vastly different numbers which implies there are significant differences in the product. It is the understanding of those differences which makes the cost data meaningful.

To make the cost data meaningful, it becomes necessary to create scaling factors, or Key Size Metrics. These KSMs essentially normalize the data. As a simplistic example, collected data may show that one house costs $2M and a second house costs $100K. They are both houses, but with vastly different sets of cost data. To understand the differences, sizing factors must be created. One house might be 20,000 square feet while the other is 1,000 square feet. If the cost was divided by the square feet, a sizing factor can be calculated as 100 $/square foot. When the two sets of data are normalized for that KSM, the price per house is the same. This step is really the most difficult and important step in using data to create a cost model.

A cost model, as described with a front end consisting of mission applicable questions and answers, is not commercially available and represents a novel advancement towards cost analysis, see Figure 17.

For Example, a mission question might be "What is the environment for the system?" The possible answers might be "1) Space, 2) Air, 3) Sea or 4) Land". In this example, with one question and one answer, the field of possible solutions has considerably narrowed. And the design of the system is significantly defined.

Because the range of answers for each question was not infinitely large, it was in fact a very limited set of possible choices, it was possible to allow Excel to calculate the effect on overall cost by changing each one of those answers to the adjacent answer. For example, for the question of system environment, the possible answers might be "1) Space, 2) Air, 3) Sea or 4) Land". The answers are arranged in order of cost impact. If, for example, the environment was selected to be "2) Air", the effect on overall system cost to modify the answer to either "1) Space" or "3) Sea" can be calculated. Furthermore, the relative effects on all the possible options are compared. This provides the system designer with trades which demonstrate the most significant impact cost drivers.

A cost model, as described with a relative sensitivity indicator is not commercially available and represents a novel advancement towards cost analysis.

Chapter 5 - Excel Based Cost Model: Auto-Generation Of Basis Of Estimate

The research referenced **[3]** discusses an excel based cost model which included a very novel feature: Auto-generation of a Basis Of Estimate.

The available COTS cost model packages have capabilities which can provide cost for a given design. Several companies and software packages can do this. But there is a limitation when it comes to creating a Basis Of Estimate, or BOE. The BOE is typically a document which describes how the cost was calculated. A cost model is typically made up of a database of applicable cost examples. All those data sets are used to create a cost for a specific new case. The problem becomes, to support the new cost bid, a cost analyst must provide the entire data base of costs which support the bid, which is prohibitive. There becomes a significant challenge to say "THIS new bid is similar to THAT bid and scaled due to complexity" since the cost model is relying on many examples to create a point solution. Furthermore, for a typical bid, the similar-to program is often determined through tribal knowledge rather than a rigorous method of documentation and comparison. The cost model referenced in this chapter offered a solution to those problems.

The problem posed to the research reference in this chapter focuses on the issue of how to justify a bid with a Basis Of Estimate, or BOE, without divulging the entire data base of cost data. Most cost models use a significant set of data for a variety of programs. For each program, the data set is analyzed for Key Size Metrics, KSMs, to normalize the data and make the data useful. With a significant set of data, the results could be plotted, and an equation could be determined which described the cost vs. the KSM, see Figure 20.

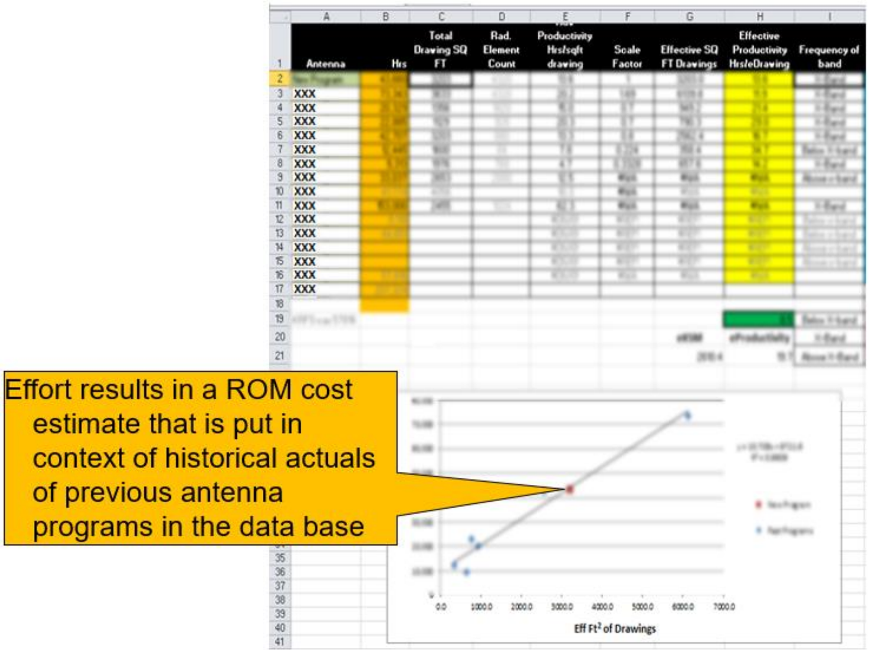


Figure 20 - Cost Model Output

Using the KSM, a new cost could be calculated for the new specific case in question which would fall on the line which describes the behavior of the KSM. For a simple example of the cost of a house, data for many houses would be collected and when adjusting for square footage, it was determined that a house would cost \$100/square foot ± \$20. This is of course a linear relationship. A graph could be created which shows various house sizes and the corresponding cost for each house. Then, for a specific example, 5,000 square feet, the house cost can be calculated using the linear equation. With the same example, 5,000 square feet x \$100/square feet = \$500,000 house. And provided the new value corresponding to 5,000 square feet falls significantly within the range of the collected data set, there is confidence in the expected cost calculation.

The problem arises when the calculated cost requires justification to a potential customer, such as Company A. To present that the cost was calculated using an equation, the complete data set would be required to justify the equation used for the calculation. In other words, the data set uses historical actuals from multiple companies such as Company A, Company B, Company C, etc., all of which could potentially be competitors. It is impossible to make all that data available to one customer, Company A, as part of a bid. This then becomes a fundamental problem with the cost model in that it can only be used as a test of reasonableness for which the auditing threshold is significantly less. In other words, calculate the cost by some other means which can be justified on its own, then compare it to this method to verify the calculation is reasonable. Take for consideration the chart in Figure 21.

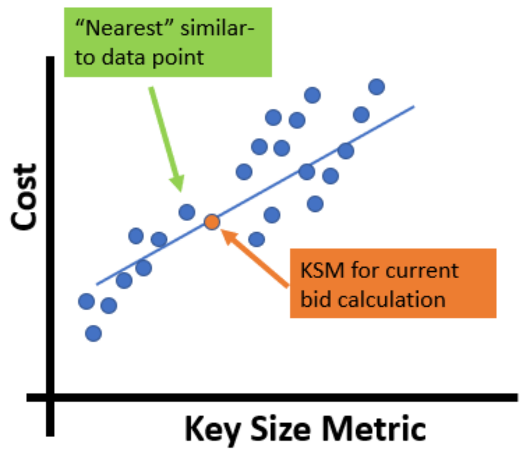


Figure 21 - KSM vs. Cost

In this figure, the blue dots represent data from many different programs which have been normalized by a scaling factor and plotted against the KSM. The blue line represents a linear curve fit for the available data. That curve fit can be described by an equation. The orange dot represents a KSM for which a cost must be calculated. In other words, this is the product for which to calculate cost. To calculate cost for the orange dot, the KSM is chosen, and the linear expression (the blue line) yields the corresponding cost. As indicated, every data set corresponding to a blue dot would have to be supplied to justify the linear cost equation. Practically speaking, it is impossible to supply all that data, which renders the linear cost equation useless except as in the case of a test of reasonableness when it would not be required to supply all the data.

It can be observed that there exists a blue dot which is very close in proximity to the orange dot. The "close" blue dot represents a cost for a product which is similar-to the product for which cost is required to be determined. In fact, it is the most similar-to product. As a cost analyst, to determine cost for a product, it would be most desirable to have a similar-to product which has many similarities in terms of scope and requirements as compared to the product for which cost is desired, that no scaling or adjustments need to be made. Ideally, it is desirable to claim that the last iteration of the program required X dollars and it is expected to cost X dollars for a subsequent iteration of the program. In a case such as that, it is very easy to justify the system cost calculation. In terms of an audit, one set of applicable data can be provided and claim that that program represents the BOE. Then the graph becomes a test of reasonableness. In this way, every shortcoming of the cost model has been eliminated. Commonly, the similar-to programs are not adequately documented or archived in such a way as to be useful for a future bid. Instead, information for a particular bid would be referenced to a similar-to program by tribal knowledge. In other words, the cost analyst, having performed their job for many years, would recall a similar program and have that data handy. That similar program may in fact NOT be the most similar program, but it is the program of highest familiarity.

The referenced research provides a novel contribution in that it demonstrates how a cost analyst, without any prior knowledge of legacy programs, can determine which specific legacy program has the most significantly similar data set as compared with the program under consideration.

By using the "close" blue dot, see Figure 21, a BOE can be written, see Figure 22. That BOE provides only one Program as a BOE. The process by which the BOE is written is mechanical but straight forward. To streamline the process, the activity can be automated.

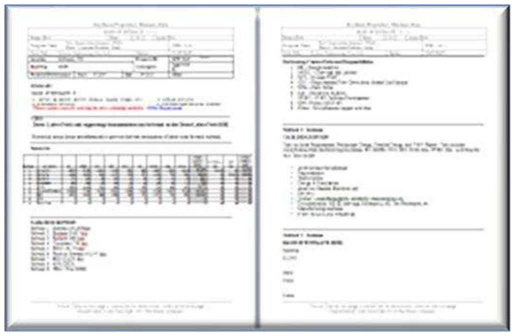


Figure 22 - BOE Template.

A BOE template document in Word, containing all the boiler plate information which would ordinarily be contained in a BOE, was created. The boiler plate information would include but not be limited to header, various sections, signature blocks, etc. A series of dynamic links to tie the Excel cost model to the Word BOE document was used, see Figure 23. As the Excel file is updated, it automatically identifies the "close" blue dot and instantly updates the Word BOE. In addition, the ability for the user to manually drive the dynamic links to select a dot other than the "close" blue dot was added. This is done because in the case of a complex system, one subsystem element may be similar in cost to Program A while a different subsystem element may be similar in cost to Program B. For a bidding exercise, it is always better to reference the same similar-to program for as many subsystems blocks as possible. For a particular subsystem element, the referenced program might not be the ideal choice, it is never-the-less preferable such that it remains in family with the other subsystem element estimates.

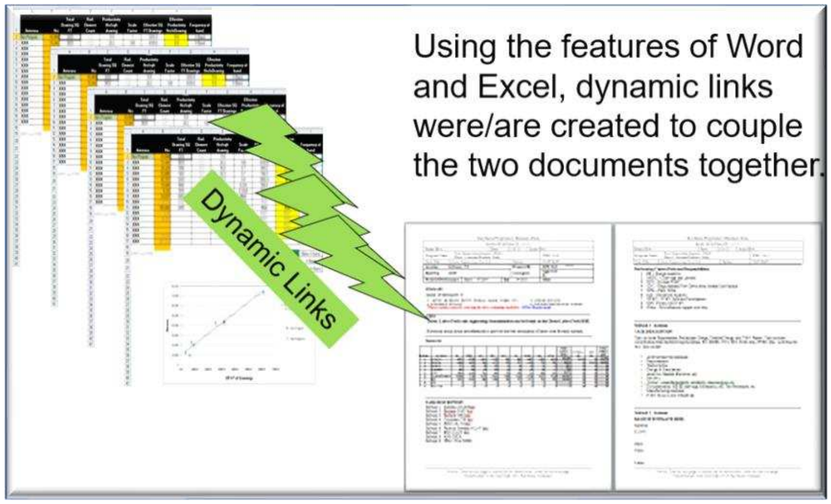


Figure 23 - BOE with dynamic links to Excel

A cost model, as described with an auto-generation feature for a Basis Of Estimate document is not commercially available and represents a novel advancement towards cost analysis.

Chapter 6 - Frequency Locked Oscillator as a Subsystem Component

The research referenced in this chapter discusses a Frequency Locked Oscillator (FLO) utilizing a discriminator method which represents a subsystem block which maintains the same cost sensitivity properties as the system. The research referenced in this chapter is applicable to the dissertation research in that it forms a subsystem element within the RADAR system which maintains all the same cost considerations in terms of modeling and cost sensitivity at a subsystem element level. The article had a novel feature in which in describes a comprehensive approach to design for the product including contributions from Digital, Analog and Software Engineering.

Countless applications exist which require a stable high frequency signal source. Unfortunately, oscillator designs, as they go up in frequency, become inherently noisy. That is a problem for a designer. A frequency lock loop is a solution which has been around for many years. Design techniques and analysis have been developed and are well understood. The typical tradeoff, and the need for such a solution, is between a high frequency source that is unstable vs. a low frequency source that is very stable. And, by locking the two signal sources together, the best of both worlds can be achieved. Most commonly design solutions employ exclusively analog elements. An interesting variation to the solution of this problem has been with the use of a delay line. Essentially the circuit uses the one frequency source, delayed in time, as its own frequency reference. That solution eliminates the need for a second stable low frequency source. Furthermore, an uncommon variation to the solution includes replacing analog elements with digital elements. This chapter outlines how to build up a solution to the problem utilizing a delay line approach which employs both analog and digital elements as well as a brief discussion on the software required to control and linearize the final circuit. This chapter is organized much like a tutorial where each element is introduced, explained, and then added to the solution. In the end, the reader should possess the basic skills and theory necessary to be able to design a solution to the problem using a delay line, analog elements, and digital elements as well as an understanding of the software requirements to implement the completed solution.

Introduction

There are many approaches to frequency, or phase lock an oscillator. The circuit in Figure 24 captures the basic elements required for a locking solution.

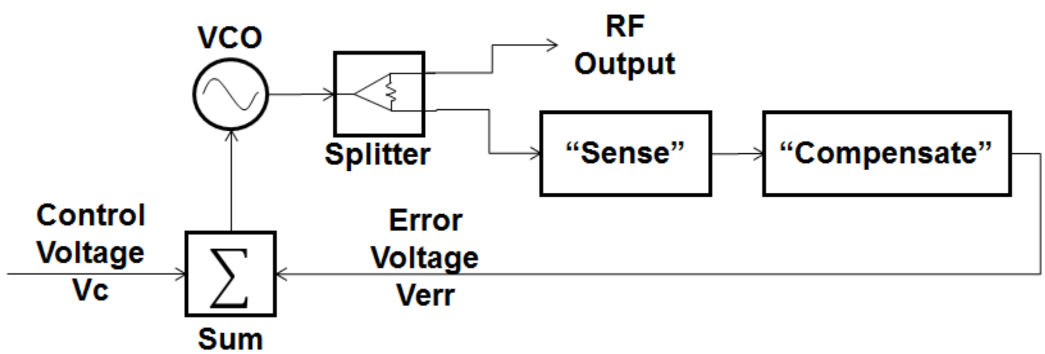


Figure 24 - Simplistic Frequency Locking Circuit.

As can be seen, the VCO is driven by a Control Voltage, Vc. The output of the VCO is routed to an element which can "Sense" a frequency variation, or drift. An element must also be included which can "Compensate" for that drift. The output of the "Compensate" element, or Error Voltage, is then summed with the Control Voltage and fed back into the VCO to compensate for the frequency drift. The circuit in Figure 24 represents a generic solution. Although there are many architectures to accomplish these functions, most every solution will involve these elements.

Simple Mixer Basics

The circuit in Figure 24 includes a "Sense" element. Typically, that element will incorporate a mixer. An understanding of mixer basics is required. A mixer is a device used to multiply two signals. The circuit in Figure 25 can be analyzed.

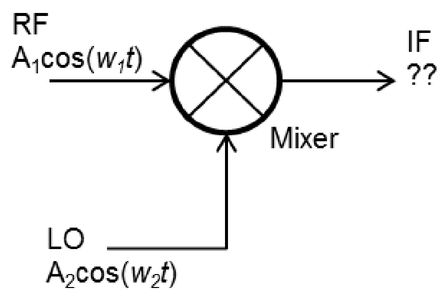


Figure 25 - Basic Mixer Circuit.

From the standard trigonometry tables, it is known:

$$\cos\alpha * \cos\beta = \tfrac{1}{2}\cos(\alpha - \beta) + \tfrac{1}{2}\cos(\alpha + \beta) \qquad (1)$$

Therefore, the signal at IF can be calculated:

$$\mathrm{IF} = \mathrm{RF} * \mathrm{LO}$$

$$= A_1\cos(w_1t) * A_2\cos(w_2t)$$

$$= \frac{A_1A_2}{\mathbf{2}}\,[\underbrace{\cos(w_1t - w_2t)}_{\text{Low f's}} + \underbrace{\cos(w_1t + w_2t)}_{\text{High f's}}]$$

Very commonly a mixer is used to down covert an RF signal to a lower, intermediate frequency. And practically, any high frequency contributions will be filtered out. If that is the case, then the equation can be simplified and becomes as follows:

$$IF = \frac{A1A2}{2}\,[\cos(w1 - w2)t] \qquad (2)$$

Adding A Delay Line

Building upon the basic mixer, the "Sense" element from Figure 24 is continued. The circuit in Figure 26 adds a Delay element to the mixer circuit from Figure 25. The circuit is excited with the ideal source cos(*wt*).

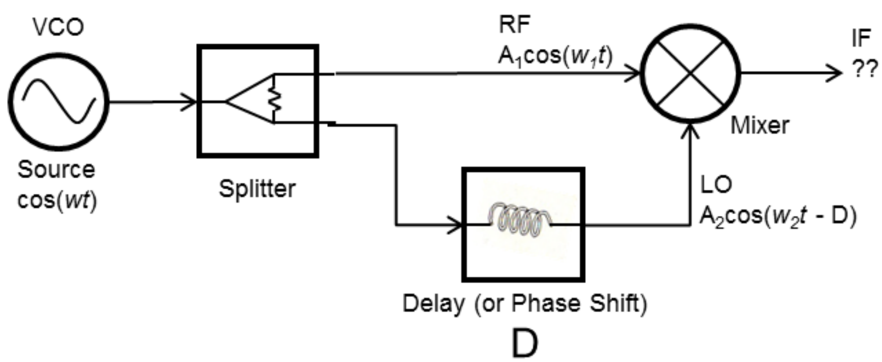


Figure 26 - Adding a Delay Line.

The output at IF can be calculated:

$$IF = RF * LO$$

$$= A_1 \cos(w_1 t) * A_2 \cos(w_2 t - D)$$

$$= \frac{A_1A_2}{2}\,[\cos(w_1 t - w_2 t + D) + \cos(w_1 t + w_2 t - D)]$$

cancels adds

Low f's High f's

It can be observed that the two frequencies, w_1 & w_2, are the same. The result can be simplified using:

$$w_1 = w_2$$

The resulting equation becomes:

$$IF = \frac{A_1A_2}{2}[\underbrace{\cos(D)}_{\text{Low f's}} + \underbrace{\cos(2w_1t - D)}_{\text{High f's}}]$$

Once again, the eventual completed design will incorporate a low pass filter, so the high frequency contributions will be filtered out. As such, the equation can further be simplified as follows:

$$IF = \frac{A1A2}{2}\cos(D) \quad (3)$$

Equation (3) is significant and bears consideration. To begin with, it should be recognized that the component "Delay" has not defined. For the proposed design, the delay element will be a "long" length of semi-rigid cable.

The delay element can be portrayed as in Figure 27. Here it can be observed that a signal is incident upon the delay line.

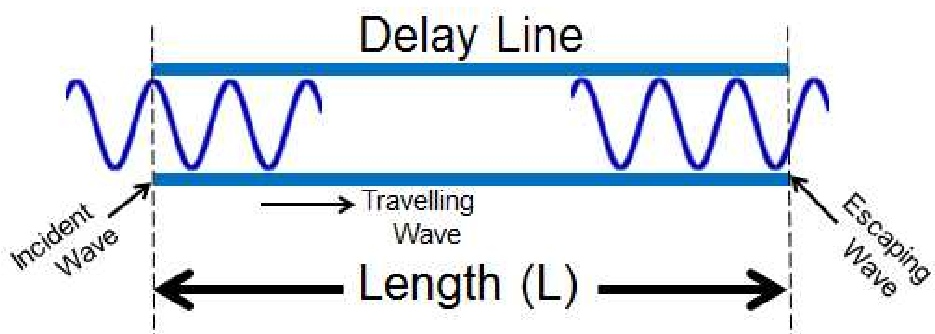


Figure 27 - A wave in a Delay Line.

In this illustration (Figure 27), the incident signal is a cosine. The wave travels down the line and eventually escapes from the line. The wave that escapes from the other end of the delay line will have some phase. The escaping wave could be, in fact, any phase and is clearly dependent on length L. Therefore, delay D is a function of length L and can be express:

$$D(L)$$

In addition, consider two cases with two very different frequencies as in Figure 28:

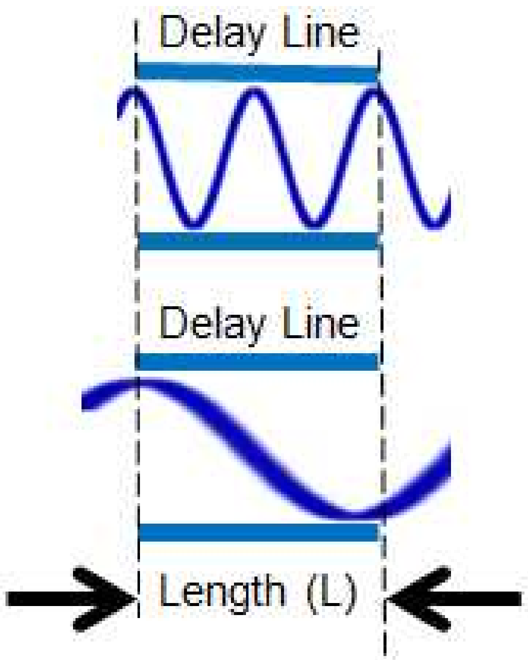


Figure 28 - Two different frequencies in a length of Delay Line.

In Figure 28, it has been represented pictorially that, for the same length of delay line, with two distinct frequencies, the phase shift is considerably different. In other words, the delay line will cause a phase shift and it is a function of frequency. So, D is a function of frequency and can be expressed:

$$\mathcal{D}(f)$$

Equation (3) can be rewritten:

$$\text{IF} = \frac{\text{A1A2}}{2}\cos[\mathcal{D}(f)] \qquad (4)$$

In Equation (3) it can be observed that the frequency contributions of the incident source signal were cancelled out and depending on the amount of delay the result will be some D.C. value, a constant.

However, in Equation (4) it can be observed that as the incident source frequency changes, the D.C. value at the output IF changes. This is the desired result and serves as the basis, or heart, of the eventual frequency locked loop. As the source frequency changes slightly, this circuit will produce a changing D.C. voltage. The next steps will develop how that changing D.C. voltage can be fed back to the source to compensate for a frequency drift.

The remaining unknowns:

The circuit in Figure 26 contains the basic elements to “Sense” the error associated with a frequency locked loop (refer to Figure 24). However, it remains insufficient for a practical application. There are too many variables and unknowns. To understand the limitations, it is necessary to explore Equation (4) further, repeated here.

$$\text{IF} = \frac{\text{A1A2}}{2}\cos[\mathcal{D}(f)] \qquad (4)$$

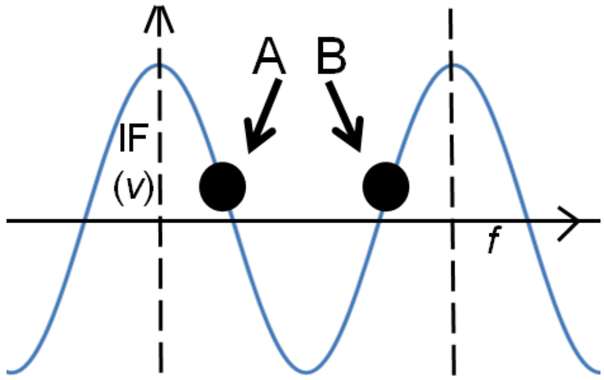


Figure 29 - Two points with the same voltage.

Figure 29 illustrates the result from Equation (4) as frequency *f* is swept. The illustration indicates that the voltage at IF will vary sinusoidally as the frequency *f* is swept. At any arbitrary source frequency, Equation (4) indicates there will be a voltage. As seen in Figure 29, the voltage will be some arbitrary point on the curve. Of course, the voltage can be measured. However, for any given voltage, it is still unknown at which point along the curve the voltage represents, as illustrated with points A or B. These two points correspond to the same voltage.

In addition, as the frequency begins to drift, the point moves along the curve. Because it is unknown in which way the frequency is drifting (up or down), it remains unknown which direction along the curve the point is moving.

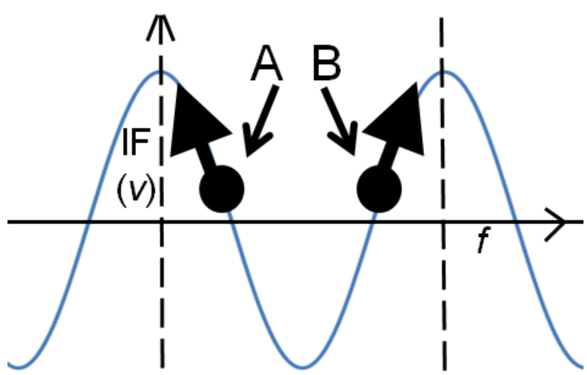


Figure 30 - Two moving points with the same voltage.

In other words, if the voltage began to increase, it is equally likely to be at point A moving to the left or at point B moving to the right, see Figure 30. In both cases the voltage would increase. Although the circuit represented in Figure 26 has the heart of a frequency locked loop, there are simply too many variables to use this circuit as it is.

Adding An IQ Power Splitter

Building upon the circuit in Figure 26, the "Sense" element (refer to Figure 24) is continued. The circuit in Figure 26 combined a Delay element with the basic mixer circuit. The circuit in Figure 31 has a few new elements: an IQ Power Splitter and a second mixer. The excitation source and other elements remain the same.

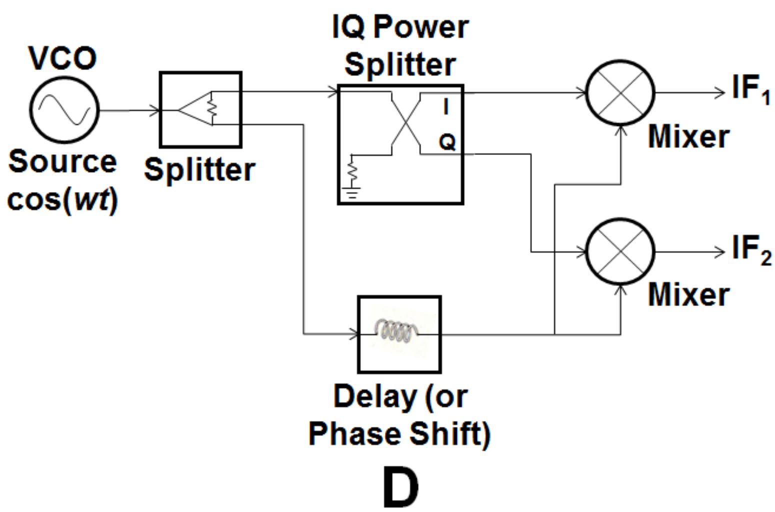


Figure 31 - Adding an IQ Power Splitter.

A brief discussion on IQ power splitters is helpful to understand why these additional elements have been added.

IQ Power Splitter Discussion:

The IQ Power Splitter gets its name in the following way:

I = in phase (cos)

$$Q \quad = \text{quadrature} \quad (\sin)$$

The term Quadrature comes from quad, or four, or four slices of a period. In other words:

$$\tfrac{1}{4}\lambda \text{ , where } \lambda \text{ is a full wavelength}$$

It is further known that:

$$\tfrac{1}{4}\lambda = \frac{360^\circ}{4} = 90^\circ$$

To put it simply, an IQ power splitter takes a signal and splits it into two signals, one signal in phase with the source and a second signal 90° out of phase with the source. And, for short, IQ.

The circuit in Figure 31 has been redrawn in Figure 32 and includes some useful equations.

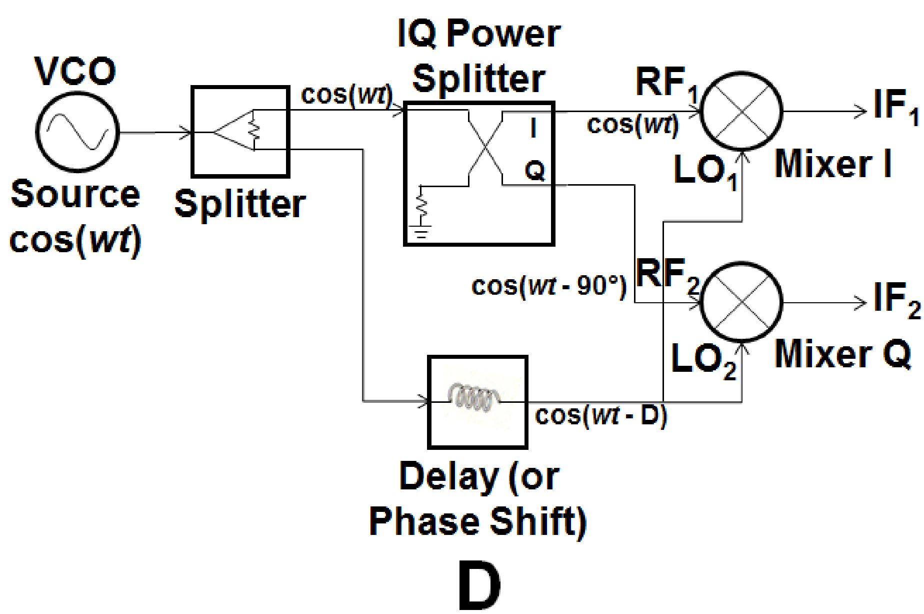


Figure 32 - An IQ Power Splitter with some equations.

Calculating the output at IF_1:

Observing the circuit in Figure 32, the following equation can be written:

$$IF_1 = RF_1 * LO_1$$

$$= \cos(wt) * \cos(wt - D)$$

If the amplitudes (or constants) are appropriately selected, the result can be simplified. Remaining consistent with the earlier constants, if the following selection is made:

$$\frac{A1A2}{2} = 1$$

Then Equation (4) can be simplified as follows:

$$IF1 = \cos[\mathcal{D}(f)] \tag{5}$$

Calculating the output at IF_2:

Again, observing the circuit in Figure 32, the following equations can be written:

$$IF_2 = RF_2 * LO_2$$

$$= \cos(wt - 90°) * \cos(wt - D)$$

Amplitudes were selected appropriately such that they become equal to "1". If selected appropriately, the following equations can be derived:

$$IF_2 = \cos(wt - 90° - wt + D) + \cos(wt - 90° + wt + D)$$

$$IF_2 = \underbrace{\cos(D - 90°)}_{\text{Low f's}} + \underbrace{\cos(2wt - 90° + D)}_{\text{High f's}}$$

And like before, the high frequency contributions will be filtered out. The equation can be simplified as follows:

$$IF_2 = \cos(D - 90°)$$

From the standard math tables, it is known:

$$\cos(\alpha - 90°) = \sin(\alpha) \quad (6)$$

The result then becomes:

$$IF_2 = \sin[\mathcal{D}(f)] \quad (7)$$

Understanding the importance of IF_1& IF_2:

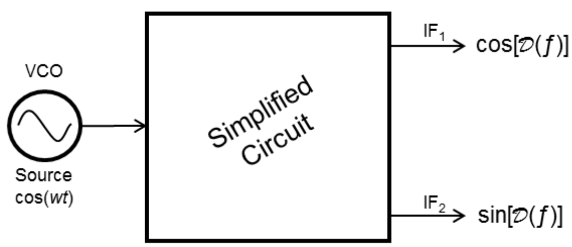


Figure 33 - Simplified representation of Figure 32

With the simplified representation of the circuit in Figure 32, it is observed in Figure 33 that if a frequency source is applied, the outputs are two voltages. The voltages are a function of the delay line, which is a function of the source frequency. As the source frequency begins to drift, a change in voltage will appear at outputs IF_1 & IF_2.

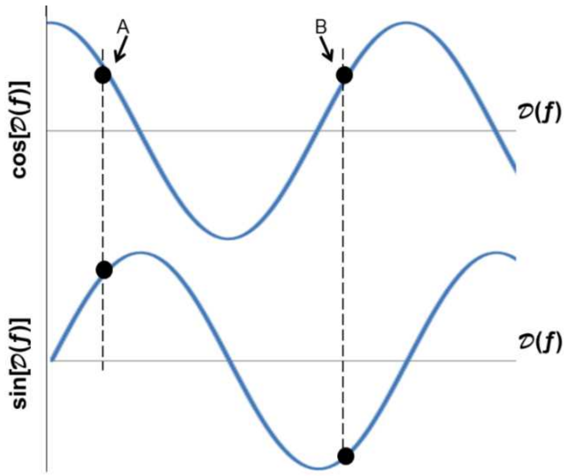


Figure 34 - Output voltages of Figure 33

The significance of the quadrature outputs is that now all the variables and unknowns have been eliminated. Consider points A & B on the cosine curve in Figure 33. As before, although the voltage can be measured, it is still indistinguishable via the voltage (in the cosine curve) if the circuit is producing the voltage at point A or B. And as before, if the source frequency were to begin to change, or drift, the voltage value will change. But since the direction of frequency shift is unknown, the direction along the curve is still unknown. However, if the voltage along the sin curve were to be simultaneously monitored, all unknowns are eliminated. Consider, as in this illustration of Figure 34, if cosine decreases in voltage and sin decreases in voltage, then the circuit could only be performing at voltage B and moving to the left.

As was demonstrated, with the unknowns eliminated, there is now a means to sense and quantify the frequency drift of the source oscillator. In other words, the "Sense" element as depicted in Figure 24 is now understood.

The next step in the development of this solution is the feedback circuit to correct for the frequency drift in the source oscillator.

A Compensation Circuit - Basics

As can be seen in Figure 24, a "Compensate" element must be included. An understanding of the simplified circuit illustrated in Figure 35 is required before proceeding to a more detailed design approach.

The purpose of this circuit is to translate the frequency drift information into a useful error voltage. For this illustration, the output voltage Verr can be calculated.

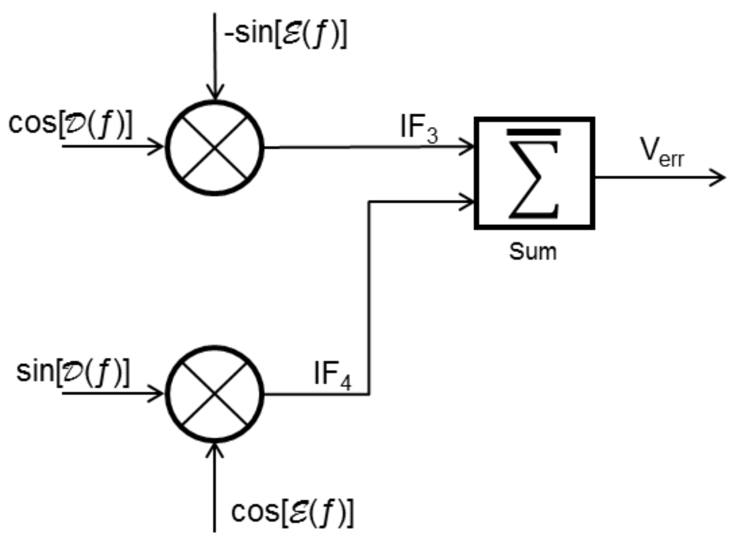


Figure 35 - Compensation Circuit - Basics

The inputs cos[$\mathcal{D}(f)$] and sin[$\mathcal{D}(f)$] should be familiar by now. The function $\mathcal{E}(f)$ has been introduced. The function $\mathcal{E}(f)$ needs to be defined and will be defined in the following way:

$\mathcal{E}$ is a function of frequency and is arbitrarily defined such that:

$$\sin[\mathcal{E}(f) - \mathcal{D}(f)] = 0, \text{ for every frequency } f \quad (8)$$

Calculating the intermediate output at IF_3***:***

In Figure 35, the signal at node IF_3 can be calculated as follows:

$$IF_3 = \cos[\mathcal{D}(f)] * -\sin[\mathcal{E}(f)]$$

$$IF_3 = -\sin[\mathcal{E}(f)]\cos[\mathcal{D}(f)] \qquad (9)$$

Calculating the intermediate output at IF_4***:***

In Figure 35, the signal at node IF_4 can be calculated as follows:

$$IF_4 = \sin[\mathcal{D}(f)]\cos[\mathcal{E}(f)] \qquad (10)$$

Calculating the output at Verr:

In Figure 35, the signal at the output Verr can be calculated as follows:

$$Verr = -(IF_3 + IF_4)$$

$$Verr = -(IF_4 + IF_3)$$

$$Verr = -(\sin[\mathcal{D}(f)]\cos[\mathcal{E}(f)] - \sin[\mathcal{E}(f)]\cos[\mathcal{D}(f)])$$

From the standard trigonometry tables, it is known:

$$\sin(\alpha - \beta) = \sin(\alpha)\sin(\beta) - \cos(\alpha)\cos(\beta) \qquad (11)$$

Therefore, the equation can be simplified as follows:

$$Verr = -\sin[[\mathcal{D}(f)] - [\mathcal{E}(f)]] \qquad \text{or}$$

$$Verr = \sin[[\mathcal{E}(f)] - [\mathcal{D}(f)]] \qquad (12)$$

The result in Equation (12) can be compared with the result derived in Equation (8). This implies that Verr is equal to zero for all frequencies f. This is, in fact, the intended result. It has been thus derived that Verr is conditionally equal to zero for all frequencies f if Equation (8) is true. And we control $\mathcal{E}(f)$. The function $\mathcal{E}(f)$ will be created so that Equation (8) is true and, therefore, Verr = 0.

Understanding the meaning of Verr = 0:

Observing the simplistic circuit indicated in Figure 24, a VCO, or Voltage Controlled Oscillator, can be seen. To tune the VCO, the voltage Vc, or control voltage, is applied. As Vc changes, the VCO is swept through its range of frequencies. The resulting Verr, or the error voltage, is added to Vc, or the control voltage. In an ideal situation, Vc is used to tune the VCO, and the VCO is stable. This, of course, will never be the case. It is expected that the VCO will drift. When it begins to drift, the goal is to sum in an error correcting voltage with Vc to pull the VCO back to its desired frequency. And once the VCO has returned to is desired frequency, Verr should return to zero. That is the goal of this design. And at this point it all hinges on the imaginary theoretical function $\mathcal{E}(f)$. What remains is to realize this function.

A Compensation Circuit - Advanced

The "Compensate" element from Figure 24 is further developed. The circuit in Figure 35 utilizes two mixers. As discussed earlier, a mixer is used to multiply two signals. The discussion to this point has been strictly analog electronics. This approach has been used successfully for many years. However, in this section, some of these analog components will be replaced with digital components in a more novel approach to an old problem

The Multiplier:

The simplistic circuit in Figure 35 was illustrated solely with analog components. There exist available digital counterparts as illustrated in Figure 36.

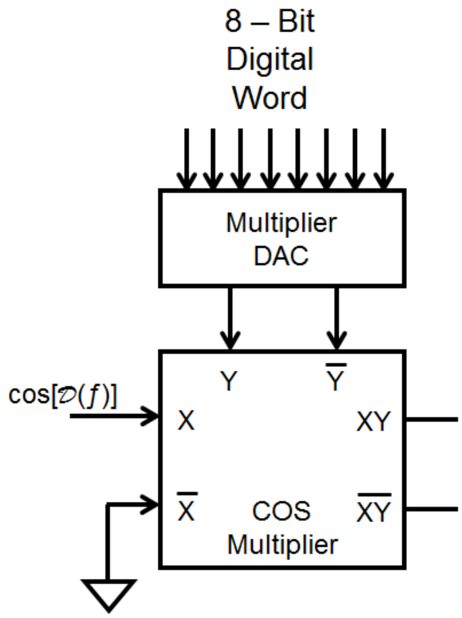


Figure 36 - Digital Multiplier IC and DAC.

Instead of using an analog mixer as in Figure 35, Multiplier IC and a DAC was explored. The two signals to be multiplied are applied to ports X & Y. At port X, the familiar signal cos[$\mathcal{D}(f)$] appears. The signal at port Y is supplied from a DAC, or a Digital to Analog Converter.

Brief DAC Discussion:

A DAC is a device that takes a digital word and converts it to an analog voltage. In this case, an 8-bit DAC was used. With 8-bits, the quantity of digital words is:

$$2^8 = 256$$

Where the range of digital words is:

$$\text{Digital Words} = 0 \text{ to } 255$$

A sample of the DAC output voltages is captured in Figure 37.

Multiplier DAC Word	Voltage At Y	Voltage At $\overline{Y}$
0	0V	-1.0V
127	-0.5V	-0.5V
255	-1.0V	0V

Condition for max voltage (at $\overline{Y}$ = -1.0V)

Condition for min voltage (at $\overline{Y}$ = 0V)

Figure 37 - DAC input words and output voltages.

Controlling the DAC – Basics:

If it is desired to see a sinusoidal signal at Y, then it is necessary to increment the 8-Bit multiplier DAC digital word between 0 and 255 at a sinusoidal rate. Please consider the following four figures:

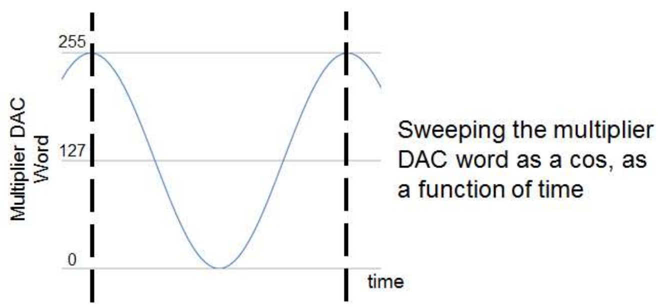


Figure 38 - Sweeping DAC vs. Time

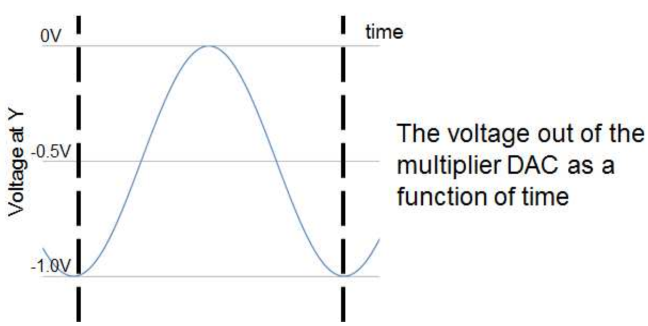


Figure 39 - DAC Voltage vs. Time

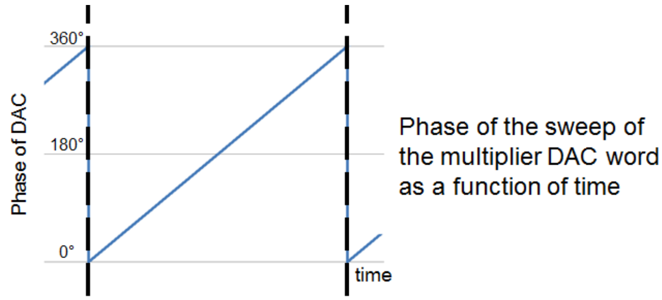


Figure 40 - DAC Phase vs. Time

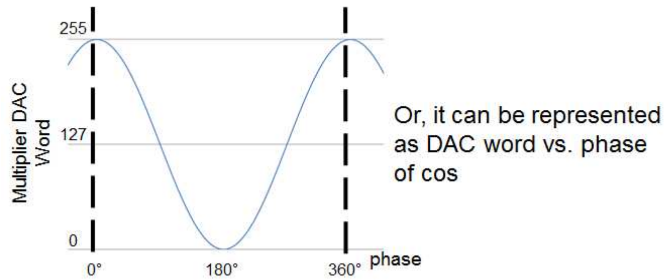


Figure 41 - DAC Word vs. Time

A Compensation Circuit Realized:

The simplistic analog circuit in Figure 35 can be realized with a hybrid approach as illustrated in Figure 42.

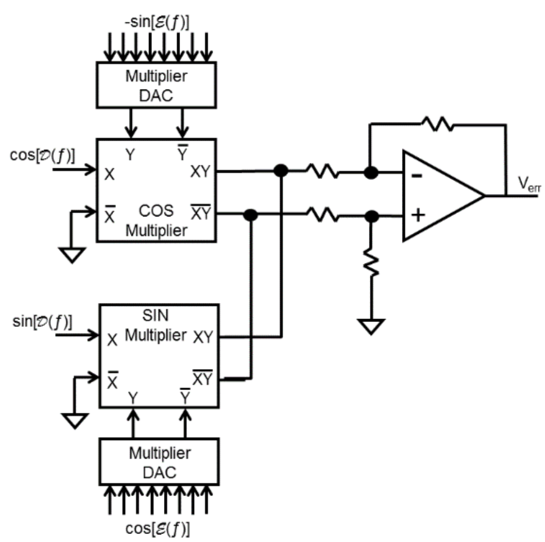


Figure 42 - Compensation circuit realized.

It has already been demonstrated from Figure 35 and Equation (12) what to expect for Verr. The result for Figure 42 is the same and is simply repeated here:

$$\text{Verr} = \sin[[\mathcal{E}(f)] - [\mathcal{D}(f)]] \tag{13}$$

Controlling the DACs – A Detailed Study:

The pair of curves depicted in Figure 43 represent the curves for the cos[$\mathcal{E}(f)$] & - sin[$\mathcal{E}(f)$] functions from Figure 42.

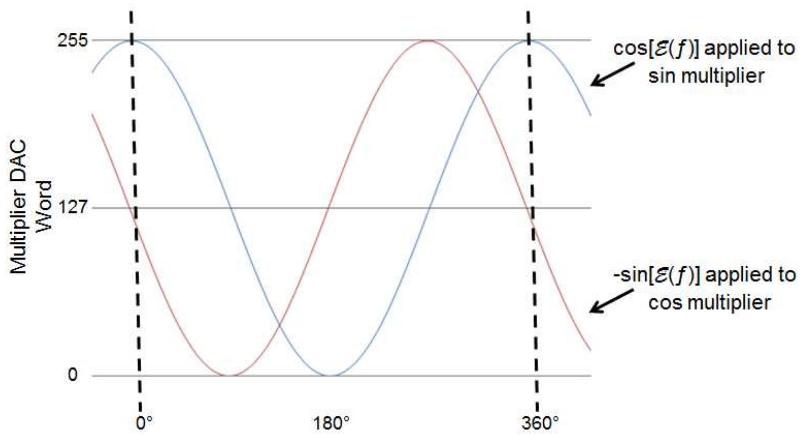


Figure 43 - Multiplier DAC word vs Phase.

If a vertical line were drawn through the curves in Figure 43, it would intersect the curves at a pair of points. For example, Table 2 captures a few obvious pairs of points:

Table 2 - A few obvious DAC word pairs.

Phase	cos[$\mathcal{E}(f)$] Multiplier Word	- sin[$\mathcal{E}(f)$] Multiplier Word
0	255	127
90	127	0
180	0	127
270	127	255

4 – pairs of points

In the example of Figure 42, there are two 8-bits DACs. The following questions must be answered: How many pairs of points are possible? What are those pairs? And at what phase do those pairs exist?

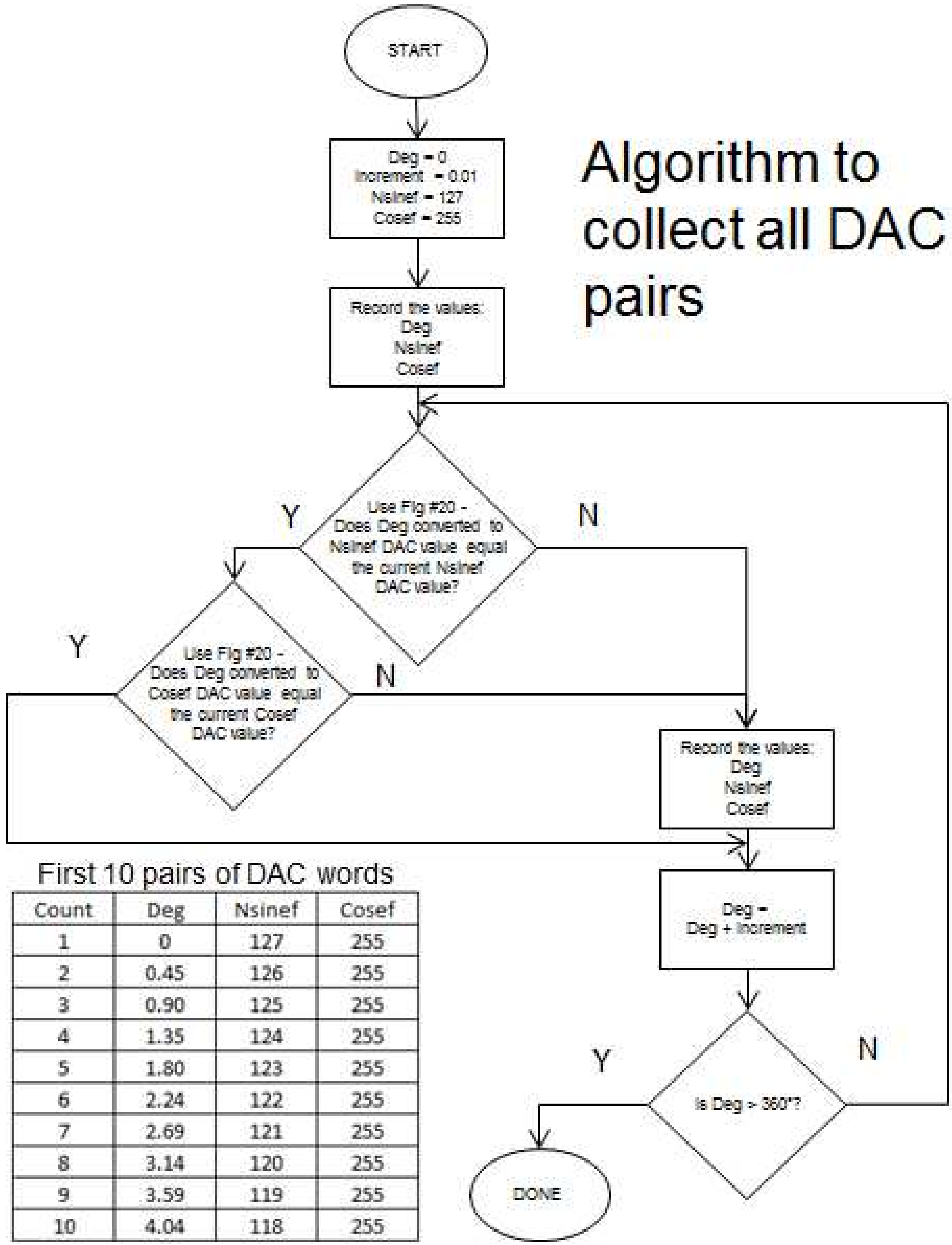
A simple BASIC program to generate all the pairs of DAC control words was written. The flow diagram in Figure 44 should be useful to a reader with some programming skills. The specific code has been left to the reader as an exercise. The answer, however, is 1,020 combinations.

As was demonstrated, there is now a means to sense and quantify the frequency drift of the source oscillator with a means to translate the drift into a voltage to compensate for the drift. In other words, the "Sense" and "Compensate" elements as depicted in Figure 24 is now understood.

The next step in the development of this solution is to put the pieces together along with some more details to complete the design.

All The Pieces Together

The final design is captured in Figure 45.



First 10 pairs of DAC words

Count	Deg	Nsinef	Cosef
1	0	127	255
2	0.45	126	255
3	0.90	125	255
4	1.35	124	255
5	1.80	123	255
6	2.24	122	255
7	2.69	121	255
8	3.14	120	255
9	3.59	119	255
10	4.04	118	255

Figure 44 - Algorithm to collect all DAC word pairs.

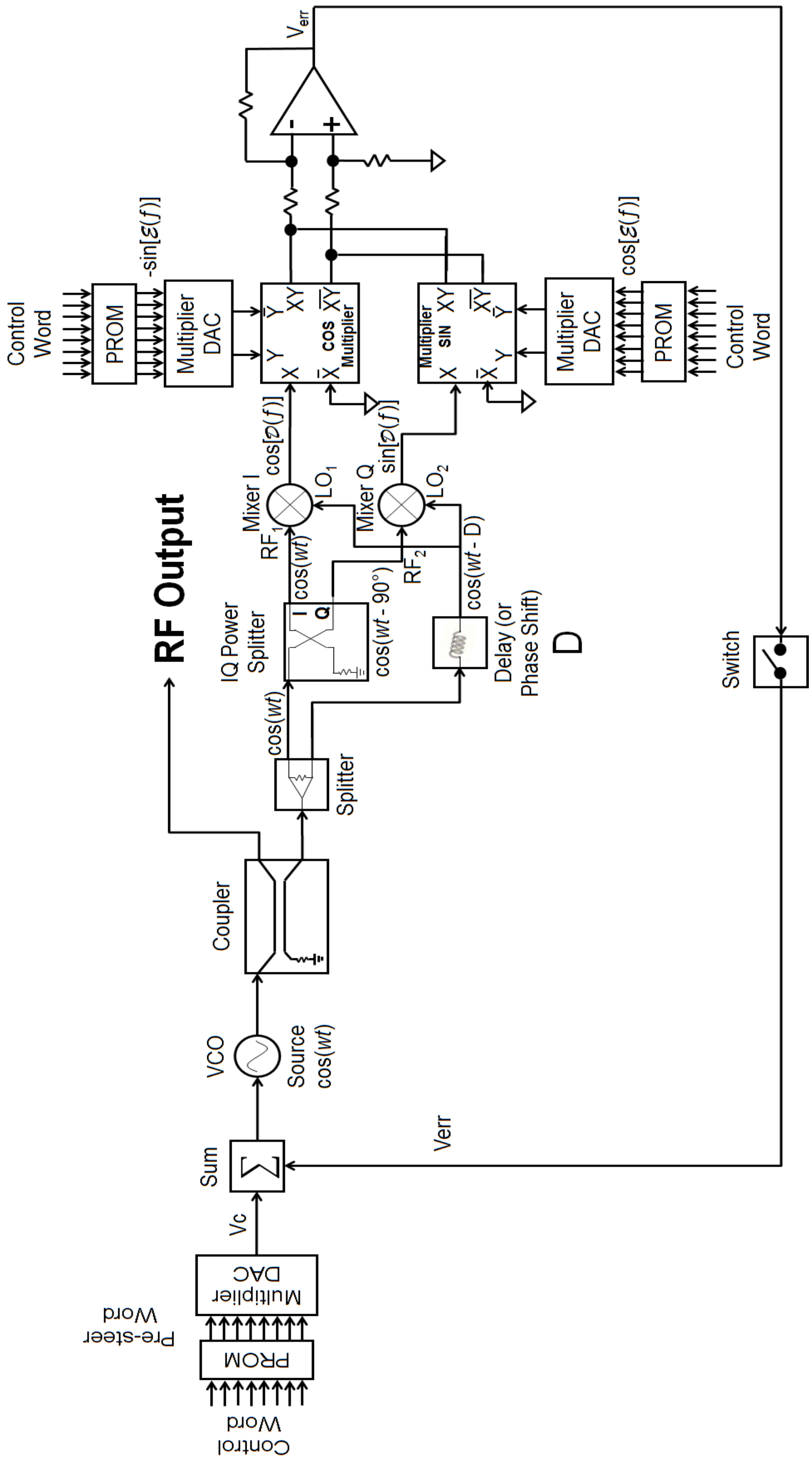


Figure 45 - The completed design.

It can be seen in Figure 45 that some new elements have been introduced. The purpose of those components should be somewhat obvious. However, a brief mention of those elements is included.

Verr:

Verr has been fed back to the oscillator and summed with Vc. Vc is the pre-steered voltage to get the oscillator on frequency. Of course, the VCO will drift, and it is shown here that Verr is summed back in to correct for frequency drift.

Also, a switch has been added to the feedback loop. This is to facilitate opening and closing the loop during test and linearization (to be discussed). Ideally, the switch will be computer controlled.

PROMs:

PROMs have been added to the circuit. As DAC words are collected, those values will need to be stored in the PROMs. Control words will come into our circuit and our PROMs will have a lookup table on how to adjust the DACs to yield the desired frequencies.

Coupler:

A coupler has been added. The point of all this was to achieve a stable frequency. Obviously, there must be some means to access that signal.

Control Word:

It may not be obvious, but “Control Word” appears in three locations, and it indicates that it is the same control word at all three locations simultaneously. The PROMs, of course, will have different lookup tables but will all use the same control word.

Adding a delay line – Discussion of length:

A delay line will have some physical length. In the section "Adding A Delay Line", it was stated: "For the proposed design, the delay element will be a "long" length of semi-rigid cable." The actual physical length of the delay line has properties worthy of a discussion. The goal in this section is to give the reader an intuitive feeling for the effect of changing delay line lengths.

In the top of Figure 46 there is an image of a sin wave incident on a "short" length of delay line. In this case the delay line is one wavelength long. Clearly, as the sin wave escapes from the delay line the phase would be 0°, or zero phase shift.

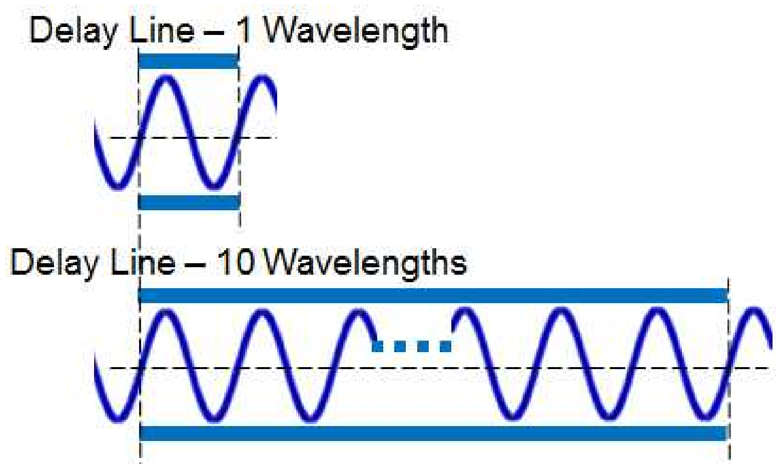


Figure 46 - Delay Line Length.

If the frequency of the wave were to increase slightly by some amount, the wavelength will be shorter. This, also, should be obvious. The phase of the escaping wave will now be some value greater than zero.

As an example, the frequency could increase by some amount such that the escaping wave now has 1° of positive phase shift.

If this were the case, to be useful the detection circuit must be able to observe the 1° phase shift and possess the ability to compensate for it. Certainly 1° of phase shift is not a lot of shift.

If, however, the delay line was to grow 10 times longer, or if it were to become 10 wavelengths long, that 1° of phase shift per wavelength is additive. In other words, the escaping wave will now be shifted by 10°.

To summarize, as the length of delay line increases, the sensitivity of the loop increases. For locking a loop, this would be recognized as a capture window. As the frequency shifts slightly, there will be a bandwidth of frequencies, or capture window, that the loop will be able to sense and for which it will be able to compensate.

Adding a delay line – Discussion of temperature stability:

One very nice property about a delay line formed from a semi-rigid cable is the temperature stability. Because the length of line is important, if it changes, that will negatively impact the circuit performance. Inherently, semi-rigid cable is rather rugged with regards to environmental conditions. So, without any extra "protection" it is already an insignificant source for error. However, it certainly is possible to attach a simple thermoelectric heater to a coil of semi-rigid cable. In this way, the cable variability can all but be eliminated for most applications.

How To Linearize

It should be assumed that the VCO and its DAC will not behave linearly. For any give DAC word, the VCO frequency cannot be predicted. Software must be written to linearize this design. During linearization, the VCO DAC is swept through its range and the resulting frequencies are collect. In addition, it has been demonstrated that the delay line behaves differently at different frequencies. Once again. The delay DACs must be swept through their range and data collected.

During linearization, the circuit is characterized, and digital words are collected. Those digital words can then be programmed as look up tables in the PROMs. These are the steps.

- During linearization, the switch in the feedback loop must be open. Otherwise, the loop will try to acquire, it will push the oscillator, and the data being collected will be garbage.

- With the circuit running open loop, increment the Vc DAC until the desired frequency is reached. This DAC value will be the first pre-steered value.

- Now measure Verr. Of course, it is desired to have Verr to be zero.

- While monitoring Verr, begin sweeping through the cos/-sin DAC values using the pairs of words collected in section “Controlling the DACs – A Detailed Study”.

- Continue through these pairs of words until Verr is as close to zero as possible. Since it is sweeping sinusoidally, it will approach zero and begin moving away. So, vary it a bit until the best Verr is determined.

- As discussed in “The Remaining Unknowns” Figure 30, the circuit may be operating at the wrong locking point. It is really very easy to check. Simply close the loop. If the oscillator stays at the proper frequency, then the data was collected at the correct point on the curve. If, however, the oscillator rails away from the desired frequency, then the Verr DAC value was incorrect.

- If the Verr DAC value was incorrect, then open the loop and simply continue sweeping through the set of cos/-sin DAC values using the pairs of words collected in section “Controlling the DACs – A Detailed Study”. As mentioned, Verr sweeps sinusoidally. So, it will eventually return to Verr equal to zero at some second pair of words.

- Once again check to see if the set of words is correct by closing the loop.

- Once it has been determined that the proper point on the linearization curve is being used, repeat the process through all Vc DAC values required for proper operation. It will be unnecessary to close the loop after each point. Once the correct point on the curve has been found, all subsequent DAC words will be relatively close.

- After all data has been collected, program the PROMs.

- The circuit is done.

Summary

This chapter has outlined a solution to the problem of an inherently noisy high frequency oscillator utilizing a delay line approach which employs both analog and digital elements and includes a brief discussion on the software required to control and linearize the final circuit.

This chapter was organized much like a tutorial where each element is introduced, explained, and then added to the solution so that the reader could gain the basic skills and theory necessary to be able to design a solution to the problem using a delay line, analog elements, and digital elements as well as an understanding of the software requirements to implement the completed solution.

One very significant advantage of the delay line version of this solution over more common traditional solutions is the absence of a separate LO oscillator. This solution presumes the oscillator has some amount of stability and then uses itself, delayed in time, as its own reference source to determine if changes are occurring. And if changes occur, it can “Sense” its own drift and then it can “Compensate” for the drift.

Also, since the delay line is realized with a length of semi-rigid cable, it is somewhat stable with regards to environmental conditions. Certainly, it can be used in conjunction with a thermoelectric heater making it even less likely to vary due to environmental conditions.

An even more significant variation to the solution, presented here, was the use of digital elements in place of their more common analog counterparts. In this application, the digital elements have less variability, are more standard, and less expensive than their analog counterparts.

Finally, this chapter offers to the reader a step-by-step suggestion on the linearization and collection of data.

Chapter 7 - Foundation of Structured Architecture, System & Cost Modeling

Modern software packages exist to estimate system cost early in the system development and procurement process. This paper begins the development of a structured systems engineering approach to system design. This paper defines a standardized modular diagram for a RADAR system applied to military applications in the aerospace industry. This modular diagram with sub-system block elements will be used to create a system model. The standardized modular diagram will also be used to create a cost model, using the same modular sub-system block elements and industry standard historical cost data. The commercially available software packages which estimate system cost are limited in their ability to aid in system optimization towards multi-objective cost and performance goals, as many require a completed system design. Methods are needed to determine which components in a system would benefit from additional modeling such as using a Multiphysics approach, and which design approach provides the best value (cost vs. performance) to the system. These methods are needed during concept development to aid in system scoping and cost estimation. To illustrate the benefits of cost optimization during early stages of design, this paper describes a sensitivity analysis approach applied to the design of an engineering system. This process seeks to use sensitivity analysis and a spiral design process to determine which cost drivers have the highest influence on overall system cost, and to realize high system performance while minimizing costs.

This work demonstrates that a system can be defined as a standard set of block diagrams for an airborne RADAR for military applications created by integrating a wide sample of the available examples. And where each of the example block diagrams could be considered a subset of the more generalized form. This work describes using the generalized block diagrams to create a WBS structure as the foundation for both a system model and a cost model. This work applies a sensitivity analysis to a cost model to direct a system designer towards a trade study for the purposes of system optimization. And finally, this work introduces a method using component cost sensitivity to determine the range of possible cost improvements to bound project return on investment.

Introduction

There are several very good commercially available cost estimation packages. To use these packages, first a system must be defined. The system must be defined in terms of hardware blocks. The hardware blocks can be arranged with a hierarchy such as a Work Breakdown Structure (WBS). Once the system is defined, the system can be entered into the cost estimation package. The package essentially converts each hardware component into a corresponding cost. In this way the cost of a system can be estimated.

To analyze a system, it is necessary to have a system upon which to perform the analysis. Typically, for a new effort a system is defined from the perspective of the designer where certain features were a priority to the respective designer. Those priorities are reflected in the various block diagrams which are produced and can be seen as areas of increased fidelity while other areas of the block diagram are simplified or even combined with other functions into one sub-block. For the current example of an airborne RADAR for military applications there are many block diagrams examples within the existing literature. However, although the end application is the same, the various block diagram examples vary widely. It can be considered that the block diagrams were tailored for each application and demonstrate the priority of the respective designer. This paper demonstrates the development of a generalized set of block diagrams for an airborne RADAR for military applications. The block diagrams were created by integrating a wide sample of the available examples where each of the examples could be considered a subset of the more generalized form.

The focus of this paper will be divided into four main topics: block diagram development, systems engineering model development, systems cost model development, and sensitivity analysis concepts applied to a system cost model.

The first section, Block Diagrams, will discuss the available literature on the topic. Specifically, research into the existence of industry standard block diagrams for an airborne based RADAR for military applications and subsequently the development of one where a standard did not exist. The level one system RADAR block diagram is defined along with the level two sub-blocks: antenna, transmitter, synchronizer, receiver, etc. A solid block diagram is frequently the best way to begin a new design. It becomes a pivot point upon which everything else is developed. All major radio frequency (RF) interfaces and divisions of functions can be seen (digital control will not be addressed).

In the second section, Model The System, a structured approach to system engineering is started and the elements of the block diagram are described. The elements are ready to be loaded into a system engineering tool and form the basis of future work which would be expanded to include operational view diagrams, logical view diagrams and other system engineering artifacts.

In the third section, Model The Cost, a structured approach to system cost modelling is discussed and the format of the model is described. The cost model utilizes the same functional blocks as defined in the block diagram section. This forms the basis for future work which will eventually lead to a robust modular cost model to describe a range of RADARs and their associated estimated costs.

In the fourth section, Sensitivity Analysis Applied To A Cost Model, a concept is introduced whereby a sensitivity analysis could be applied to a cost model to direct a system designer towards a trade study for the purposes of system optimization. In addition, it is shown that a sensitivity analysis provides an upper bound of potential cost improvements which then forms the basis for a Return On Investment (ROI).

This work is novel in that it demonstrates that a system can be defined as a standard set of block diagrams for an airborne RADAR for military applications created by integrating a wide sample of the available examples. And where each of the example block diagrams could be considered a subset of the more generalized form. This work is novel in that it describes using the generalized block diagrams to create a WBS structure as the foundation for both a system model and a cost model. This work is novel in that it introduces a sensitivity analysis applied to a cost model in order to direct a system designer towards a trade study for the purposes of system optimization. And finally, this work is novel in that it introduces a method using component cost sensitivity to determine the range of possible cost improvements to bound project return on investment.

Related Work

Literature Assumptions and Search Terms:

The available literature was consulted, primarily using Google searching. The assumption of the author was that a standardized block diagram for an airborne RADAR for military applications already exists. The assumption was that there was a standard upon which all designs were based. Research was done using keyword search terms such as “standard block diagram”, “RADAR block diagram”, “airborne RADAR block diagram”, etc.

Literature Results:

For each effort, many search results were obtained. There were countless block diagrams for all types of RADARs. And, for the specific platform of airborne RADAR, again, there were many different results. However, although the results varied each version of a block diagram had some similarities. The differences appeared to be due to the focus of the respective system designers. In other words, if the designer's focus was upon a specific sub-function, that area of the block diagram had significantly more fidelity. Conversely, other areas of the block diagram would be abbreviated or even combined with other sub-functions. In this way the designer could highlight an area or sub-block for increased emphasis.

Block Diagrams

A robust block diagram is frequently the best way to begin a new design. And it is a recommended first step. It is not uncommon for engineers to jump right into a design and begin designing. Each engineer responsible for a portion of the system has ideas on how to best proceed. And frequently those best ideas are competing rather than complimenting one another. Therefore, a robust discussion early on regarding system goals is critical. Without a clear set of system goals, it is unlikely a system will be designed correctly on the first attempt. And a first step towards defining those goals is to create a block diagram. It becomes a pivot point upon which everything else is developed. All major interfaces and divisions of functions can be seen.

Not only do block diagrams align system and sub-system designers but also block diagrams are key tools for cost analysts. The most obvious area of contribution is defining interfaces. With a good visual representation, interfaces are considered, and meaningful requirements can be created. Those requirements affect many variables including cost, performance, life span, operation, etc. A cost analyst does not need to be a system designer. But having some familiarity with the basic building blocks of the system is critical. A good cost analyst should actively participate in the early generation of system block diagrams to help influence the direction of the system design.

Interfaces and Functions:

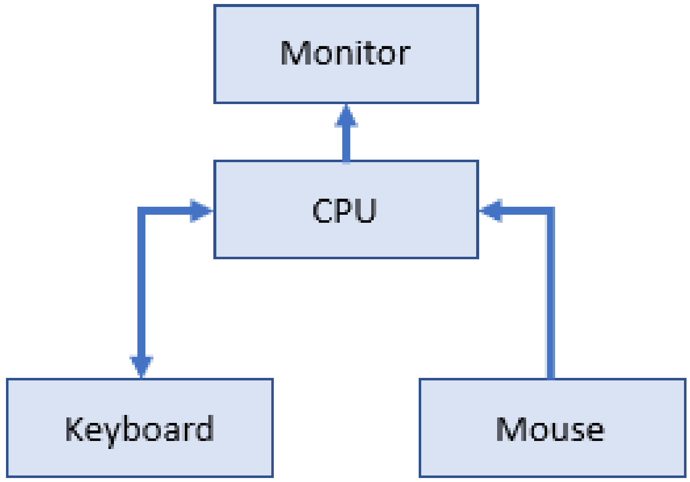


Figure 47 - Computer Block Diagram, Simple.

Figure 47 shows a simple block diagram of a standard computer. In this case, the computer is made up of four sub-blocks. For purposes of this paper, the entire computer will be considered Level 1 while the sub-blocks indicated in the figure will be considered Level 2.

From this figure, the central processing unit (CPU) is the main sub-block in that it interfaces with all the other blocks. And none of the other sub-blocks talk directly to one another. By observing the arrows, some of the interconnects are 2-way communication, such as between the keyboard and CPU, while other interconnects are 1-way communication, such as from the CPU to the monitor. In addition, there clearly are four blocks. Each block is labeled by function. Each block has distinct responsibilities for the system performance. And it could be clearly defined what those interfaces should be for those blocks to communicate with one another. For a team designing a computer system, this very simple block diagram already contains very valuable information which will help guide the designers towards a successful system design. This is the value of a block diagram. It can be done early, simply, and it can contain an enormous amount of critical system information.

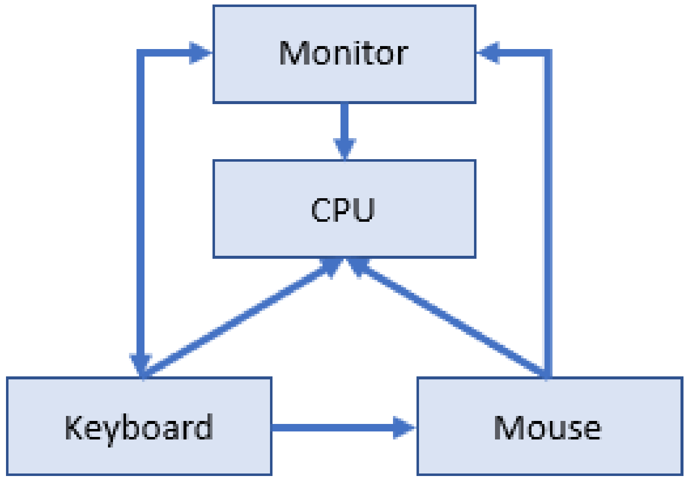


Figure 48 - Computer Block Diagram, Complex.

As a comparison, imagine a block diagram as the one indicated in Figure 48. Although the blocks are the same as Figure 47, the interfaces are clearly more complex. And the interactions between the blocks more closely resemble a network rather than a command-and-control structure such as that indicated in Figure 47. Clearly, block diagrams offer a shorthand to an enormous amount of information in a simple easy to read format.

Standardized RADAR Block Diagram:

With no clear standardized block diagram, the author used the available information to piece together one comprehensive solution (**[8] [9] [10] [11] [12] [13] [14] [15] [16] [17] [18]**). The goal here was to generalize the sub-blocks in such a way as to incorporate a wide sample of the available examples. Any sample block diagram could be considered a simplified, or tailored version of the more generalized form.

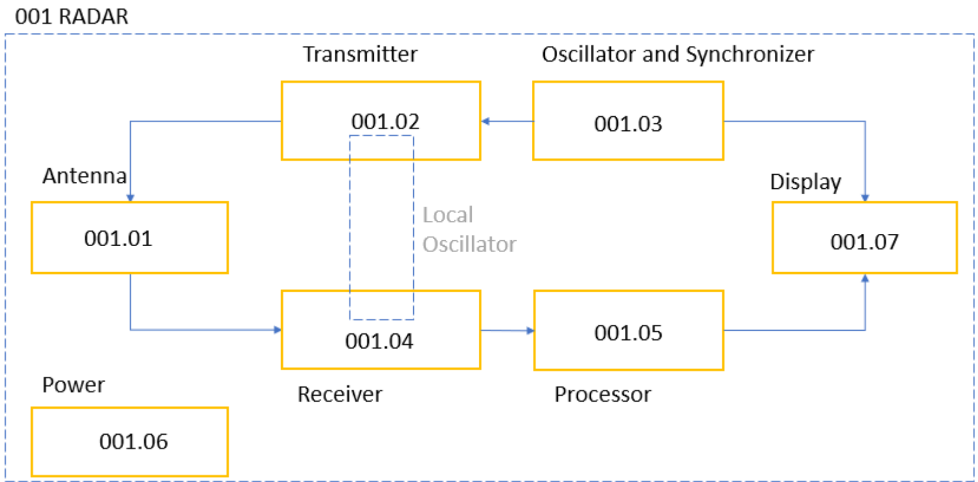


Figure 49 - Generalized RADAR System Block Diagram.

Figure 49 is a generalized RADAR system block diagram for an airborne based military application. The outer dashed line can be considered Level 1, the complete RADAR System. The sub-blocks indicated in the figure comprise the Level 2 blocks and consist of antenna, transmitter, etc. This block diagram as well as the Level 3 block diagrams appear in the appendices.

What can be concluded here is that every airborne RADAR system for a military application will have an antenna. Antenna designs can vary widely. It could have any number of radiator elements: 1, 10, 100, 1000, etc. It could have any type of radiator: notch, patch, whip, etc. However, it most certainly will have some form of antenna. And so that sub-block appears in the block diagram. In this case it has been labeled 001.01 signifying the first (01) of the Level 2 sub-blocks. All of the sub-blocks have been correspondingly numbered. This will come up again within this paper when the WBS structure and models are discussed.

The block diagram also contains arrows which demonstrate the direction and flow of information between the sub-blocks. The directional flow is exclusively 1-way. It should be noted that this is limited to the signal flow including radio frequency (RF) interfaces. There could potentially be cases of multi-directional flow for purposes of digital control which are not addressed. For example, the processor might turn off the transmitter and then receive some feedback that the transmitter has indeed been disabled. However, that is a control signal and is not captured by this signal flow diagram.

The local oscillator appears as a dashed line and crosses into the transmitter and receiver as well as the white space in between. This is done because while those two sub-blocks require a local oscillator (LO) for operation, in some hardware configurations they have a resident dedicated LO, while in some configurations there is a separate sub-block dedicated to the LO function. And this representation is deliberately created to accommodate either physical hardware solution or implementation.

The descriptions for each sub-block (Level 1, 2, or 3) were generated in the same manner as the block diagrams. The available literature was widely explored, and the various descriptions were collected. Then, the various descriptions were combined into a higher order, more general version for which all descriptions could be considered a simplified, or tailored version of the more general versions presented here.

Sub-Block: 001.01 Antenna:

A numbering convention was selected, and each element is assigned a unique identifier. The numbering convention identifies hardware "Levels" (1, 2, 3, etc.). The antenna appears in Figure 50 as element 001.01 which was previously noted to designate the first of the Level 2 elements. The numbering convention was consistently applied throughout the development for the block diagrams, system model, and cost model.

The antenna **[19]** is the coupling element between free space and the other RADAR elements. The antenna transfers the RADAR energy from the transmitter into free space. And the antenna collects the echo energy from free space and delivers it to the receiver for down conversion and processing.

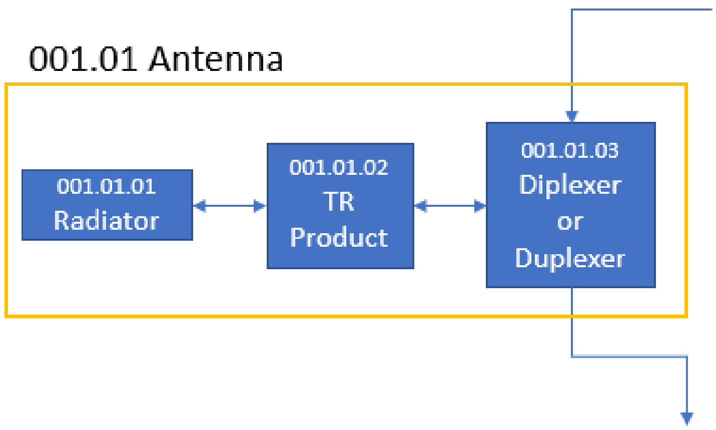


Figure 50 - Antenna Block Diagram.

As illustrated in Figure 50, the antenna sub-block can be further decomposed into Level 3 blocks: radiator, transmit-receive (T/R) product, and duplexer. The element numbers have been assigned as indicated in the figure.

An anticipated criticism regarding this block diagram would come from the perspective of the hardware configurations. Typically, for an airborne system, there are two main hardware configurations. The first typical configuration is where each element is independent. Each element has a radiator and a T/R product. And then a bank of those channels is combined to make an array, a vertical configuration. The second configuration is where all the radiators are assembled in a bank of radiators, almost like a plate of radiators. And then those radiators are mated against a bank or plate of T/R products, a lateral configuration. A designer might argue that the image captured in Figure 50 is only one of the two possible configurations. However, the representation in Figure 50 is a functional view irrespective of the hardware configuration. Therefore, both configurations are applicable. The block diagrams contained within this document are created to showcase the functional sub-blocks and the associations between them. The diagrams were intended to satisfy all physical instantiations.

The radiator is an exchanger between the propagating waves and the electric currents. The T/R product is a device where common circuitry for both transmit and receive functions are combined into a single module or element. The duplexer in a high-power RADAR system is the element that switches the antenna path between the transmitter and the receiver paths for a system where the two paths share an antenna. It is also used to protect the receiver from high power transmissions entering directly from the transmitter.

Sub-Block: 001.02 Transmitter:

The transmitter appears in Figure 51 as element 001.02. The transmitter **[20]** modulates, or up converts, the wave to a transmission frequency. Then, if required to increase the signal power before transmission, the transmitter amplifies the wave for the antenna to send into operating space.

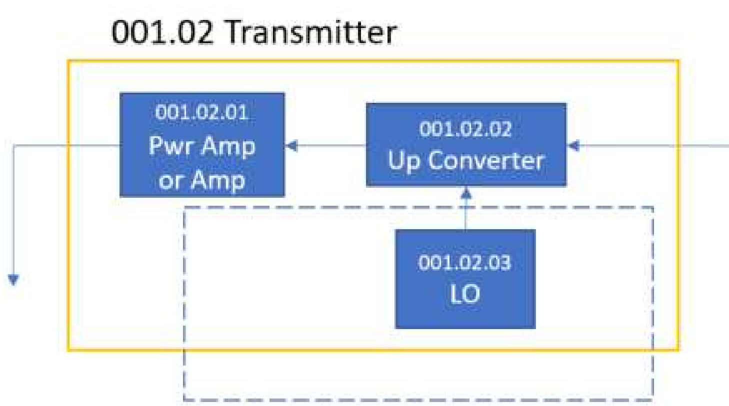


Figure 51 - Transmitter Block Diagram.

As illustrated in Figure 51, the transmitter sub-block can be further decomposed into Level 3 blocks: power amplifier, up converter, and local oscillator (LO). The element numbers have been assigned as indicated in the figure.

The power amplifier **[21]** is a device that converts a low power signal into a higher power signal. In the past, the high-power amplifier was more likely to be some sort of traveling wave tube. But certainly, the more contemporary approach would be a solid-state high-power amplifier. Regardless of the hardware configuration, the block diagram is a functional view and represents either approach. The LO is an oscillator which is used to change the frequency of the signal.

Local oscillators **[22]** often employ some means of a phased locked loop (PLL). Typically, it is easier to have a stable oscillation when the frequency generated is low. And it is easier to generate a high frequency oscillation when the oscillation is less stable. By means of a PLL, it is possible to take the best of both and create a device which is stable at high frequencies. In the paper Digital Control Of Frequency Locked Oscillator, Microwave Journal March 2020, stable high frequency oscillations were achieved by locking a single oscillator to itself. It was accomplished by passing the signal through a long semi-rigid cable and then frequency locking to the time delayed reference.

The up converter is a nonlinear electrical circuit that creates new frequencies from two signals applied to it. Most commonly, this is accomplished with a mixer **[23]**. In the case of a transmitter, the mixing product is a multiple of the sum of the input signal and the LO signal. The purpose of the up converter is to modulate the signal from the synchronizer for the antenna. But it is widely known that by means of a mixer, multiple harmonics are generated. And, by use of filtering, any higher order harmonic can be isolated, amplified, and used as the mixing product.

Sub-Block: 001.03 Synchronizer:

The synchronizer appears in Figure 52 as element 001.03. The synchronizer coordinates the timing of the RADAR. It generates timing pulses that are used to control the RADAR pulse repetition frequency (PRF). Signals are sent simultaneously to both the transmitter and the display to align the sweep echo pulses.

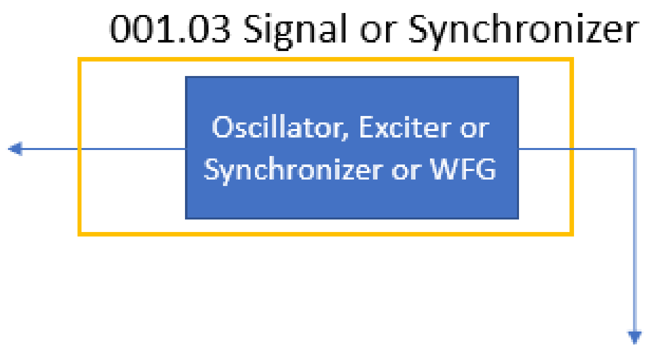


Figure 52 - Transmitter Block Diagram.

The Synchronizer sub-block can be further decomposed into Level 3 blocks. However, for purposes of this analysis, the synchronizer will be considered as elemental, and cannot be further subdivided. This is done because there does exist a variety of synchronizer architectures and further analysis will need to be performed to determine a standardized Level 3 architecture. Thus, no element numbers have been assigned as indicated in Figure 52.

Sub-Block: 001.04 Receiver:

The receiver appears in Figure 53 as element 001.04. The receiver detects an incoming echo signal bounced off a target, receives, amplifies, demodulates, and converts the analog signal to digital format for further analysis in the digital processor.

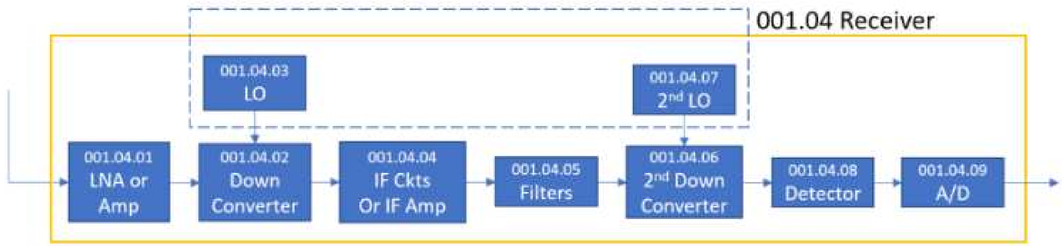


Figure 53 - Receiver Block Diagram.

As illustrated in Figure 53, the receiver sub-block can be further decomposed into Level 3 blocks: low noise amplifier, down converter, intermediate frequency (IF) amplifier, filters, 2nd down converter, detector, and analog to digital converter. The element numbers have been assigned as indicated in the figure.

The local oscillator here is the same as that discussed in the RADAR and transmitter sections. As mentioned earlier, the LO is an oscillator which is used to change the frequency of the signal. In this application the LO is used to down convert a signal while in the transmitter it is used to up convert a signal. The low noise amplifier (LNA) boosts the signal while adding as little additional noise as possible. The goal is to maximize the signal to noise ratio (SNR) of the echo signal.

In a receive chain, the signal to noise ratio is determined primarily by the first element. The first element of the chain dominates the entire chain's performance. Low noise amplifiers, as the name suggests, are a special sub-class of amplifiers designed for this purpose. Therefore, an architect should assume an LNA as a first element.

The down converter is a nonlinear electrical circuit that creates new frequencies from two signals applied to it. Most commonly, this is accomplished with a mixer. In the case of a receiver, the mixing product is a multiple of the difference of the inputs signal and LO signal. The purpose of the down converter is to demodulate the signal and the RF frequency of the LNA to a lower or intermediate frequency (IF) where amplification and filtering can be done more easily. Generally, multiple mixers would be used.

Later in the signal chain, the signal will be converted from analog to digital. The signal must be digitized for meaningful data processing of a modern system. At some future time, it may be possible to directly convert an X-band signal to digital, but in today's practical terms that is not yet possible. Therefore, the conversion is required. The detector and analog to digital (A/D) converter, as their names imply convert an analog signal into a digital signal. To convert the signal from analog to digital, a digital clock must be used. If the transmitted signal is "high," using the Nyquist criteria, the sampling frequency must be "higher." This is the fundamental limitation of the A/D converter. As mentioned, as time passes the technology is improving, but in today's practical solutions, the architect's options are still somewhat limited.

Sub-Block: 001.05 Processor:

The processor appears in Figure 54 as element 001.05. The processor **[24]** decides if an echo is a target and determines if and how to present a depiction to the display. Typically, this may include number, location, and movement of targets.

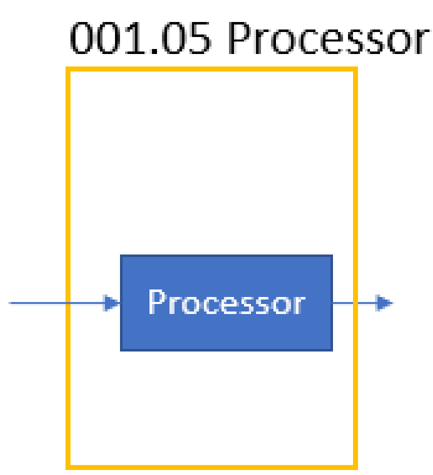


Figure 54 - Processor Block Diagram.

The processor sub-block can be further decomposed into Level 3 blocks. However, for purposes of this analysis, the processor will be considered as elemental, and cannot be further subdivided. This is done because the processor is primarily a digital device, and the focus of this analysis is upon the analog path for the signals. Thus, no element numbers have been assigned as indicated in Figure 59.

Sub-Block: 001.06 Power:

The power block appears in Figure 55 as element 001.06. The power block converts the primary power from the platform to the required forms needed for each sub-block.

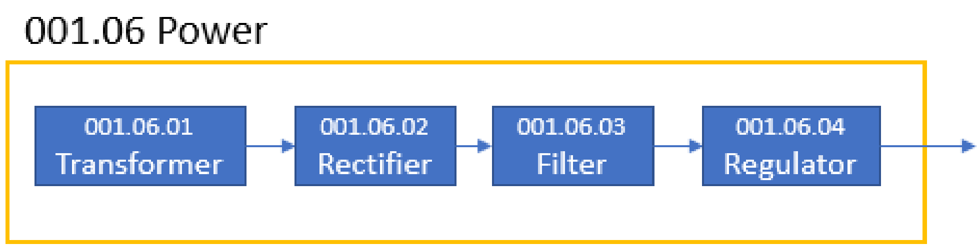


Figure 55 - Power Block Diagram.

As illustrated in Figure 55, the power sub-block can be further decomposed into Level 3 blocks: transformer, rectifier, filter, and regulator. The element numbers have been assigned as indicated in the figure.

The transformer **[25]** is a device which transfers electrical energy through electromagnetic induction. The transformer is comprised of coils. As current passes through a coil it generates an electro-magnetic field. If a second coil is placed within that field, a current is generated in the second coil. By adjusting the ratio of turns for each coil, a voltage can be stepped up or down. In a familiar power supply, such as for a laptop, the power supply converts 120 volts AC from the wall outlet down to 12 volts DC for use by the computer. The transformation from 120 volts to

12 volts, called a step down, is done by means of a transformer. Because the transformer is comprised of coils, the transformer is always the heaviest component in the power supply.

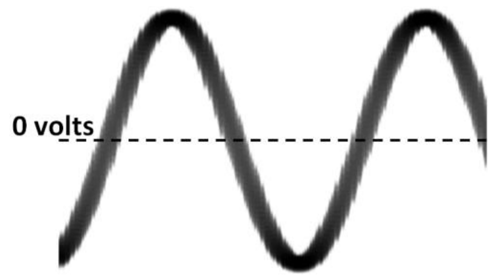


Figure 56 - Sinusoidal Signal.

The rectifier **[26]** is a device which converts electricity from AC to DC. The most common rectifier is a bridge rectifier and can be created with four diodes. For a sinusoidal signal (Figure 56) half of the time the voltage is positive, and half of the time the voltage is negative. Through the rectifier, the negative half cycle of the sinusoidal signal is flipped up to be positive (Figure 57). As a result, the wave form is now a series of positive going sinusoidal voltage "bumps," like humps of a camel's back.

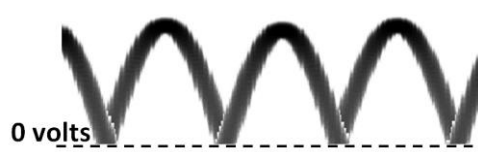


Figure 57 - All Positive Voltages.

The filter **[27]** is a device used to remove unwanted frequency components. After the voltage has been transformed into a series of positive going sinusoidal voltage "bumps," it is necessary to smooth it out. When the voltage value approaches zero the slope of the curve is negative. Once the voltage has reached zero volts, the voltage begins to rise and has a positive slope. The transition between a negative voltage slope and a positive voltage slope is instantaneous. The instantaneous nature of the voltage in the time domain corresponds to a high frequency effect in the frequency domain. By removing that high frequency component of the signal, by means of a filter, the corresponding waveform will be smoother.

The regulator **[28]** is a device which stabilizes a DC voltage independent of the load current. The signal, post filtering, approximates a fixed DC value. However, the value is not stable. The voltage continues to have some residual effects from its sinusoidal origin. For use in a system, voltages must be stabilized. A regulator removes frequency components of a "dirty" DC signal and clamps it to a predetermined value. Post regulation, the signal is a "clean" DC value.

Sub-Block: 001.07 Display:

The display appears in Figure 58 as element 001.07. The display presents a depiction, in a usable form, of received targets. Typically, this may include number, location, and movement of targets.

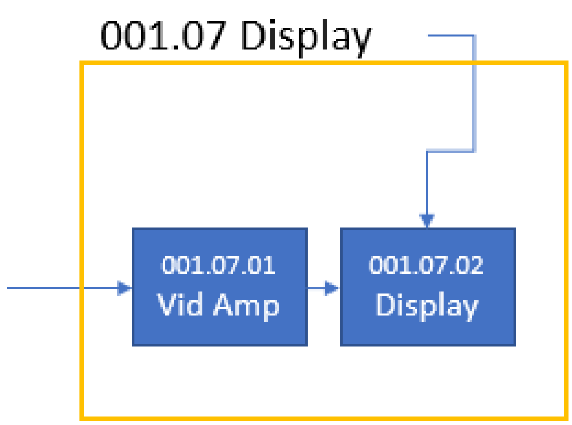


Figure 58 - Display Block Diagram.

As illustrated in Figure 58, the display sub-block can be further decomposed into Level 3 blocks: video amplifier **[29]** and display. The element numbers have been assigned as indicated in the figure.

The video amplifier is a device which is designed to process video signals. The display is a device which is used for presenting images and video. Typically, this would be a cathode ray tube (CRT) or more recently some type of liquid crystal display (LCD) such as a laptop screen or monitor.

Model The System

Once a rigorous block diagram for a system has been created the next step is to begin modeling the system, or otherwise referred to as architecting a system.

Not only do system models align system and sub-system designers but also system models are key tools for cost analysts. System models include many artifacts such as documented requirements, documented use cases, logical view diagrams, operational view diagrams, etc. Just as with block diagrams, a cost analyst need not be an expert in all these areas but certainly a firm understanding would be most helpful. An awareness of system modeling and the related artifacts enables a cost analyst to better understand the system and then to estimate a cost with a higher fidelity. Just as with block diagrams, a good cost analyst should actively participate in the development of a system model to help define the system design.

There is a very important book on the topic entitled “Architecting Information-Intensive Aerospace Systems” by Dr. John M. Borky **[30]**. In it, the author writes that architecting is done “to create systems and enterprises that are well organized, expandable, and evolvable, robust under the stresses of real-world use, and affordable to own and operate. In short, the essence of the art and science of architecture is manifested in results that are beautiful in the eyes of their users while satisfying those users' practical needs."

Robust models created through Model-Based Systems Engineering (MBSE) are the foundation of the entire System Engineering (SE) process and provides a clear and unambiguous definition of the system.

While there are several tools which may do similar functions, for architecting the system in this paper, the COTS software tool used was chosen because it contained all the tools required to document requirements, document use cases, create logical view diagrams, operational view diagrams, and other system engineering artifacts.

System Modelling Approach:

When creating a structured architecture utilizing a COTS system engineering tool, one very useful structure is indicated in Table 3.

Table 3 - System Model Structure.

System Engineering Model
Components
Internal Block Diagrams
Packages
a_Requirements
b_UseCases
c_Structure
d_Behaviour
e_Data
f_Services
g_Context
PredefinedTypes (REF)
z_Default

These items are referred to as packages. This is a very solid structure and provides an architect with designated locations for creating system artifacts. With a structure such as this, virtually any artifact required, or created, can be sorted, and stored into one of these packages.

RADAR System Modelling Structure:

An indentured set of numbers was created for this system and appears in Table 4. In Table 4, the hierarchy of the system design with Level 1, 2, & 3 sub-block names and numbers can be seen. These numbers form the basis for a Work Breakdown Structure (WBS).

Table 4 - Indentured System Numbering Structure.

Level 1	Level 2	Level 3	Block Name
001			Radar
	001.01		Antenna
		001.01.01	Radiator
		001.01.02	TR Product
		001.01.03	Duplexer
	001.02		Transmitter
		001.02.01	Power Amplifier
		001.02.02	Up Converter
		001.02.03	Local Oscillator
	001.03		Synchronizer
		001.03.01	Synchronizer
	001.04		Receiver
		001.04.01	Low Noise Amplifier
		001.04.02	Down Converter
		001.04.03	Local Oscillator
		001.04.04	IF Amplifier
		001.04.05	Filters
		001.04.06	2nd Down Converter
		001.04.07	2nd Local Oscillator
		001.04.08	Detector
		001.04.09	Analog to Digital Converter
	001.05		Processor
		001.05.01	Processor
	001.06		Power
		001.06.01	Transformer
		001.06.02	Rectifier
		001.06.03	Filter
		001.06.04	Regulator
	001.07		Display
		001.07.01	Video Amplifier
		001.07.02	Display

Work Breakdown Structure (WBS):

A Work Breakdown Structure (WBS) (MIL-STD-881D) is a tool used to define a project in discrete work elements in a hierarchical format. It displays and defines the product, or products, to be developed and/or produced. It relates the elements of work to be accomplished to each other and to the end-product. As described in the SmallBusiness website **[31]**, "The main purpose of a WBS is to reduce complicated activities to a collection of tasks." It is a very useful management tool. By arranging the architecture in such a manner, not only will it describe the breakdown of the hardware from Level 1 to Level 2 and so on, but it can also form the foundation of the set of tasks required to design the hardware. For example, a radiator is designated as block 001.01.01. The design cycle for any block, such as a radiator, will likely follow a standard design cycle: requirements, preliminary design, detailed design, and integration, verification & validation (IV&V). Those are phases which are made up of tasks and could be designated with the further indentured designators:

001.01.01.01 Requirements Phase
001.01.01.02 Preliminary Design Phase
001.01.01.03 Detailed Design Phase
001.01.01.04 IV&V Phase

These phases could additionally be designated with even lower-level numbers and even more specific tasks. The point is, creating a system structure with an eye towards a WBS is a best practice and frequently is a contract requirement.

RADAR COTS System Model:

The RADAR system demonstrated earlier as a set of block diagrams can be loaded into a COTS system engineering modeling tool (Table 3). The information from the block diagrams could be loaded into the requirements package. The various block diagrams can be loaded into the structure package. And, in general, all system artifacts could be documented within the COTS tool.

The numbering and indenture of the entries should remain consistent with that presented in the earlier sections of this paper (Table 4). The specifications within the requirements package should also be consistent with the information contained within the block diagrams.

A robust system model helps to align system and sub-system designers and are key tools for cost analysts. A good cost analyst need not be an expert in system engineering tools. But some familiarity with system engineering tools would be very advisable. And early participation in a product life cycle will help a cost analyst to not only influence the direction and the development of the system design but also to then be in a far better position to generate system cost estimates.

Model The Cost

Once a rigorous block diagram and system model for a system has been created the next step is to begin modeling the system cost. As mentioned, with early participation in a product life cycle a cost analyst will be in a far better position to generate system cost estimates.

A new emphasis introduced here is a structured approach which includes a modular approach to modeling the system cost. At a later phase in the system design, it will be necessary to perform trade studies. The most common trade will be between two performance profiles. For example, "better" performance using more power vs. "worse" performance using less power. This is a very common trade in industry.

To achieve the two profiles, a modular approach will be used to swap out blocks for either "better" or "worse" performance. This is the elegance of a modular approach to system architecture. If a parallel effort could be taken to create a corresponding cost model for each block, then as blocks are swapped in and out for performance trades, a cost trade could simultaneously be performed.

As with system modeling, there are several COTS cost tools which may do similar functions. For costing the system in this research, the software package utilized was selected because it contains all the elements required to enable a user to create a modular cost model. Blocks can be created and turned on and off to simulate substituting one block for another. For each cost model block, there are parameters which can be adjusted to influence cost. Those parameters correspond to various ranges of hardware design details ranging from a very high level of detail to a very low level of detail depending on the user's familiarity with the hardware being modeled. All the cost data within the COTS tool is pulled from industry standards, so no cost data needs to be loaded. Of course, for any user the tool data can be modified for specific applications and past performance actuals. Some cost tools include information regarding the sensitivity of a particular parameter being adjusted. The presentation of the sensitivity factors is currently a bit crude, but it should be possible to pull the data for further analysis outside of the tool.

Cost Modelling Approach:

Unlike the system tool structure which focused on artifacts and a manner by which to organize them (See Table 3), a cost tool focuses on the hardware, or more precisely, the indentured organization of the hardware. This allows a system architect to utilize the WBS numbering system directly in the tool. The tool directly estimates cost based on what hardware will be included. So, to create a cost model a user needs to first consider the indenture of the hardware.

Level 1 – Deliverable hardware
- Level 2 – 1st Sub-block of hardware
- Level 2 – 2nd Sub-block of hardware
- Level 2 – 3rd Sub-block of hardware
- Etc.

RADAR System Cost Modelling Structure:

As with the system model, an indentured set of numbers was created for this system cost model and appears in Table 5. The entries in Table 5 are very similar to those from Table 4. However, Table 5 has additional rows for "Roll Up." A cost tool could call out Level 2 hardware, for example an antenna. However, an antenna is also a collection of Level 3 hardware blocks. In this case, both options are included in the cost model. And when the model is run to produce an estimate either, but not both would be selected.

Table 5 - Indentured Cost Numbering Structure.

Level 1	Level 2	Level 3	Block Name
001			Radar
	001.01		Antenna
	001.01RU		Antenna Roll Up
		001.01.01	Radiator
		001.01.02	TR Product
		001.01.03	Duplexer
	001.02		Transmitter
	001.02RU		Transmitter Roll Up
		001.02.01	Power Amplifier
		001.02.02	Up Converter
		001.02.03	Local Oscillator
	001.03		Synchronizer
	001.03RU		Synchronizer Roll Up
		001.03.01	Synchronizer
	001.04		Receiver
	001.04RU		Receiver Roll Up
		001.04.01	Low Noise Amplifier
		001.04.02	Down Converter
		001.04.03	Local Oscillator
		001.04.04	IF Amplifier
		001.04.05	Filters
		001.04.06	2nd Down Converter
		001.04.07	2nd Local Oscillator
		001.04.08	Detector
		001.04.09	Analog to Digital Converter
	001.05		Processor
	001.05RU		Processor Roll Up
		001.05.01	Processor
	001.06		Power
	001.06RU		Power Roll Up
		001.06.01	Transformer
		001.06.02	Rectifier
		001.06.03	Filter
		001.06.04	Regulator
	001.07		Display
	001.07RU		Display Roll Up
		001.07.01	Video Amplifier
		001.07.02	Display

RADAR COTS Cost Model:

The RADAR system demonstrated earlier as a robust set of block diagrams and system model can be loaded into a COTS cost estimation tool. With the structured approach, the numbering and indenture of the entries should remain consistent with that presented in the earlier sections of this paper (Table 5).

As part of the COTS tool, each sub-block contains functional parameters which can be tuned for the specific application. For example, the weight of the specific hardware sub-block could be modified. What is initially loaded is an industry standard value to be used as a starting point.

This is the domain of the cost analyst. A good cost analyst having participated early in the design life cycle of the product will have familiarity with the WBS and product requirements. With a robust cost model which mirrors the system model a cost analyst is well positioned to generate a robust cost estimate and can rapidly participate in trade study alternatives. Because of the modular nature of the model structure, it is possible to turn blocks "on" or "off" to select what is to be included for an estimate. In this way, blocks can be swapped in a modular fashion allowing a cost analyst to work with the system architect the ability to perform cost trades.

Multivariable Analysis & Trade Studies

A robust structured system modelling approach utilizes the concept of modularity. If the system is comprised of modules, then the possibility exists where modules could be swapped to modify the system for various performance characteristics. At the same time, if the cost model mirrors the system model, then as the system is being defined, a rough cost estimation could be determined simultaneously.

Even with a modular approach, when designing a system more than one variable must be considered. Choices are made regarding those variables. In most cases, variable choices have competing impacts. For example, one design architecture may have “better” performance using more power vs. “worse” performance using less power. Decisions for a sub-system need to be evaluated at a system level. A system designer needs to consider the design as a system and realize that any change potentially has an impact beyond the sub-system. It is not usually possible to make an architecture or hardware change irrespective of the larger view of the system. This is really the heart of system engineering, consideration of an entire system, not just a collection of sub-system parts.

This is particularly important when considering cost because it is not possible to swap out cost as modular blocks and estimate new costs without understanding that there are affects to the system. There are multilevel impacts when modular blocks are substituted. Simply swapping out a block and estimating cost gives a first order indication of the cost impact. But until the design is finalized it is only a rough estimate. There is a spiral approach to design. As choices are made, impacts are assessed, costs can be estimated, new choices are made, and eventually the design spirals into a solution.

To decide between competing variables a trade study can be employed. A trade study is a useful tool which allows a designer to compare and contrast the various possible choices to determine which solution would be “best” for the given application.

To perform a trade study, first the various options are clearly defined. Criteria must be selected. Criteria are the items which are impacted by the options. Typical criteria are cost, schedule, performance, supportability, etc. A matrix is made with the options vs. the criteria, Table 6. A grade is given in the matrix for each criterion and option. Then the criteria are assigned a weight. The grades are scaled by the weighting factors. And then a score for the options can be calculated by adding up the weighted grades for each option. The option with the highest score "wins" the trade study and represents the "best" solution.

Table 6 - Sample Trade Study Matrix.

		Option #1		Option #2		Option #3	
	Weight	Grade	Weighted Grade	Grade	Weighted Grade	Grade	Weighted Grade
Criteria #1	5%	9	0.45	1	0.05	9	0.45
Criteria #2	5%	9	0.45	5	0.25	9	0.45
Criteria #3	15%	9	1.35	5	0.75	5	0.75
Criteria #4	10%	5	0.5	9	0.9	1	0.1
Criteria #5	30%	5	1.5	5	1.5	9	2.7
Criteria #6	25%	5	1.25	5	1.25	9	2.25
Criteria #7	10%	1	0.1	5	0.5	5	0.5
Total	100%		5.6		5.2		7.2

Sensitivity Analysis Applied To A Cost Model

The limitation of the commercially available cost estimation packages is that it is essentially a unidirectional process. A user defines a system and uses the cost estimation package to estimate cost. The user can then experiment with alternatives or modifications to the system and estimate the corresponding associated system cost. What is missing is a bidirectional interaction with the software package. There is very little guidance from the cost estimation package which suggests to the user system modifications for consideration. It lacks suggestions to the designer which modifications would have the greatest impact to the overall cost of the system.

It is desirable to have a feature within a cost estimation package which can analyze the components of the system to determine which components have the greatest impact. In other words, which components have the highest sensitivity for modification as it pertains to the overall cost of the system.

Although sensitivity analysis is well understood the application of sensitivity analysis upon a cost model for the purposes of maximizing the impact to the overall system cost is novel. It should be possible, and is explored in a follow-on paper, an effort to generate a cost sensitivity algorithm of the various components in a system to analyze a system and determine which subsystem components in a chosen design solution have the highest sensitivity to cost for the overall system. The analysis should highlight the areas to which a system designer could apply focus to reduce the overall system cost early in the life cycle of a Program.

Sensitivity Analysis Potential:

Lack of adequate cost analysis tools early in the design life cycle of a system contributes to non-optimal system design choices both in performance and cost. A goal is to develop algorithms for an automated tool/approach utilizing cost element sensitivity to enable a system designer the ability to understand the relative cost impacts of various decision/choices which affect system design early in the design cycle for an airborne based RADAR system for military aerospace applications.

Most cost estimations are a unidirectional process. First a design is selected then the design cost is estimated. If the cost is not good the only feedback is typically to “reduce” cost. Then a new design is chosen, and the design cost is again estimated. But typically, the process lacks meaningful feedback which demonstrates how or where to make design changes to impact cost most significantly. Instead, the designer typically modifies an area of particular interest to the designer.

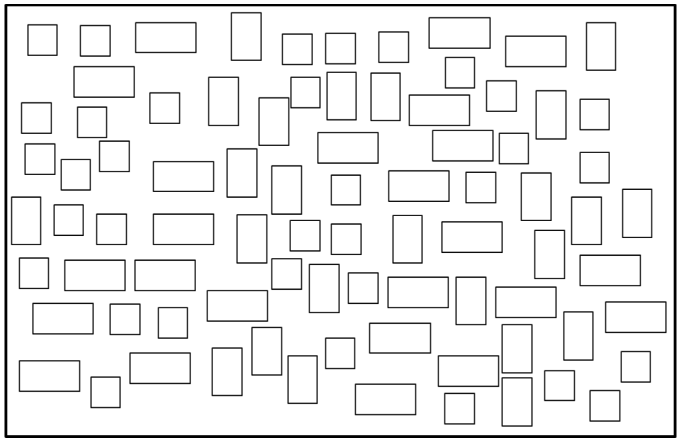


Figure 59 - Complex System of Sub-System Blocks.

Consider a complex system made up of sub-system blocks, Figure 59. To estimate the cost of the entire system, the cost of the sub-system blocks is estimated and then rolled up into the top-level system cost. Typically, the cost is too high and there needs to be some effort to reduce the overall cost. So, trade studies are performed which focus on specific sub-system blocks. Of course, if any given block is modified, there will be an effect on other blocks known as secondary effects. For example, "better" performance using hardware which requires more power vs. "worse" performance using hardware which requires less power. A change such as that may have a secondary effect of increased copper thickness on other printed wiring boards which would then increase costs in other areas of the design. The secondary effects must be dealt with and considered but that would occur later in the process when a trade study is performed. Initially, there is the challenge of trying to determine which sub-block to apply focus to impact cost most significantly for the entire system, Figure 60. This is where the tools are significantly lacking. In the absence of sophisticated tools, the selection becomes somewhat arbitrary. It is desirable for the cost analyst to actively participate with the system designer to identify the areas of focus where the greatest impact to overall cost could be achieved.

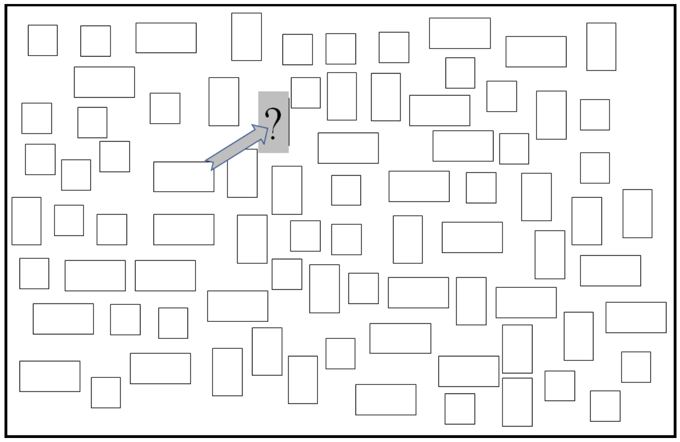


Figure 60 - Arbitrary Sub-Block Selection.

Used in conjunction with a COTS cost estimation tool, it should be possible to develop an algorithm to understand the system sub-blocks in terms of cost sensitivity to overall system cost. With such an algorithm a cost analyst could analyze a system and determine the relative cost sensitivity for each sub-block.

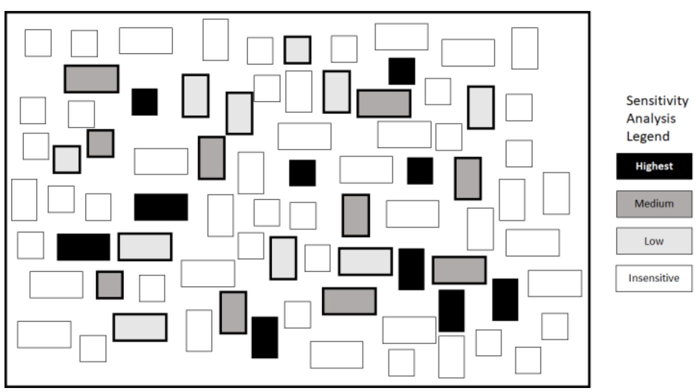


Figure 61 - Sensitivity Analysis Results.

Once the sub-system blocks have a relative sensitivity value, the sub-blocks could be ranked from most sensitive to least sensitive, Figure 61. Then using knowledge of the system, a cost analyst could suggest a few sub-blocks to focus attention for reasonable improvement goals, Figure 62.

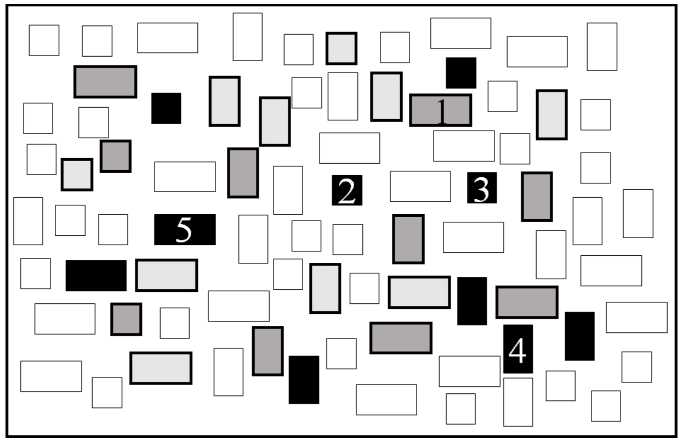


Figure 62 - Selection of Five Sub-Blocks.

Once a few sub-blocks have been selected for reasonable improvement goals, and using the COTS cost estimation tool, an estimate could be made to determine the impact of cost to the entire system from simultaneous improvements to these few selected sub-blocks, Figure 63.

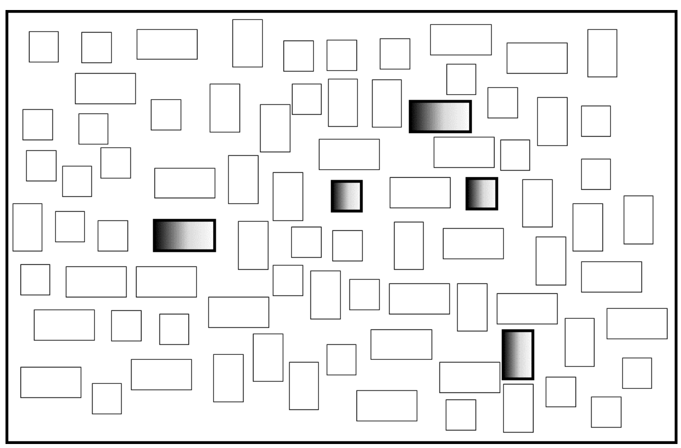


Figure 63 - Simultaneous Modification of Five Sub-Blocks.

The overall impact estimation would provide a bound, or maximum, for potential cost improvements. To realize any potential component cost improvements there would need to be some amount of investment of resources. Any investment up to the estimated maximum potential value would yield a profit. This then forms the basis for a Return On Investment (ROI).

Again, a full trade study would need to be performed to evaluate the potential impacts to the rest of the system. But those trade studies would be paid for using the ROI estimations.

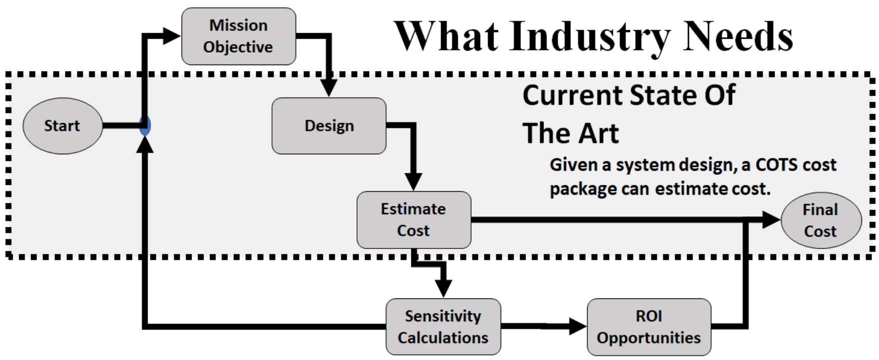


Figure 64 - Ideal Cost Estimation Process.

The limitation of the commercially available cost estimation tools is that it is essentially a unidirectional process. A user defines a system and uses the cost estimation package to estimate cost, Figure 64. What is missing is a bidirectional interaction with the software tool. By performing a sensitivity analysis upon the cost model, a cost analyst can offer suggestions to the system designer where to focus attention to most significantly impact overall system cost.

Summary

This paper presents an approach to generating a set of block diagrams for describing a standardized modular RADAR system applied to military applications in the aerospace industry. The resulting block diagrams were created using a compilation from a wide sample of available industry data and references integrated into a higher level, more generalized version. In addition to block diagrams, generalized block descriptions were also created along with a generalized numbering structure.

This paper demonstrates an approach to implementing the generalized block diagrams and numbering structure to create both a system model as well as a cost model for the RADAR under consideration. By means of the numbering structure, the system and cost models could be generated in such a way as to be modular to facilitate eventual trade studies for performance and cost improvements.

This paper discusses the potential advantages of a cost sensitivity algorithm applied upon the system cost model to analyze and determine which subsystem components in a chosen design solution have the highest sensitivity to overall cost. This paper illustrates that such an analysis could direct system designers to the areas of focus to most significantly impact the overall system cost early in the life cycle of a program.

Finally, the paper discusses using the sensitivity analysis results to select a few sub-blocks for reasonable improvement goals for simultaneous improvements. Simultaneous improvements to these few selected sub-blocks would provide a bound, or maximum, for potential system cost improvements. Any investment up to the estimated maximum potential value would yield a profit. This then forms the basis for a Return On Investment (ROI). The ROI estimate would in turn fund future trade studies.

Chapter 8 – A System Engineering Approach Using Sensitivity Analysis For Reducing System Cost

Modern software packages exist to estimate system cost early in the systems development and procurement process. The commercially available software which estimates system cost are limited in their ability to aid in system optimization towards multi-objective cost and performance goals, as many require a completed system design. To illustrate the benefits of cost optimization during early stages of design, this paper describes a sensitivity analysis applied to the design of an engineering system. This process seeks to use sensitivity analysis and a spiral design process to determine which cost drivers have the highest influence on overall system cost, and to realize high system performance while minimizing costs. This work is novel in that it describes a method and toolkit to enable simultaneous consideration of system costing with system engineering. This work is novel in that it demonstrates how to determine the cost sensitivities of components in a system, and how the sensitivity values can be used to suggest component parameter variations to maximize the impact to overall system cost. And finally, this work is novel in that it demonstrates a method using component cost sensitivity to determine the range of possible cost improvements to bound project return on investment.

Introduction

There are several very good commercially available cost estimation packages. To use these packages, first a system must be defined. The system must be defined in terms of hardware blocks. The hardware blocks can be arranged with a hierarchy such as a work breakdown structure (WBS). Once the system is defined the system can be entered into the cost estimation package. The package essentially converts each hardware component into a corresponding cost. In this way the cost of a system can be determined.

The limitation of the commercially available cost estimation packages is that it is essentially a unidirectional process. A user defines a system and uses the cost estimation package to estimate cost. The user can then experiment with alternatives or modifications to the system and estimate the corresponding associated system costs. What is missing is a bidirectional interaction with the software package. There is very little guidance from the cost estimation package which suggests to the user system modifications for consideration. It lacks suggestions to the designer which modifications would have the greatest impact to the overall cost of the system.

It is desirable to have a feature within a cost estimation package which can analyze the components of the system to determine which components have the greatest impact. In other words, which components have the highest sensitivity for modification as it pertains to the overall cost of the system.

Although sensitivity analysis is well understood the application of sensitivity analysis upon a cost model for the purposes of maximizing the impact to the overall system cost is novel. This paper explores an effort to generate a cost sensitivity algorithm of the various components in a system to analyze a system and determine which subsystem components in a chosen design solution have the highest sensitivity to cost for the overall system. In addition, the analysis highlights the areas to which a system designer could apply focus to reduce the overall system cost early in the life cycle of a Program.

The focus of this paper will be divided into five main topics: 1) an understanding of the current industry capabilities, 2) development of a cost sensitivity algorithm for application upon a system cost model including the development of sensitivity key size metrics (KSMs) for the component parameters for use with the cost sensitivity algorithm, 3) an example application of the full cost sensitivity algorithm with KSMs on a sample system cost model, 4) an example application of the full cost sensitivity algorithm with KSMs on a "real" system cost model, and 5) a brief discussion on return on investment (ROI) utilizing the results of the cost sensitivity algorithm.

In the first section, the tools available in industry are explained. Specifically, how cost packages are structured and designed for use. It will be explained that the available tools offer a user the ability to take a specific system architecture and estimate an expected cost for that system. There does not currently exist a tool which can offer a robust examination of the system and offer feedback to the user on how to improve or optimize the system architecture. It is that lack of feedback in the process which is addressed within this paper. Specifically, the use of a cost sensitivity algorithm to highlight areas of focus which can most significantly impact overall system cost.

In the second section, the development of a sensitivity algorithm is presented. It will be demonstrated that each parameter within a component could be considered either minor or impactful to overall system cost. The impactful parameter could then be varied (up or down) by some arbitrary amount to affect overall system cost (up or down). It is shown that the variation of the impactful parameter drives the estimated cost away from the baseline cost differently depending upon the component for which it applies. These differences constitute a cost sensitivity of a component parameter upon the overall system cost. These cost sensitivities can then be collected for each parameter and analyzed to determine the relative ranking of the cost sensitivity parameters. The highest-ranking cost sensitivity parameters are of particular importance to a system designer interested in optimizing a system architecture for cost versus performance trade studies.

A limitation was identified in the development of the cost sensitivity algorithm related to the factor by which the component parameters were varied to calculate cost sensitivity. Included is the discussion devoted to resolving the limitation of the usage of an arbitrary and uniform variation factor. Instead, KSMs were determined which allow for unique variation factors for each type of parameter.

In the third section, the fully developed cost sensitivity algorithm with KSMs was applied to an example cost model. The result of applying the algorithm is presented and demonstrates how the results can be applied to show potential cost improvements.

In the fourth section, the fully developed cost sensitivity algorithm with KSMs was applied to a "real" example cost model based on a generalized work breakdown structure (WBS) for a RADAR system applied to military applications in the aerospace industry. The result of applying the algorithm is presented and demonstrates the significant impact achievable to overall system cost when focus is applied appropriately to the areas for which overall system cost is most sensitive.

In the fifth section, the algorithm highlights the areas where trade studies could be performed and yields a target return on investment (ROI) budget, or a limit of money to spend as investment to achieve those cost improvements.

This work is novel in that it describes a method and toolkit to enable simultaneous consideration of system costing with system engineering. This study demonstrates how to determine the cost sensitivities of components in a system, and how the sensitivity values can be used to suggest component parameter variations to maximize the impact to overall system cost. And finally, this work is novel in that it demonstrates a method using component cost sensitivity to determine the range of possible cost improvements to bound project return on investment.

Related Work

One of the issues which complicates predicting the cost for any system is the concept of size. Systems which might have the same general function could vary significantly in terms of cost. For example, the engine in a supercar is more expensive than the engine in a commuter car even though they are both engines and have the same general function. It then becomes an exercise to understand what about those systems yield such significantly diverse costs. This concept is referred to as size. The application of sizing is not limited to a system. Normalization could be applied to variations in agricultural costs to normalize prices **[32]**. Again, this is the concept of a size with anomalies due to seasonal variations which must be normalized to determine standardized pricing. In a more generalized approach to normalizing cost data, the data could first be divided into broad categories. For example, the data could be grouped into three categories: normalizing for content, normalizing for quantity, and normalizing for inflation **[33]**. But again, the concept is the same. There is a base normalized size with factors which complicate the data. The International Cost Estimation and Analysis Association (ICEAA) offers instructional courses in cost estimation. The module "Data Collection and Normalization" **[34]** is devoted to this topic. Here cost can be normalized in one of three categories: Cost Units, Quantities, and Sizing Units. And for Sizing Units, the subcategories include weight, density or volume, and for software, Lines Of Code (LOC).

The variables that comprise a system may also influence the cost of the system. These variables are referred to as cost drivers. "Cost drivers are the structural determinants of the cost of an activity, reflecting any linkages or interrelationships that affect it" **[35]**. Of course, these definitions do not make a distinction between cost drivers which are impactful verses non-impactful. While any variables can contribute towards cost, each will have different sensitivities associated with them.

Related to the topic of cost drivers to a system is the concept of a trade study. A trade study is defined as "the activity of a multidisciplinary team to identify the most balanced technical solutions among a set of proposed viable solutions" **[36]**. This is a generalized definition but its applicability toward cost drivers applies. A trade study can be further defined where a "trade study is a formal tool that supports decision making" **[37]**. And in this reference, it notably applies to "realistic alternatives" and includes objects such as "performance" and "cost." However, the reference fails to adequately offer a solution as to how a cost trade study may be conducted. A system trade study can be performed using a standardized approach **[38]** based on the more generalized 'Standard Approach to Trade Studies' **[39]**. This more focused approach offers "significant developments" in cost benefits resulting from the trade study. The referenced author offers Cost As an Independent Variable (CAIV) and draws a connection between system cost and the results of the system trade.

However, in this reference the topic of "cost" and "risk" are deliberately removed from the "tradable criteria list". Whereas for a typical trade study, cost and risk are two of the main criteria for decision making. The reference is a typical approach for bidding where the system performance is defined early, and cost is included towards the end of the evaluation. To "converge on recommendations that are robust in the presence of uncertainty" a framework for a standardized trade study may be employed **[40]**. However, although the article does devote much attention to the area of cost, it does not offer any insight into cost as a variable with which to optimize a solution.

When considering many variables, or categories of variables, the topic of Multi Criteria Decision Analysis (MCDA) appears relevant. MCDA is used "as an umbrella term to describe a collection of formal approaches which seek to take explicit account of multiple criteria in helping individuals or groups explore decisions that matter" **[41]**. The author describes a structured approach to decision making which serves to help "decision makers to understand and to define their preferences, rather than descriptive, describing what they do and seeking simply to elicit their preferences." However, the article does not include a use case where discreet variables rather than stakeholders are considered.

The same observation is applicable for several works regarding MCDA. Considerations such as "economic, social, and environmental criteria are nowadays involved in practically all decision situations" **[42]**. The referenced author indicates the "decision process should naturally explore the conflicting nature of the criteria, the corresponding tradeoffs, the goals set by the decision makers, and of course the way that these can be introduced in an appropriate decision model that takes into account the subjectivity of the decision process and the preferences of the decision makers." The author describes "discrete problems." But by "discrete problems" the author refers "to decision situations involving the evaluation of a finite set of alternatives and actions over a predefined set of evaluation criteria." To illustrate, the author considers "a company or a public institution, where a manager and/or a group of people are confronted with a decision situation or "problem" that requires them to make a decision." Although the article is devoted to decision making it approaches decision making from a more global view with a wide variety of contributing factors and does not offer any specific insight into the cost variable.

A more refined approach to decision making is developed in the Analytic Hierarchy Process (AHP) and the Analytic Network Process (ANP). Both processes have criteria and each criteria have a value and a weight. By combining value and weight for each criteria a net result can be obtained which indicates the preferable decision choice. In the case of AHP **[43]** the decision is structured into a hierarchy with a goal, decision criteria, and alternatives. The Analytic Network Process (ANP) **[44]** is a more general form of AHP where the decision is structured as a network. For multicriteria analysis including AHP & ANP each criteria is associated with a weight of importance. Importance in this context is a relative measure between various criteria. By contrast each criteria may also have a degree of criticality. By critical, we mean the degree to which a change in that criteria's weight affects the final decision. It is possible that a criteria with a small weight in importance may be more critical to the final decision. This is the concept of sensitivity.

Although the topic of sensitivity analysis has been explored, the application of sensitivity analysis upon a system's cost is limited. In "An Introduction to Sensitivity Analysis" **[45]** the referenced author offers an introduction to sensitivity analysis using a series of papers on the subject. The article relies on the STELLA software which is an application for system modeling. The author writes that "Sensitivity analysis is used to determine how "sensitive" a model is to changes in the value of the parameters of the model and to changes in the structure of the model." The article describes exploratory exercises where the function of the system is described, and the sensitivities of the various system elements are considered. However, the techniques described do not mention cost as a variable. An alternative approach where system sensitivities are reduced to limit the effect upon variations in process parameters may be employed **[46]**. This varies from the concept of normalization in that normalization removes variations such that meaningful comparisons could be made. Whereas in this reference the variations are removed to dampen the effects for enhanced performance as in the case of a control system. A "process of recalculating outcomes under alternative assumptions to determine the impact of a variable under sensitivity analysis can be useful for a range of purposes" **[47]**. The reference specifically mentions both an "increased understanding of the relationships between input and output variables in a system or model" as well as "enhancing communication from modelers to decision makers." These purposes could be applied upon a system to determine cost sensitivities of the various elements to allow a system designer to understand the relationships and make informed decisions.

An interesting study of a production inventory system made a case for the application of sensitivity analysis to bring the "model solutions closer to the complexities of real systems" **[48]**. The referenced author makes the case that in the absence of sensitivity analysis the designer's predictions rely on history and assumes the same trend. While this may be a good assumption, by using sensitivity analysis it becomes a more predictive method, not solely based on history but with some understanding of the sensitivity of the variables. Cost was a secondary consideration and limited by first understanding the inventory levels and then calculating the corresponding costs. A more impactful result might demonstrate, for example, that cost could be significantly reduced if production runs were modified in a quantifiable way, and therefore inventory would have to be modified accordingly.

Sensitivity analysis can be applied in "many fields such as environmental risk assessment, behavior of agronomic systems, structural reliability or operational safety" **[49]**. The author explains that even "an environmental impact problem may be framed through the lenses of economics, and presented as a cost benefit or risk analysis, while the issue has little to do with costs or benefits or risks and a lot to do with profits, controls, and norms." The referenced author mentions sensitivity analysis "provides users of mathematical and simulation models with tools to appreciate the dependency of the model output from model input, and to investigate how important is each model input in determining its output." Although the referenced author explains sensitivity analysis can be used to investigate the dependency of output to input variables there is no significant example demonstrating a dependency with regards to cost.

The study on the topic of ownership of an electric transportation system in Swedish medium sized cities aimed "to emphasize on sensitivity analysis for the total cost of ownership (TCO) to reduce uncertainty by identifying which factors of interest that most likely cause the estimated cost values for the electric bus" **[50]**. The study does help to illustrate how versatile is sensitivity analysis and that it can be used to address cost in a wide variety of applications. However, the study focused on infrastructure as a system where the term "system" is very broad in its application. The study is diminished in that it starts with the assumption that electric alternatives reduce greenhouse gases but neglects to address the contributions of pollution created in the generation of electric power. The study upon marine renewable energy uses sensitivity analysis for cost reduction **[51]**. The analysis which largely focuses on cost "highlights the sensitivity of marine energy to three key parameters: the capital cost of first devices, the level of deployment before sustained cost reduction emerges, and the average rate of cost reduction with deployment (learning rate)." In this case the analysis focuses more on the different phases in the lifecycle of the system rather than on the elements of a system. While both include cost and sensitivity neither reference considers a system as a collection of components as in the case of a RADAR system.

There are numerous available Commercial Off The Shelf (COTS) cost modeling tools. In addition to providing the cost for an existing design, the PRICE Cost Analytics tool offers the system designer an ability to translate "needs" into "requirements" **[52]**. As noted for Design-to-Cost Targets, "It is widely accepted that 80% to 90% of cost is determined at the design or development stage." This highlights the need to perform trade studies early to optimize a solution before the design architecture has been defined. The Constructive Systems Engineering Cost

Model (COSYSMO) developed at MIT is an industry standard "to estimate the Systems Engineering effort for large-scale systems (both software and hardware)" **[53]**. Unfortunately, the tool focuses on the cost associated with the system engineering aspects of the design rather than a more comprehensive estimation of all disciplines associated with development or production costs. SEER by Galorath offers a tool for a system engineer to estimate a system cost once the system has been defined **[54]**. And the tool offers some rudimentary features regarding sensitivity. By varying the parameter inputs to the cost model, the user can observe the effect on overall system cost. But to utilize this ability to determine cost sensitivities for every parameter is manual and labor intensive. What is missing from all these COTS packages is the ability to directly calculate a cost sensitivity of each component in the system design to direct a system designer towards a cost optimized solution. This is not to imply that individual system component costs directly translate into the total cost of the system. Alternative selections of components would likely require some wider consideration of components which would necessarily have some cost increases and decreases. Instead, this method is a practical solution for a real problem. This work is intended to provide a designer a "compass" on where to focus attention by identifying the components with the highest cost sensitivities. This work is novel in that it describes a method and toolkit to enable simultaneous consideration of system costing with system engineering. This work is novel in that it demonstrates how to determine the cost sensitivities of components in a system, and how the sensitivity values can be used to suggest component parameter variations to maximize the impact to overall system cost. And finally, this work is novel in that it demonstrates a method using component cost sensitivity to determine the range of possible cost improvements to bound project return on investment.

Sensitivity Algorithm Development

Contrary to other efforts where cost has been normalized out of the equation or removed altogether this paper directs the effort toward the beginning of the design life cycle to optimize a solution before the final architecture has been selected.

A sensitivity algorithm was applied in several steps. First, a sample system cost model was identified such that the algorithm could be tested upon that sample. Next, the uncertainty parameters were determined. Then, the range of variation was determined. And finally, the results were calculated.

Selection Of A System Cost Model

The intent here is not to create a cost estimation tool. Instead, what is presented is a practical solution for a real problem used in conjunction with a COTS tool. It is assumed the math behind the analysis performed by the COTS tool forms a sufficient foundation upon which to develop a sensitivity algorithm. The outcome of this work is intended to provide a designer a "compass" on where to focus attention by identifying the components with the highest cost sensitivities. For purposes of creating a sensitivity algorithm for use with a cost estimation tool it is necessary to have a cost model upon which to apply an algorithm. There does exist a detailed and generic WBS structure which was developed for a RADAR system applied to military applications in the aerospace industry **[7]**. However, in the early stages of algorithm development it is sufficient to use a sample cost model. Most commercially available cost estimation packages come with a library of sample cost models. These sample cost models were surveyed to find an example which was complex enough to contain a significantly large WBS structure to allow for analysis while at the same time was not so large as to inhibit the process of algorithm development.

A cost model provided by SEER Galorath was identified and can be seen in Table 7. The WBS is indented down to four levels. The WBS includes both analog and digital subsystem blocks (e.g., Receiver Module, Digital Processing) as well as structural components (e.g., Receiver Chassis). This sample cost model was determined to provide for a significantly large enough WBS structure to allow for analysis. Also, the quantity of components, in this case nineteen, should provide for a significant quantity of component parameters with which to experiment.

Table 7 - Sample Cost Model WBS Structure

WBS Number	Component
1	NewGen Listening Station
1.1	Equipment Configuration
1.1.1	Receiver Module
1.1.1.1	Receiver
1.1.1.2	RF Module
1.1.1.3	RF Machined Housing
1.1.1.4	Receiver Chassis
1.1.2	Digital Processing
1.1.2.1	Converter & Noise Reduction
1.1.2.2	Data Processing
1.1.2.3	Purchased Memory
1.1.2.4	Interconnect – Data Bus
1.1.2.5	Instrumentation Panel
1.1.2.6	Digital Processing Chassis
1.1.2.7	Controller Software
1.1.3	Misc. Equipment
1.1.3.1	Wire Interconnects
1.1.3.2	Purchased Racks
1.1.3.3	Purchased Power Supply
1.2	Operational and Support Sites
1.2.1	Northeast Auxiliary
1.2.2	Atlantic Operations Center

1.2.3	Western Operations Center
1.2.4	Midwest Repairs
1.2.5	Express Repairs

Definition Of Terms

To standardize terminology some definitions are presented. The definitions are for the current paper and all efforts were made to adhere to conventional industry definitions. The following sections will elaborate considerably and provide context.

1. System – The System refers to the highest level WBS item.

2. Subsystem block – The Subsystem block refers to the highest-level hardware within the WBS structure below the System. As in Table 7, examples are 1.1.1. Receiver Module or 1.1.2. Digital Processing.

3. Component – The Component refers to the lowest level hardware within the WBS structure. As in Table 7, examples are 1.1.1.1. Receiver, 1.1.1.2. RF Module, etc.

4. Parameter – A Parameter in this context is a variable associated with a Component. An example may be Total Printed Circuit Boards (n) or Circuit Board Size (in2).

5. Impactful Parameter – In the more general usage a key cost driver impacts more significantly overall cost than other factors. In this context it was necessary to find Parameters which more so than others affect the overall System cost. Although a Component may have many Parameters, only a small subset of those could be considered an Impactful Parameter based on its effect upon cost.

6. Variation Factor – When an Impactful Parameter value is changed by a fixed percentage, the percentage by which it is changed is the Variation Factor.

Identification Of Minor vs. Impactful Parameters

Sensitivity analysis is used to determine the relationships between independent variables and dependent variables under certain conditions. In this case, it is desirable to determine the effect of changes to parameter values (input) on the overall cost of a system (output).

The sensitivity analysis method consists of three steps. First, the uncertainty parameters are determined. Second, the range of variation is determined. And third, the results are calculated **[55]**. In this case, the uncertainty parameters are the parameters included in the cost model, the range of the parameters will be a fixed percentage variation which is initially arbitrarily assigned, and then the effect of modifying the parameters will be observed as the output of the cost model. The development of the process is described in detail along with the results.

The first step for the sensitivity analysis is to determine the model parameters which apply for this sensitivity analysis. In general, the parameters for a component are all the same if the components are similar. An analog amplifier and an analog filter may be similar and have the same parameters while a chassis would have very dissimilar parameters.

It is understood that all the parameters of a component contribute to cost in some way. However, not all parameters contribute equally. In some cases, the impact to overall system cost may be quite negligible. Prior to automating the process, it can be quite prohibitive to utilize every parameter indiscriminately. However, if the set of parameters could be limited then achieving meaningful results with a manual method becomes practical. It therefore becomes necessary to determine which parameters could be considered Minor Parameters versus Impactful Parameters.

The available cost parameters were screened for their effect on overall system cost. The top six contributors in the case of an electrical component and the top three contributors in the case of a mechanical component were selected. These selected parameters were identified as Impactful Parameters as opposed to Minor Parameters.

The entries in Table 8 indicate the results of the parameter survey. The parameters fall under either one of two categories: Electrical Impactful Parameters or Mechanical Impactful Parameters.

Table 8 - Parameter Survey Results.

EE Impactful Parameters	ME Impactful Parameters
Total CCA	Weight
PCB Size	Volume
Discreet Components per PCB	
Integrated Components per PCB	
Clock Speed	

The second step in a sensitivity analysis is to determine the range by which the parameters are varied. In this case, instead of assigning a range of values, an initial fixed variation factor was selected to yield a sufficient spread in the results to demonstrate the method and begin to draw some conclusions. An initial fixed variation factor was arbitrarily assigned as 20%. A table was created which shows all the components with their respective parameters and their variation factors (Table 9 column F).

Table 9 – Full Data Set Using 20% Variation Factor.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1		Hardware		Parameters					Calculated System Cost				Interpretation of Cost Data						
2	Count	Sub System Block	Component	Parameter	Prmtr Value	Vary Amt	Varied "Up" Prmtr Value	Varied "Down" Prmtr Value	blank	Baseline Cost	Varied "up" cost	Varied "Down" cost	blank	delta "up" cost	delta "down" cost	delta "mid"	delta "range"	Rank of impact	Color Rank
3		Receiver Module																	
4			Receiver																
5	1		Receiver	Total CCAs	2	20%	2.4	1.6		8464002	8635812	8292434		171810	171568	171689	121	3	RED
6	2		Receiver	PCB Size	30	20%	36	24		8464002	8464234	8463702		231	300	266	34	30	GRN
7	3		Receiver	Discreet Components per PCB	40	20%	48	32		8464002	8465383	8460653		1381	3350	2365	984	23	GRN
8	4		Receiver	Integrated Components per PCB	45	20%	54	36		8464002	8554945	8355278		90943	108724	99834	8891	6	RED
9	5		Receiver	Clock Speed	240	20%	288	192		8464002	8470485	8454122		6483	9880	8181	1699	16	YEL
10			RF Module							8464002								39	
11	6		RF Module	Total CCAs	0.5	20%	0.6	0.4		8464002	8532808	8368659		68806	95343	82074	13269	7	RED
12	7		RF Module	PCB Size	30	20%	36	24		8464002	8465564	8462440		1562	1562	1562	0	25	GRN
13	8		RF Module	Discreet Components per PCB	6	20%	7	5		8464002	8464373	8463262		370	740	555	185	27	GRN
14	9		RF Module	Integrated Components per PCB	4	20%	5	3		8464002	8480703	8446047		16700	17955	17328	627	13	YEL
15	10		RF Module	Clock Speed	800	20%	960	640		8464002	8471924	8457594		7922	6408	7165	757	17	YEL
16			RF Machined Housing							8464002								40	
17	11		RF Machined Housing	Weight	3.5	20%	4.2	2.8		8464002	8482248	8445424		18245	18578	18412	166	12	YEL
18	12		RF Machined Housing	Volume	0.4	20%	0.48	0.32		8464002	8464044	8463938		42	64	53	11	34	
19			Rcv Chassis							8464002								41	
20	13		Rcv Chassis	Weight	15	20%	18	12		8464002	8477620	8439277		13618	24725	19171	5554	11	YEL
21	14		Rcv Chassis	Volume	4	20%	4.8	3.2		8464002	8464007	8463998		4	4	4	0	37	
22		Digitial Processing								8464002								42	
23			Converter & Noise Reduction							8464002								43	
24	15		Converter & Noise Reduction	Total CCAs	3	20%	3.6	2.4		8464002	8906581	8338728		442579	125275	283927	158652	2	RED
25	16		Converter & Noise Reduction	PCB Size	30	20%	36	24		8464002	8464444	8463561		442	442	442	0	28	GRN
26	17		Converter & Noise Reduction	Discreet Components per PCB	35	20%	42	28		8464002	8470804	8462064		6802	1939	4370	2432	20	YEL
27	18		Converter & Noise Reduction	Integrated Components per PCB	120	20%	144	96		8464002	8598094	8303686		134091	160316	147204	13112	4	RED
28	19		Converter & Noise Reduction	Clock Speed	20	20%	24	16		8464002	8465109	8442527		1107	21475	11291	10184	14	YEL
29			Data Processing							8464002								44	
30	20		Data Processing	Total CCAs	2	20%	2.4	1.6		8464002	8669292	7967018		205289	496985	351137	145848	1	RED
31	21		Data Processing	PCB Size	30	20%	36	24		8464002	8464402	8463640		400	362	381	19	29	GRN
32	22		Data Processing	Discreet Components per PCB	250	20%	300	200		8464002	8466173	8460964		2171	3038	2604	434	22	GRN
33	23		Data Processing	Integrated Components per PCB	750	20%	900	600		8464002	8601269	8377674		137267	86328	111798	25469	5	RED
34	24		Data Processing	Clock Speed	20	20%	24	16		8464002	8474208	8451898		10206	12104	11155	949	15	YEL
35			Purchased Memory							8464002								45	
36	25		Purchased Memory	Total CCAs	1	20%	1.2	0.8		8464002	8465693	8462322		1691	1681	1686	5	24	GRN
37	26		Purchased Memory	PCB Size	30	20%	36	24		8464002	8464015	8463989		13	13	13	0	35	
38	27		Purchased Memory	Discreet Components per PCB	55	20%	66	44		8464002	8464195	8463720		193	282	237	44	31	
39	28		Purchased Memory	Integrated Components per PCB	43	20%	52	34		8464002	8465579	8462505		1577	1497	1537	40	26	GRN
40	29		Purchased Memory	Clock Speed	20	20%	24	16		8464002	8464015	8463783		13	219	116	103	32	
41			Interconnect - Data Bus							8464002								46	
42	30		Interconnect - Data Bus	Total CCAs	1	20%	1.2	0.8		8464002	8515076	8413228		51073	50774	50924	150	8	RED
43	31		Interconnect - Data Bus	PCB Size	30	20%	36	24		8464002	8464063	8463941		61	61	61	0	33	
44	32		Interconnect - Data Bus	Discreet Components per PCB	55	20%	66	44		8464002	8468355	8456537		4353	7465	5909	1556	19	YEL
45	33		Interconnect - Data Bus	Integrated Components per PCB	43	20%	52	34		8464002	8494482	8430804		30480	33198	31839	1359	9	RED
46	34		Interconnect - Data Bus	Clock Speed	20	20%	24	16		8464002	8467464	8460131		3462	3871	3667	204	21	GRN
47			Instrumentation Panel							8464002								47	
48	35		Instrumentation Panel	Weight	2.5	20%	3	2		8464002	8470998	8456819		6996	7183	7090	94	18	YEL
49	36		Instrumentation Panel	Volume	0.4	20%	0.48	0.32		8464002	8464002	8464002		0	0	0	0	38	
50			Digital Processing Chassis							8464002								48	
51	37		Digital Processing Chassis	Weight	18	20%	21.6	14.4		8464002	8481946	8437212		17944	26790	22367	4423	10	RED
52	38		Digital Processing Chassis	Volume	4	20%	4.8	3.2		8464002	8464007	8463997		5	5	5	0	36	

Variation Of A Single Parameter

For the third step of sensitivity analysis the results are calculated. And in this case, the modification of the cost model parameters will be observed and recorded to understand the effect on the output of the overall cost of the system. This third step is where much of the work occurs. The development of the process is described in detail along with the results. The process demonstrates the method first with a single parameter before then demonstrating an example with all Impactful Parameters.

1) Establishing Baseline Cost

To understand the effects of modifications to the cost model to overall system cost it becomes necessary to establish a baseline, or a baseline cost. In this case, a sample cost model was chosen from the library of existing cost models from the commercially available package.

The selected model had the WBS structure as it appears in Table 7. It can be seen there is a system, subsystem blocks and components. After the specific model was selected, the application was run to estimate the cost for the overall system.

For purposes of illustration the following discussion will be applied to a specific parameter within a specific component. In this case, the number of PCBs in the Receiver will be analyzed. This case appears in Table 9, row 5.

2) Vary Component Parameter Up 20% for "Up" Cost

The selected parameter is varied up by an amount of 20%. For example, the number of CCAs in the component is modified from 2 to 2.4 (Table 9 15, row 5, column G). It is understood that practically it is unrealizable to have 2.4 CCAs and that only whole integers are possible. However, the values are strictly theoretical and used to determine the sensitivity of a particular parameter. After the sensitivity factors have been determined the paper will suggest the selection of practical and realizable values, e.g., 1 vs 2 CCAs. The new parameter value is applied to the cost model and the modified overall system varied "up" cost is estimated (Table 9, row 5, column K). The varied "up" cost is then compared to the baseline cost to estimate a delta "up" cost value (Table 9, row 5, column N).

3) Vary Component Parameter Down 20% for "Down" Cost

The selected parameter is then varied down by an amount of 20%. For example, the number of CCAs in the component is modified from 2 to 1.6 (Table 9, row 5, column H). The new parameter value is applied to the cost model and the modified overall system varied "down" cost is estimated (Table 9, row 5, column L). The varied "down" cost is then compared to the baseline cost to estimate a delta "down" cost value (Table 9, row 5, column O).

4) Calculate the Average Delta and Range

Referring to Figure 65A, the varied "up" and varied "down" costs can be seen graphically with respect to the baseline cost. When a parameter is varied from its baseline value it has the effect of driving the overall system cost away from the baseline. To put the data in a useful format the absolute value of the two results is graphed, see Figure 65B. In this figure both deviations are illustrated as driving the cost positively away from the baseline cost. And it can be observed that the degree by which the two deviations drive the cost away from the baseline is not the same. It should be noted that if the parameter resides upon a linear portion of a cost curve, then these two deltas would be identical. The fact that they are not the same indicates there is some sort of non-linearity for the cost curve. For purposes of this analysis, it is unnecessary to fully understand the cost curve. It is the magnitude of each delta which is of particular importance to the current discussion.

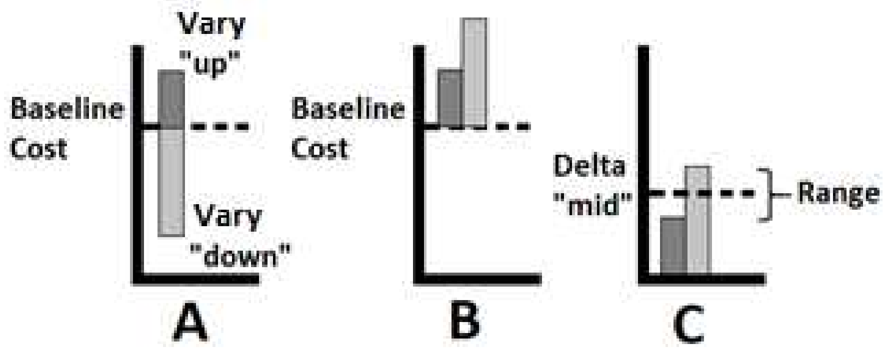


Figure 65 - Variation Of Cost

The two deltas are then normalized and averaged. The result is the delta "mid" value and the "up" and "down" deltas form the range, see Figure 65C. This gives a quantitative value for the sensitivity of the one parameter for a component upon the overall cost of the system. For this specific example, the numerical equivalent of Figure 65 appears in context in Table 9, row 5, columns K – O.

Expanded Analysis For Every Parameter

The ability to associate cost sensitivity to the various components of a system was previously, but briefly explored in an Excel based cost model **[2]**. The current paper parallels to a small degree the Excel effort where the cost model, which after cost calculations, had the ability to evaluate the sensitivity of various architectures associated with the design. The results were color coded by impact such that the user could identify where to apply focus to have the greatest effect on cost.

With the sensitivity algorithm established for a single parameter the next step is to apply the algorithm to every Impactful Parameter in the cost model and calculate all cost sensitivities. The full table of values appears in Table 9. The set of Impactful Parameters was chosen and appear in Table 9, column D. The varied amount, as discussed, was a uniform value of 20% (Table 9, column F). The spreadsheet calculates the varied "up" and varied "down" parameter values (Table

9, columns G & H). The cost tool was then run repeatedly and for each consecutive run only one parameter from the list was changed keeping all other values in their baseline condition. The overall system cost was then collected for each permutation of parameter value (Table 9, columns K & L). Using the results of each run from the cost tool the delta “mid” and delta “range” values were calculated for each parameter (Table 9, columns P & Q). With the delta “mid” values calculated the values could be ranked in order of greatest to least impact to overall system cost. The ranking appears in the table and is color coded with the ten most Impactful Parameters as Red, the next ten as Yellow, the next ten as Green, and the remainder as uncolored (Table 9, column R).

Interim Results – Uniform Variation Factor

The results are graphically illustrated in Figure 66. The various parameters have a wide variety of impact to overall system cost. This is as expected.

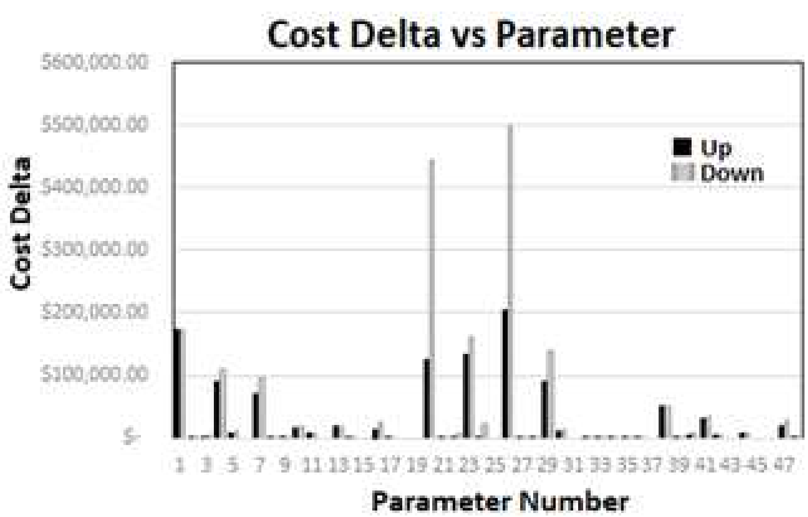


Figure 66 - Cost Delta vs. Parameter.

In addition, the results were sorted by parameter, see Table 10. It is very clear from Table 10 that for any parameter there exists a set of components upon which the parameter applies. And for each of the components there is clearly a difference in the sensitivity of that parameter depending upon to which component it is applied. In the case of Clock Speed, for example, the parameter is associated with six different components. The Converter and Noise Reduction component has the highest sensitivity for this parameter, and in addition, ranks as 14th most Impactful Parameter in sensitivity for the entire system.

Table 10 - Initial Sorted By Parameter

Parameter	Component	Sensitivity or Delta "mid"	Rank Of Impact
Clock Speed (USD/MHz)	Converter & Noise Reduction	$11.29K	14
	Data Processing	$11.15K	15
	Interconnect – Data Bus	$3.66K	21
	Purchased Memory	$115	32
	Receiver	$8.18K	16
	RF Module	$7.16K	17
Discreet Comp per PCB (USD/n)	Converter & Noise Reduction	$4.37K	20
	Data Processing	$2.60K	22
	Interconnect – Data Bus	$5.90K	19
	Purchased Memory	$237	31
	Receiver	$2.36K	23
	RF Module	$555	27
Integrated Comp per PCB (USD/n)	Converter & Noise Reduction	$147.20K	4
	Data Processing	$111.79K	5
	Interconnect – Data Bus	$31.83K	9
	Purchased Memory	$1.53K	26
	Receiver	$99.83K	6
	RF Module	$17.32K	13
Volume (USD/ft3)	Digital Processing Chassis	$5	36
	Instrumentation Panel	$0.09	38
	RCV Chassis	$4	37
	RF Machined Housing	$53	34
Weight (USD/lb.)	Digital Processing Chassis	$22.36K	10
	Instrumentation Panel	$7.08K	18
	RCV Chassis	$19.17K	11

	RF Machined Housing	$18.41K	12
Total CCAs (USD/n)	Converter & Noise Reduction	$283.92K	2
	Data Processing	$351.13K	1
	Interconnect – Data Bus	$50.92K	8
	Purchased Memory	$1.68K	24
	Receiver	$171.68K	3
	RF Module	$82.07K	7
PCB Size (USD/in2)	Converter & Noise Reduction	$441	28
	Data Processing	$380	29
	Interconnect – Data Bus	$61	33
	Purchased Memory	$13	35
	Receiver	$265	30
	RF Module	$1.56K	25

In Table 11 the maximums for each parameter are collected. In other words, for the parameter of Total CCAs it was determined in Table 10 that of the six components, Data Processing had the highest sensitivity and is in fact ranked as 1st overall. Therefore, in Table 11 for the parameter of Total CCAs, only the component Data Processing is listed with its corresponding sensitivity or delta "mid" value. The same logic applies for all the other parameters listed in Table 11.

Table 11 - Parameter Maximums

Parameter	Variation Factor	Component	Sensitivity or Delta "mid"
Total CCA	20%	Data Processing	$351.13K USD/n
Integrated Components per PCB	20%	Converter & Noise Reduction	$147.20K USD/n
Weight	20%	Digital Processing Chassis	$22.36K USD/lb.
Clock Speed	20%	Converter & Noise Reduction	$11.29K USD/MHz
Discreet Components per PCB	20%	Interconnect – Data Bus	$5.90K USD/n

PCB Size	20%	RF Module	$1.56K USD/in2
Volume	20%	RF Machined Housing	$189 USD/ft3

In addition, all cost scenarios were numbered in order of overall impact to system cost (Table 9, column R). Included are the corresponding components along with the parameter which influences the component costs. As mentioned, the ranking was identified grouped with red for the highest impact or sensitivity (1-10), yellow for medium (11-20) and green for low (21-30). It can be seen, for example, that the total number of CCAs in the Data Processing component has the highest cost sensitivity and was ranked correspondingly with a value of 1.

The results of the analysis are beginning to demonstrate some real-world implications. As a system designer this information is very useful. It indicates to a system designer which piece of hardware should receive focus to reduce cost to the overall system. In other words, for this system, the system designer should consider reducing the number of CCAs in the Data Processing component or increasing the number of Integrated Components per PCB in the Converter & Noise Reduction component, etc.

What is interesting to notice in Table 11 is that for two separate parameters (Integrated Components per PCB and Clock Speed), the Converter & Noise Reduction component had the biggest impact. This implies that if a system designer can only focus resources on one component, that component should be the Converter & Noise Reduction component since it clearly has the potential, with some improvements or modifications, to have the greatest impact to overall system cost. In practice simply swapping out a block and estimating cost gives a first order indication of the cost impact. It is not possible to swap out cost as modular blocks and estimate new costs without understanding that there are affects to the system. As design choices are made, impacts are assessed, costs can be estimated, new choices are made, and eventually the design spirals into a solution. As mentioned previously, this technique is intended as a tool for a cost analyst to provide a designer a “compass” on where to focus attention by identifying the components with the highest cost sensitivities.

Key Size Metric (KSM) Development

The data has been sorted by parameter, see Table 10. The first grouping represents all the occurrences where the numerical parameter for the Clock Speed was adjusted or varied. Varying the parameter had a different impact to overall cost depending upon which component contained that parameter. As can be seen in the table the component which had the biggest impact to overall cost when varying the parameter Clock Speed was the Converter & Noise Reduction component.

One significant issue to be addressed is that a uniform variation factor of 20% was used for every parameter. With some consideration it seems that using a uniform variation factor for every parameter is not sufficient and may yield misleading results. For example, consider the decision to vary total number of CCAs by the same factor as Clock Speed. In the case of the number of CCAs it may be reasonable, for example, to reduce the design from 3 to 2 CCAs while at the same time it may be possible to double the Clock Speed. Clearly it is not computationally sensible to consider the unit step size to be the same from one parameter to the next.

To overcome this limitation a set of Key Size Metrics (KSMs) must be developed. The KSMs would specify a unique value (other than a uniform 20%) for each parameter. In this way the relative impact to overall cost between parameters could be determined.

The first step is to establish a sorting of the parameters in order of impact. To do this some amount of engineering judgement and some familiarity with real systems and the associated cost is required. As an example, consider the parameter Discreet Components per PCB. With some effort it may be possible to combine various discreet parts together and in so doing reduce the quantity. This of course would have some impact on overall system cost. If enough of these improvements could be achieved, then it may be possible to reduce the size of the PCB. In this way it can be considered that the parameters should be arranged in a hierarchy of impact.

In addition, consider the possibility to reduce the count of CCAs. With some effort it may be possible to reduce the count of CCAs. But in a practical sense it does not ordinarily occur where a majority of the CCAs could be eliminated. Instead, it is a slight reduction in count. However, the impact of that reduction is typically very significant. By comparison, altering the Clock Speed may be significant in value (doubling the clock speed, for example) and may have an impact to cost as well. Typically, the impact is considerably less than reducing the CCA count. With considerations such as these in mind the parameters were sorted, and the results appear in order in Table 12.

Table 12 - Key Size Metric (KSM) Values.

Expected Parameter Sequence	KSM To Yield The Expected Sequence
Total CCA	20%
Weight	60%
Integrated Components per PCB	5%
Clock Speed	20%
PCB Size	40%
Discreet Components per PCB	10%
Volume	75%

The set of maximum parameters listed in Table 11 were used. These entries represent the cases for the components with the greatest impact to overall system cost. The goal was to adjust the variation factor for these few cases such that the resulting sequence of impact would match that as indicated in Table 12. Observing the effect on the overall cost impact the variation factor was varied for each of these parameters. Eventually, through a rigorous method of trial, error, and extrapolation, KSM values indicated in Table 12 were determined.

Using these KSM values for every parameter within the cost model the revised estimated overall system cost should yield results in this same sequence. The next step was to repeat the analysis of the previous sections, obtain another full set of data and analyze the results. The analysis should yield results in the same sequence as that indicated in Table 12.

Example: Sensitivity Algorithm Applied To Sample Cost Model

With the sensitivity algorithm established and with a suitable set of KSMs derived to vary each parameter with a unique value the previous effort was repeated.

The same system cost model was used as in the previous effort. The full table of values appears in Table 13. Because of the same subsystem blocks, the same list of applicable Impactful Parameters was chosen (Table 13, column D). The baseline values for each parameter remain unchanged (Table 13, column E). The new KSMs are applied (Table 13, column F). As before the spread sheet calculates the varied “up” and varied “down” values for the parameters (Table 13, columns G & H). The cost tool was then run repeatedly and for each consecutive run only one parameter from the list was changed keeping all other values in their baseline condition. The overall system cost was then collected for each permutation of parameter value (Table 13, columns K & L). Using the results of each run from the cost tool the delta “mid” and delta “range” values were calculated for each parameter (Table 13, columns P & Q). With the delta “mid” values calculated the values could be ranked in order of greatest to least impact to overall system cost. The ranking appears in the table and is color coded, as before, with the ten most Impactful Parameters as red, the next ten as yellow, the next ten as green, and the remainder as uncolored (Table 13, column R).

Table 13 – Full Data Set Using KSM Variation Factors.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1		**Hardware**		**Parameters**						**Calculated System Cost**				**Interpretation of Cost Data**					
2	Count	Sub System Block	Component	Parameter	Prmtr Value	Vary Amt (KSM)	Varied "Up" Prmtr Value	Varied "Down" Prmtr Value	blank	Baseline Cost	Varied "up" cost	Varied "Down" cost	blank	delta "up" cost	delta "down" cost	delta "mid"	delta "range"	Rank of impact	Color Rank
3		Receiver Module																	
4			Receiver																
5	1		Receiver	Total CCAs	2	20%	2.4	1.6		8464002	8635812	8292434		171810	171568	171689	121	3	RED
6	2		Receiver	PCB Size	30	65%	49.5	10.5		8464002	8464754	8463050		752	952	852	100	26	GRN
7	3		Receiver	Discreet Components per PCB	40	5%	42	38		8464002	8463608	8462425		-394	1577	591	986	28	GRN
8	4		Receiver	Integrated Components per PCB	45	5%	48	42		8464002	8513752	8436029		49750	27973	38862	10888	9	RED
9	5		Receiver	Clock Speed	240	20%	288	192		8464002	8470485	8454122		6483	9880	8181	1699	17	YEL
10			RF Module							8464002								39	
11	6		RF Module	Total CCAs	0.5	20%	0.6	0.4		8464002	8532808	8368659		68806	95343	82074	13269	4	RED
12	7		RF Module	PCB Size	30	65%	49.5	10.5		8464002	8469078	8458926		5076	5077	5076	0	19	YEL
13	8		RF Module	Discreet Components per PCB	6	5%	7	5		8464002	8464373	8463262		370	740	555	185	29	GRN
14	9		RF Module	Integrated Components per PCB	4	5%	5	3		8464002	8480703	8446047		16700	17955	17328	627	13	YEL
15	10		RF Module	Clock Speed	800	20%	960	640		8464002	8471924	8457594		7922	6408	7165	757	18	YEL
16			RF Machined Housing							8464002								40	
17	11		RF Machined Housing	Weight	3.5	60%	5.6	1.4		8464002	8517543	8423335		53541	40667	47104	6437	8	RED
18	12		RF Machined Housing	Volume	0.4	75%	0.7	0.1		8464002	8464120	8463741		118	261	189	72	31	
19			Rcv Chassis							8464002								41	
20	13		Rcv Chassis	Weight	15	60%	24	6		8464002	8514311	8397695		50309	66308	58308	7999	6	RED
21	14		Rcv Chassis	Volume	4	75%	7	1		8464002	8464018	8463986		16	16	16	0	37	
22		Digitial Processing								8464002								42	
23			Converter & Noise Reduction							8464002								43	
24	15		Converter & Noise Reduction	Total CCAs	3	20%	3.6	2.4		8464002	8906581	8338728		442579	125275	283927	158652	2	RED
25	16		Converter & Noise Reduction	PCB Size	30	65%	49.5	10.5		8464002	8465683	8462709		1681	1293	1487	194	24	GRN
26	17		Converter & Noise Reduction	Discreet Components per PCB	35	5%	37	33		8464002	8464556	8463448		554	554	554	0	30	GRN
27	18		Converter & Noise Reduction	Integrated Components per PCB	120	5%	126	114		8464002	8542729	8470211		78727	-6209	36259	42468	10	RED
28	19		Converter & Noise Reduction	Clock Speed	20	20%	24	16		8464002	8465109	8442527		1107	21475	11291	10184	15	YEL
29			Data Processing							8464002								44	
30	20		Data Processing	Total CCAs	2	20%	2.4	1.6		8464002	8669292	7967018		205289	496985	351137	145848	1	RED
31	21		Data Processing	PCB Size	30	65%	49.5	10.5		8464002	8465301	8462880		1298	1122	1210	88	25	GRN
32	22		Data Processing	Discreet Components per PCB	250	5%	263	237		8464002	8464245	8462891		243	1111	677	434	27	GRN
33	23		Data Processing	Integrated Components per PCB	750	5%	788	712		8464002	8517175	8460534		53173	3468	28320	24853	12	YEL
34	24		Data Processing	Clock Speed	20	20%	24	16		8464002	8474208	8451898		10206	12104	11155	949	16	YEL
35			Purchased Memory							8464002								45	
36	25		Purchased Memory	Total CCAs	1	20%	1.2	0.8		8464002	8465693	8462322		1691	1681	1686	5	22	GRN
37	26		Purchased Memory	PCB Size	30	65%	49.5	10.5		8464002	8464045	8463970		43	32	37	6	35	
38	27		Purchased Memory	Discreet Components per PCB	55	5%	58	52		8464002	8464059	8463904		57	98	77	20	34	
39	28		Purchased Memory	Integrated Components per PCB	43	5%	52	34		8464002	8465579	8462505		1577	1497	1537	40	23	GRN
40	29		Purchased Memory	Clock Speed	20	20%	24	16		8464002	8464015	8463783		13	219	116	103	33	
41			Interconnect - Data Bus							8464002								46	
42	30		Interconnect - Data Bus	Total CCAs	1	20%	1.2	0.8		8464002	8515076	8413228		51073	50774	50924	150	7	RED
43	31		Interconnect - Data Bus	PCB Size	30	65%	49.5	10.5		8464002	8464200	8463855		198	147	173	25	32	
44	32		Interconnect - Data Bus	Discreet Components per PCB	55	5%	58	52		8464002	8465471	8461479		1469	2523	1996	527	21	GRN
45	33		Interconnect - Data Bus	Integrated Components per PCB	43	5%	52	34		8464002	8494482	8430804		30480	33198	31839	1359	11	YEL
46	34		Interconnect - Data Bus	Clock Speed	20	20%	24	16		8464002	8467464	8460131		3462	3871	3667	204	20	YEL
47			Instrumentation Panel							8464002								47	
48	35		Instrumentation Panel	Weight	2.5	60%	4	1		8464002	8484563	8456765		20561	7237	13899	6662	14	YEL
49	36		Instrumentation Panel	Volume	0.4	75%	0.7	0.1		8464002	8464003	8464002		0	0	0	0	38	
50			Digital Processing Chassis							8464002								48	
51	37		Digital Processing Chassis	Weight	18	60%	28.8	7.2		8464002	8524752	8388856		60750	75146	67948	7198	5	RED
52	38		Digital Processing Chassis	Volume	4	75%	7	1		8464002	8464021	8463984		18	18	18	0	36	

Results – Nonuniform Variation Factor (KSM)

The results were collected similarly as were done in Figure 66. Consistent with the earlier example the various parameters have a wide variety of impact to overall system cost.

The results were numbered by parameter (Table 13, column R). It is very clear from Table 13 that for any parameter there exists a set of components upon which the parameter applies. And for each of the components there is clearly a difference in the Sensitivity of that parameter depending upon to which component it is applied. In the case of Clock Speed, for example, the parameter is associated with six different components. The Converter and Noise Reduction component has the highest sensitivity for this parameter, and in addition, ranks as 15th most Impactful Parameter in sensitivity for the entire system.

In Table 14 the maximums for each parameter are collected. In other words, for the parameter of Total CCAs it was determined in Table 13that of the six components, Data Processing had the highest sensitivity and is in fact ranked as 1st overall. Therefore, in Table 14 for the parameter of Total CCAs only the component Data Processing is listed with its Corresponding sensitivity, or delta “mid” value. The same logic applies for all the other parameters listed in Table 14.

Table 14 - Parameter Maximums

Parameter	Variation Factor	Component	Sensitivity
Total CCA	20%	Data Processing	$351.13K USD/n
Weight	60%	Digital Processing Chassis	$67.94K USD/lb
Integrated Components per PCB	5%	Receiver	$38.86K USD/n
Clock Speed	20%	Converter & Noise Reduction	$11.29K USD/MHz
PCB Size	40%	RF Module	$5.07K USD/in2
Discreet Components per PCB	10%	Interconnect – Data Bus	$1.99K USD/n
Volume	75%	RF Machined Housing	$189 USD/ft3

In addition, all cost scenarios were numbered in order of overall impact to system cost (Table 13, column R). Included are the corresponding components along with the parameter which influences the component costs. As mentioned, the ranking was color coded with red for the highest impact or sensitivity (1-10), yellow for medium (11-20) and green for low (21-30). It can be seen, for example, that the total number of CCAs in the Data Processing component has the highest cost sensitivity and was ranked correspondingly with a value of 1.

Return On Investment (ROI): Sample Cost Model

While all the calculations and results presented thus far are of theoretical importance the value of this work lies in the application of the results. The question of primary concern relates to how a system architecture can be optimized in terms of performance and cost early in the life cycle of a program. To illustrate the significance of the results upon potential improvements to the sample system the following discussion is offered.

The information which appears Table 13, column R indicates the top 30 most Impactful Parameters. It is unrealistic to consider improvements to a system in such a broad number of parameters over a broad set of components. Instead, focus will be applied to a more conservative subset. For consideration, the top five most Impactful Parameters will be analyzed such that the potential improvements to the sample system can be determined. Table 15 lists the top six most Impactful Parameters.

Table 15 - Baseline/Try Parameter Values.

Component	Parameter	Rank Of Impact	Parameter Baseline	Parameter Try
Data Processing	Total CCAs	1	2	1
Converter & Noise Reduction	Total CCAs	2	3	2
Receiver	Total CCAs	3	2	1
RF Module	Total CCAs	4	0.5	0.5
Digital Processing Chassis	Weight	5	18	17
Rcv Chassis	Weight	6	15	14

The next step was to assign Baseline/Try values, Table 15. The column Parameter Baseline indicates the value of the parameter which was used in the baseline cost estimations. Observing Table 15, a system designer should understand that the total number of CCAs in the Data Processing component is the most sensitive parameter within the entire system and has the greatest impact to overall cost. Therefore, a system designer should focus resources at this location to optimize the system for performance vs. overall system cost. With some effort, as an example, it may be possible to combine parts such that CCA board space could be reduced and ultimately the need for an entire CCA might be eliminated. A reasonable reduction goal in this Parameter could be from 2 CCAs down to 1 CCA. The goal of 1 CCA, in this example, is listed under the column heading Parameter Try. In fact, this column contains a reasonable reduction in parameter value for five of the highest sensitivity parameters. The RF Module was eliminated from this exercise because it was unreasonable to reduce the parameter value below its baseline value.

Once the top five most Impactful Parameters were determined and reasonably achievable Parameter Try values were assigned the cost model could be run with ALL the potential improvements applied simultaneously and the corresponding impact to cost could be observed. It is important to mention that no longer are the KSMs involved in the calculation. KSMs were only used to create a variation factor to understand the sensitivity, it was a theoretical adjustment. In this exercise, real values are being explored.

The two scenarios in Table 16 compare the Baseline Cost with the Try Cost. The Try Cost includes all the Parameter values from the Parameter Try column of Table 15 applied simultaneously.

Table 16 - Baseline vs. Potential Try Cost.

	Value
Baseline System Cost	$8,464K
New System Cost (Try Cost)	$6,682K
Savings	$1,781K
% Improvement	21%

By modifying the top five cost driving parameters from a baseline value to an achievable and improved value it is demonstrated that there would be a significant improvement to overall system cost. The results of the two scenarios are summarized in Table 16.

As can be seen in Table 16, the percentage improvement is 21% over the baseline which is a significant impact! Another way to interpret this result is in terms of return on investment (ROI). To modify a parameter value, it would of course be necessary to expend some resources to achieve the new value. For example, to reduce a design from 3 CCAs to 2 CCAs some amount of resources, or investment, must be made. To perform some amount of research, design, analysis, or trade study, there must be some expended resource which yields a parameter improvement. A system designer should know the cost of that expended resource. In this case if the system designer remains below a $1.7M investment then the project overall would demonstrate an improvement. In other words, a system designer could spend up to $1.7M to achieve improvements in those top five parameters which most significantly impact overall system cost. And of course, anything less than $1.7M contributes to profit margin. If those achievements could be realized there would be a 21% improvement in overall system cost which is clearly a significant improvement.

Example: Sensitivity Algorithm Applied To "Real" Cost Model

With the cost sensitivity algorithm fully developed and understood, the effort now turns towards implementation of a "real" example. For the development of the cost sensitivity algorithm a sample cost model has been used. While this has led to a theoretical benefit, what remains to be seen is if this algorithm can be utilized in a more real-life example. To address and satisfy this question, a real example is available. In particular, a cost model based on the standardized WBS structure for an airborne RADAR system for a military aerospace application has been developed and can be utilized to test the cost sensitivity algorithm.

Selection Of A "Real" Cost Model

In A System Engineering Approach Using Sensitivity Analysis For Reducing System Cost, JACP April 2022, an effort was made to consolidate block diagrams from a wide sample of available examples. This was done to create a generalized block diagram of an airborne RADAR for military applications and where each of the examples could be considered a subset of the more generalized form. The resulting block diagrams and definitions were then organized into a WBS structure (Table 17).

Table 17 - Indentured System Numbering Structure.

Level 1	Level 2	Level 3	Block Name
001			Radar
	001.01		Antenna
		001.01.01	Radiator
		001.01.02	TR Product
		001.01.03	Duplexer
	001.02		Transmitter
		001.02.01	Power Amplifier
		001.02.02	Up Converter
		001.02.03	Local Oscillator
	001.03		Synchronizer
		001.03.01	Synchronizer
	001.04		Receiver
		001.04.01	Low Noise Amplifier
		001.04.02	Down Converter
		001.04.03	Local Oscillator
		001.04.04	IF Amplifier
		001.04.05	Filters
		001.04.06	2nd Down Converter
		001.04.07	2nd Local Oscillator
		001.04.08	Detector
		001.04.09	Analog to Digital Converter
	001.05		Processor
		001.05.01	Processor
	001.06		Power
		001.06.01	Transformer
		001.06.02	Rectifier
		001.06.03	Filter
		001.06.04	Regulator
	001.07		Display
		001.07.01	Video Amplifier
		001.07.02	Display

The generalized WBS structure was shown useful as a foundation for both a system model and a cost model. As can be seen, the suggested cost model (Table 18) is the same as the WBS in that it maintains the same structure however the cost model includes additional rows for "Roll Up." A cost tool could call out Level 2 hardware, for example an antenna. However, an antenna is also a collection of Level 3 hardware blocks. In this case, both options are included in the cost model. And when the model is run to produce an estimate either, but not both would be selected.

Table 18 - Indentured Cost Numbering Structure.

Level 1	Level 2	Level 3	Block Name
001			Radar
	001.01		Antenna
	001.01RU		Antenna Roll Up
		001.01.01	Radiator
		001.01.02	TR Product
		001.01.03	Duplexer
	001.02		Transmitter
	001.02RU		Transmitter Roll Up
		001.02.01	Power Amplifier
		001.02.02	Up Converter
		001.02.03	Local Oscillator
	001.03		Synchronizer
	001.03RU		Synchronizer Roll Up
		001.03.01	Synchronizer
	001.04		Receiver
	001.04RU		Receiver Roll Up
		001.04.01	Low Noise Amplifier
		001.04.02	Down Converter
		001.04.03	Local Oscillator
		001.04.04	IF Amplifier
		001.04.05	Filters
		001.04.06	2nd Down Converter
		001.04.07	2nd Local Oscillator
		001.04.08	Detector
		001.04.09	Analog to Digital Converter
	001.05		Processor
	001.05RU		Processor Roll Up
		001.05.01	Processor
	001.06		Power
	001.06RU		Power Roll Up
		001.06.01	Transformer
		001.06.02	Rectifier
		001.06.03	Filter
		001.06.04	Regulator
	001.07		Display
	001.07RU		Display Roll Up
		001.07.01	Video Amplifier
		001.07.02	Display

This new RADAR cost model represents a real-life example of a RADAR cost model upon which to verify the benefits of employing the cost sensitivity algorithm.

Impactful Parameters & KSM Values

The work of determining the Impactful Parameters has already been completed. The same set of Impactful Parameters which were previously used and appear in Table 11 will be used once again in this analysis. No additional work in this area is required.

The work of determining the KSM values has already been completed. The same set of KSM values which were previously used and appear in Table 12 will be used once again in this analysis. No additional work in this area is required.

Re-Run Of Algorithm & Data Collection

As before, an Excel file was created for the real-life cost model analysis which shows all the components with their respective parameters and their variation factors (Table 19). The table was again used to collect and organize the information including hardware components, Impactful Parameters, KSM values, etc. As before, the spread sheet calculates the vary "up" and vary "down" values for the parameters. To establish a baseline cost, the cost model was run to estimate the cost for the overall system and the results were compiled.

Table 19 – Full Data Set – Algorithm Using "Real" System Cost Model.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1		Hardware		Parameters					Calculated System Cost					Interpretation of Cost Data						
2	Count	Level 1	WBS	Level 2	Parameter	Prmtr Value	Vary Amt (KSM)	Varied "Up" Prmtr Value	Varied "Down" Prmtr Value	blank	Baseline Cost	Varied "up" cost	Varied "Down" cost	blank	delta "up" cost	delta "down" cost	delta "mid"	delta "range"	Rank of impact	Color Rank
3		RADAR	001.01	Antenna																
4	1	RADAR		Antenna	Total CCAs	2	20%	2.4	1.6		16975050	17192074	16759573		217024	215478	216251	773	6	RED
5	2	RADAR		Antenna	PCB Size	30	65%	49.5	10.5		16975050	16975050	16975050		0	0	0	0	29	GRN
6	3	RADAR		Antenna	Discreet Components per PCB	48	5%	51	45		16975050	16976386	16964376		1336	10674	6005	4669	26	GRN
7	4	RADAR		Antenna	Integrated Components per PCB	40	5%	42	38		16975050	17052589	16943311		77539	31739	54639	22900	16	YEL
8	5	RADAR		Antenna	Clock Speed	400	20%	480	320		16975050	16984776	16877596		9726	97454	53590	43864	17	YEL
9		RADAR	001.02	Transmitter							16975050								30	GRN
10	6	RADAR		Transmitter	Total CCAs	2	20%	2.4	1.6		16975050	17308905	16166897		333855	808153	571004	237149	3	RED
11	7	RADAR		Transmitter	PCB Size	30	65%	49.5	10.5		16975050	16975050	16975050		0	0	0	0	31	
12	8	RADAR		Transmitter	Discreet Components per PCB	55	5%	58	52		16975050	17004571	16972409		29521	2642	16081	13440	22	GRN
13	9	RADAR		Transmitter	Integrated Components per PCB	35	5%	37	33		16975050	17162777	16958067		187727	16983	102355	85372	12	YEL
14	10	RADAR		Transmitter	Clock Speed	400	20%	480	320		16975050	17003903	16664333		28852	310717	169785	140932	9	RED
15		RADAR	001.03	Synchronizer							16975050								30	GRN
16	11	RADAR		Synchronizer	Total CCAs	1	20%	1.2	0.8		16975050	17009736	16595557		34685	379493	207089	172404	7	RED
17	12	RADAR		Synchronizer	PCB Size	30	65%	49.5	10.5		16975050	16975050	16975050		0	0	0	0	33	
18	13	RADAR		Synchronizer	Discreet Components per PCB	40	5%	42	38		16975050	16975766	16967186		715	7864	4290	3574	27	GRN
19	14	RADAR		Synchronizer	Integrated Components per PCB	10	5%	11	9		16975050	16987219	16962910		12168	12140	12154	14	25	GRN
20	15	RADAR		Synchronizer	Clock Speed	2000	20%	2400	1600		16975050	16995861	16886362		20810	88688	54749	33939	15	YEL
21		RADAR	001.04	Receiver							16975050								30	GRN
22	16	RADAR		Receiver	Total CCAs	3	20%	3.6	2.4		16975050	17638334	15810925		663284	1164125	913705	250420	1	RED
23	17	RADAR		Receiver	PCB Size	20	65%	33	7		16975050	16975050	16975050		0	0	0	0	35	
24	18	RADAR		Receiver	Discreet Components per PCB	60	5%	63	57		16975050	16996058	16953025		21007	22025	21516	509	21	GRN
25	19	RADAR		Receiver	Integrated Components per PCB	45	5%	48	42		16975050	17205498	16845203		230448	129847	180147	50300	8	RED
26	20	RADAR		Receiver	Clock Speed	250	20%	300	200		16975050	17020973	16731077		45923	243973	144948	99025	10	RED
27		RADAR	001.05	Processor							16975050								30	GRN
28	21	RADAR		Processor	Total CCAs	2	20%	2.4	1.6		16975050	17310613	16164481		335562	810569	573066	237503	2	RED
29	22	RADAR		Processor	PCB Size	30	65%	49.5	10.5		16975050	16975050	16975050		0	0	0	0	37	
30	23	RADAR		Processor	Discreet Components per PCB	48	5%	51	45		16975050	16978503	16949246		3453	25804	14629	11176	23	GRN
31	24	RADAR		Processor	Integrated Components per PCB	40	5%	42	38		16975050	17043041	16946892		67990	28158	48074	19916	19	YEL
32	25	RADAR		Processor	Clock Speed	800	20%	960	640		16975050	17042420	16888982		67370	86068	76719	9349	13	YEL
33		RADAR	001.06	Power							16975050								30	GRN
34	26	RADAR		Power	Total CCAs	2	20%	2.4	1.6		16975050	17138839	16579579		163789	395471	279630	115841	5	RED
35	27	RADAR		Power	PCB Size	20	65%	33	7		16975050	16975050	16975050		0	0	0	0	39	
36	28	RADAR		Power	Discreet Components per PCB	80	5%	84	76		16975050	16991472	16964320		16422	10730	13576	2846	24	GRN
37	29	RADAR		Power	Integrated Components per PCB	30	5%	32	28		16975050	17109809	16962497		134759	12553	73656	61103	14	YEL
38	30	RADAR		Power	Clock Speed	50	20%	60	40		16975050	17005124	16933110		30074	41941	36007	5933	20	YEL
39		RADAR	001.07	Display							16975050								30	GRN
40	31	RADAR		Display	Total CCAs	4	20%	4.8	3.2		16975050	17615069	16563280		640018	411770	525894	114124	4	RED
41	32	RADAR		Display	PCB Size	20	65%	33	7		16975050	16975050	16975050		0	0	0	0	41	
42	33	RADAR		Display	Discreet Components per PCB	28	5%	30	26		16975050	16975570	16971410		520	3640	2080	1560	28	GRN
43	34	RADAR		Display	Integrated Components per PCB	35	5%	37	33		16975050	17175029	16956949		199979	18101	109040	90939	11	YEL
44	35	RADAR		Display	Clock Speed	100	20%	120	80		16975050	17006207	16901432		31157	73618	52387	21231	18	YEL

The cost tool was then run repeatedly, and for each consecutive run, only one parameter from the list was changed keeping all other values in their baseline condition. The overall system cost was then collected for each permutation of parameter value. Using the results of each run from the cost tool, the delta "mid" and delta "range" values were calculated for each parameter. With the delta "mid" values calculated, the values could be ranked in order of greatest to least impact to overall system cost. The ranking appears in the table and is color coded, as before, with the ten most Impactful Parameters as red, the next ten as yellow, the next ten as green, and the remainder as uncolored (Table 19).

Results – Algorithm On "Real" Cost Model

The results are graphically illustrated in Figure 67. As before, the various parameters have a wide variety of impact to overall system cost.

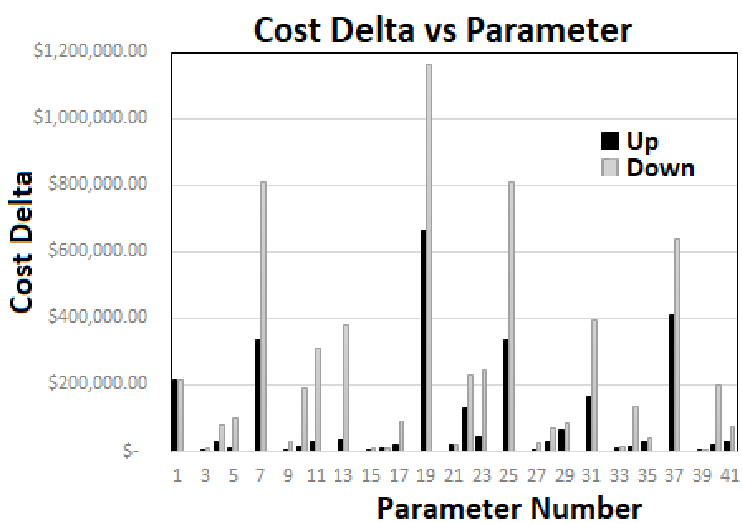


Figure 67 - Real-Life Cost Delta vs. Parameter

The results were sorted by parameter (Table 20). It is very clear from the table that for any parameter, there exists a set of components upon which the parameter applies. And, for each of the components, there is clearly a difference in the sensitivity of that parameter depending upon to which component it is applied. In the case of Clock Speed, for example, the parameter is associated with seven different components. The Receiver component has the highest sensitivity for this parameter, and in addition, ranks as 10th most Impactful Parameter in sensitivity for the entire system.

Table 20 - Real-Life Cost Model Results Sorted By Parameter.

Parameter	Component	Sensitivity or Delta "mid"	Rank Of Impact
Clock Speed (USD/MHz)	Antenna	$53.59K	17
	Display	$52.39K	18
	Power	$36K	20
	Processor	$76.72K	13
	Receiver	$144.95K	10
	Synchronizer	$54.75K	15
	Transmitter	$169.79K	9
Discreet Comp per PCB (USD/n)	Antenna	$6K	26
	Display	$2.08K	28
	Power	$13.58K	24
	Processor	$14.63K	23
	Receiver	$21.52K	21
	Synchronizer	$4.29K	27
	Transmitter	$16.08K	22
Integrated Comp per PCB (USD/n)	Antenna	$5.46K	16
	Display	$109.04K	11
	Power	$73.66K	14
	Processor	$48.07K	19
	Receiver	$180.15K	8
	Synchronizer	$12.15K	25
	Transmitter	$102.36K	12
PCB Size (USD/in2)	Antenna	0	29
	Display	0	41
	Power	0	39
	Processor	0	37
	Receiver	0	35
	Synchronizer	0	33
	Transmitter	0	31
Total CCAs (USD/n)	Antenna	$216.25K	6
	Display	$525.89K	4
	Power	$279.63K	5
	Processor	$573.07K	2
	Receiver	$913.70K	1

	Synchronizer	$207.09K	7
	Transmitter	$571.00K	3

In Table 21, the maximums for each parameter are collected. In other words, for the parameter of Total CCAs, it was determined in Table 20 that of the seven components, Receiver had the highest sensitivity and is in fact ranked as 1st overall. Therefore, in Table 21, for the parameter of Total CCAs, only the component Receiver is listed with its corresponding sensitivity, or delta "mid" value. The same logic applies for all the other parameters listed in Table 21.

Table 21 - Real-Life Parameter Maximums.

Parameter	Variation Factor	Component	Sensitivity or Delta "mid"
Total CCA	20%	Receiver	$913.71K USD/n
Integrated Components per PCB	5%	Receiver	$180.15K USD/n
Clock Speed	20%	Transmitter	$169.79K USD/MHz
Discreet Components per PCB	10%	Receiver	$21.52K USD/n
PCB Size	40%	Antenna	$0K USD/in2

In addition, all cost scenarios were numbered in order of overall impact to system cost (Table 19, column R). Included are the corresponding components along with the parameter which influences the component costs. As mentioned, the ranking was color coded with red for the highest impact or sensitivity (1-10), yellow for medium (11-20) and green for low (21-30). It can be seen, for example, that the total number of CCAs in the Receiver has the highest cost sensitivity and was ranked correspondingly with a value of 1.

Multivariable Analysis & Trade Studies

A robust structured system modelling approach utilizes the concept of modularity. If the system is comprised of modules, then the possibility exists where modules could be swapped to modify the system for various performance characteristics. At the same time, if the cost model mirrors the system model, then as the system is being defined, a rough cost estimation could be determined simultaneously.

Even with a modular approach, when designing a system more than one variable must be considered. Choices are made regarding those variables. In most cases, variable choices have competing impacts. For example, one design architecture may have "better" performance using more power vs. "worse" performance using less power. Decisions for a sub-system need to be evaluated at a system level. A system designer needs to consider the design as a system and realize that any change potentially has an impact beyond the sub-system. It is not usually possible to make an architecture or hardware change irrespective of the larger view of the system. This is really the heart of system engineering, consideration of an entire system, not just a collection of sub-system parts.

This is particularly important when considering cost because it is not possible to swap out cost as modular blocks and estimate new costs without understanding that there are affects to the system. There are multilevel impacts when modular blocks are substituted. Simply swapping out a block and estimating cost gives a first order indication of the cost impact. But until the design is finalized it is only a rough estimate. There is a spiral approach to design. As choices are made, impacts are assessed, costs can be estimated, new choices are made, and eventually the design spirals into a solution.

To decide between competing variables a trade study can be employed. A trade study is a useful tool which allows a designer to compare and contrast the various possible choices to determine which solution would be "best" for the given application.

This work is intended as a tool for a cost analyst to provide a designer a "compass" on where to focus attention by identifying the components with the highest cost sensitivities.

Return On Investment (ROI): "Real" Cost Model

Consistent with the sample cost model example, all the calculations and results presented remain of theoretical importance. However, the value of this work lies in the application of the results. The question which is of primary concern relates to how a system architecture can be optimized in terms of performance and cost early in the life cycle of a program. To illustrate the significance of the results upon potential improvements to the real-life system, the following discussion is offered.

The information which appears in Table 19 includes the top 29 most Impactful Parameters. What was discovered was that a few parameters have the greatest sensitivity affecting the cost of the overall system. It is unrealistic to consider improvements to a system in such a broad number of parameters over a broad set of components. Instead, focus will be applied to a more conservative subset. What remains to be done is to modify a reasonable set of those Impactful Parameters to see realistically how it will affect system cost. In Table 22, three options are presented: option A represents a reasonable and achievable change in hardware, option B represents improvements to the top three parameters, and option C represents improvements to the top five cost driving parameters.

Table 22 - Was/Try Parameter Value Options.

Option A: Was/Try, Reasonable Expectation

Component Level 2	Parameter	Rank of impact	Parameter "Was"	Parameter "Try"
Receiver	Total CCAs	**1**	3	2
Processor	Total CCAs	**2**	2	1
Receiver	Integrated Components per PCB	**8**	45	42
Transmitter	Clock Speed	**9**	400	300
Receiver	Clock Speed	**10**	250	300

Option B: Was/Try, Multiple Teams

Component Level 2	Parameter	Rank of impact	Parameter "Was"	Parameter "Try"
Receiver	Total CCAs	**1**	3	2
Receiver	Integrated Components per PCB	**8**	45	42
Transmitter	Clock Speed	**9**	400	300

Option C: Was/Try, Top 5

Component Level 2	Parameter	Rank of impact	Parameter "Was"	Parameter "Try"
Receiver	Total CCAs	**1**	3	2
Processor	Total CCAs	**2**	2	1
Transmitter	Total CCAs	**3**	2	1
Display	Total CCAs	**4**	4	3
Power	Total CCAs	**5**	2	1

The column Parameter "Was" indicates the value of the Parameter which was used in the baseline cost calculations. The value in the "Try" column contains a reasonable modification to parameter value. In other words, with some reasonable effort, it may be an achievable goal to modify the "was" to the "try" value.

Option A is a reasonable effort. This is an option which if undertaken, it may result in achieving these goals. Options B & C are not very realistic. Option B, for example, because it involved three distinct parameters, it would require three separate disciplinary teams. While option C indiscriminately selects the top five drivers and is hardly likely to be achievable. Still, options B & C help to bound the possible improvements.

The cost model could be run with ALL the potential improvements applied simultaneously and the corresponding impact to cost could be observed. It is important to mention that no longer are the KSMs involved in the calculation. KSMs were only used to create a variation factor to understand the sensitivity, it was a theoretical adjustment. In this exercise, real values are being explored.

By modifying the cost driving parameters from a baseline value to an achievable and improved value (Table 19), it is demonstrated that there would be a significant improvement to overall system cost. Table 23 is a summary of the system cost result when the options are exercised.

Table 23 - Was/Try Parameter Value Cost Results.

Option A: Was/Try, Reasonable Expectation

$16,975,050	<-- Baseline System Cost
$13,080,099	<-- New Improved System Cost
$3,894,951	<-- Savings
23%	<-- % Improvement

Option B: Was/Try, Multiple Teams

$16,975,050	<-- Baseline System Cost
$14,799,474	<-- New Improved System Cost
$2,175,576	<-- Savings
13%	<-- % Improvement

Option C: Was/Try, Top 5

$16,975,050	<-- Baseline System Cost
$10,008,258	<-- New Improved System Cost
$6,966,793	<-- Savings
41%	<-- % Improvement

As can be seen in Table 23, the percentage improvements are 23%, 13% and 41% over the baseline which is a significant impact! Of course, as mentioned, option A is really the only option under consideration and has a 23% potential improvement in cost.

Another way to interpret this result is in terms of return on investment (ROI). To modify a parameter value, it would of course be necessary to expend some resources to achieve the new value. For example, to reduce a design from 3 CCAs to 2 CCAs some amount of resources, or investment, must be made. There must be some amount of research, design, analysis, or trade study. There must be some expended resource which yields a parameter improvement. A system designer should know the cost of that expended resource. In this case, if the system designer remains below a $3.9M investment then the project, overall, would demonstrate an improvement. In other words, a system designer could spend up to $3.9M to achieve improvements in those parameters for option A which most significantly impact overall system cost. And of course, anything less than $3.9M contributes to profit margin. If those achievements could be realized there would be a 23% improvement in overall system cost which is clearly a significant improvement.

Conclusion

This paper documents the application of a cost sensitivity algorithm upon the various components in a system to analyze and determine which subsystem components in a chosen design solution have the highest sensitivity to overall cost. This paper highlights the areas to which a system designer could apply focus to reduce the overall system cost early in the life cycle of a program. It was shown using sensitivity analysis that a cost sensitivity algorithm was developed including a discussion on key size metrics. It was shown the cost sensitivity algorithm was applied to a sample cost model and that it demonstrates which component parameters were most sensitive and the biggest cost drivers in the system design. In addition, an alternative was suggested which offered the system designer a significant opportunity to improve cost. A return on investment (ROI) was calculated using the result to suggest a trade study budget for achieving the potential cost improvements. The fully developed cost sensitivity algorithm with KSMs was then applied to a “real” example cost model based on a generalized work breakdown structure (WBS) for a RADAR system applied to military applications in the aerospace industry. The result of applying the algorithm was presented and demonstrates the significant impact achievable to overall system cost when focus is applied appropriately to the areas for which overall system cost is most sensitive. And finally, for the “real” example, the algorithm highlights the areas where trade studies could be performed and yields a target return on investment (ROI) budget to achieve those cost improvements at the beginning of the life cycle of a program.

Tools, Resources, and Methods

The primary tool which was utilized in the development of the cost sensitivity algorithm was SEER, by Galorath. The software package allowed the author to consecutively iterate through a variety of cost model options and collect the output data. This effort could likely have been accomplished with another commercially available tool as well.

In the future, it is hoped that the cost sensitivity algorithm will be made available for any system designer to utilize. In particular, the possibility exists to incorporate this algorithm in the commercially available cost tools to eliminate the cumbersome nature of the algorithm and accomplish the same results with a few simple clicks of the computer mouse.

Verification / Validation Method

Verification is the process of determining whether or not a design alternative fulfills the requirements or specifications established for it. In this case, the goal was to offer the reader a method by which a system designer could have a compass to find design alternatives to positively impact the overall system cost early in the life cycle of a design. This paper, and in particular chapter 8, documents the application of a cost sensitivity algorithm upon the various components in a system to analyze and determine which subsystem components in a chosen design solution have the highest sensitivity to overall cost. It was shown using sensitivity analysis that a cost sensitivity algorithm was developed including a discussion on key size metrics. The fully developed cost sensitivity algorithm with KSMs was then applied to a "real" example cost model based on a generalized work breakdown structure (WBS) for a RADAR system applied to military applications in the aerospace industry. The result of applying the algorithm was presented and demonstrates when focus is applied appropriately to the areas for which overall system cost is most sensitive a potential improvement in cost of 23% for the proposed system is possible.

Validation is the assessment of a planned or delivered system to meet the customer's operational and/or financial need. In this case it would mean using the algorithm to help select an optimized solution and demonstrate that it was indeed the best choice. The most comprehensive way to validate optimized cost predictions is to estimate cost for each possible solution, create all the system design alternatives and then compare actual costs across all the various solution alternatives against the predicted costs. This is clearly prohibitive for application of a cost sensitivity algorithm with regards to a complex system. It is simply not feasible to create multiple system designs to validate that a system optimization was successful. So, the question of validation must be justified by some other means.

In this application, a system designer is trying to avoid incurring cost and is instead directing their efforts towards discovering potential lower cost alternatives. The higher cost alternatives are avoided meaning there is no actual higher cost systems for which to collect cost. It is a situation of cost avoidance.

For a subsystem or component, the case is considerably different. Take for example an amplifier where a certain performance is desired. The investment for an amplifier as compared to that of a complex system is negligible. As such, multiple design alternatives could be explored. In fact, it is quite common for an amplifier designer to undertake a trade study which includes multiple design alternatives. Each amplifier could be designed, fabricated, and tested. The trade study would consider multiple criteria such as cost, performance, schedule, etc. The criteria would be scored and weighted yielding a numerical value which indicates a preferred design selection, or a design “winner”.

By contrast, a complex system such as a RADAR system applied to military aerospace applications in the aerospace industry could run in the tens of millions of dollars. In addition, resources to develop systems are typically in short supply. This includes not only material resources but also experienced and talented designers. Typically, a system in the aerospace industry is already struggling to find resources for one design. So, for a complex system it is simply not practical to develop multiple systems and then down select by means of a trade study. Again, the question of validation must be justified by some other means.

When proposing a new system solution, a system designer can reference a legacy design, system A. The new design, system B, will be based on the legacy design, system A. By using the developed algorithm, alternatives could be investigated in a theoretical space. Those alternatives would include modifications which would affect performance which would need to be evaluated but also would include a reduction in overall system cost. In other words, system B will be "mostly" the same as system A but with some differences which likely adversely affect performance, but which positively impact cost. The supporting simulations and trade studies would be sufficient to form the practical method for validating the algorithm. However, the goal of the research was to offer a system designer tools for identifying where to apply focus. And the next step of performing trade studies upon the various options is out of scope for this research. Of course, if system B is selected, designed, developed, and produced, the cost could be collected and then in retrospect demonstrate further that the algorithm was validated.

Another method for validation would be to take a past program with actuals, apply the algorithm, and compare the potential cost improvements against the actuals previously recorded. This method is problematic because program actuals and design details are considered intellectual property and is typically tightly controlled. To overcome this limitation, a standardized WBS structure and a corresponding cost model was created to represent a "real" system (chapter 7). And the algorithm was applied to that "real" system to demonstrate potential improvements (chapter 8). This in part was a method used for validation.

Another means employed to validate the proposed algorithm is by way of social data or expert opinion. To that end, a few routes were explored. In the generation of the algorithm, the software SEER by Galorath, a commercial off the shelf cost software package, was utilized. Upon completion of the research, the algorithm was presented to Galorath and the reaction was that the algorithm was "novel" and "exciting". Initial groundwork was established to consider the feasibility to add the algorithm to the COTS package as one of the available standard tools. In addition, both chapters 7 and 8 were submitted for peer review with the Journal of Cost Analysis and Parametrics (JCAP). The journal is part of the International Cost Estimation and Analysis Association, an internationally recognized association. In addition to both articles being accepted for publication, the editor remarked that this material is wholly in line with the interests of the readership, consisting of cost analysts and industry experts. In addition, the material was presented at local chapter meetings of both INCOSE and ICEAA with very favorable audience reactions. In addition, the material was presented to multiple executives at Raytheon to obtain initiative funding for further research and was given encouragement to file formally for the initiative funding.

Summary

This paper documents the generation of a cost sensitivity algorithm of the various Components in a system to analyze a System and determine which Subsystem Components in a chosen design solution have the highest sensitivity to cost for the overall system and highlights the areas to which a System Designer could apply focus to reduce the overall System cost early in the life cycle of a Program. It was shown that a cost sensitivity algorithm was developed. Early research efforts to create an Excel based ROM cost model, although rudimentary, revealed that it was possible to influence design choices using cost sensitivity analysis. Research was performed regarding a standardized block diagram and standardized Work Breakdown Structure. It was then demonstrated that a cost sensitivity algorithm was developed and was applied to a sample cost model. The results demonstrated which Component Parameters were most sensitive, and the biggest cost drivers in the System design. In addition, an alternative was suggested which offered the system designer a significant opportunity to improve cost. Then, the cost sensitivity was applied to a real-life system cost model based on the standardized WBS structure. And finally, Return On Investment, or ROI, was calculated to suggested a Trade Study budget for achieving the potential cost improvements. The potential cost improvements with some realistic design alternatives were demonstrated to be a 23% improvement in overall system cost which is clearly a significant improvement.

Conclusion

It is possible to offer a System Designer some meaningful feedback regarding a potential design using a cost sensitivity algorithm in conjunction with a cost model. The algorithm, if automated, can provide a System Designer immediate cost saving alternatives and areas to apply focus at the beginning of the life cycle of a Program.

Presentation Plans and Publications

In support of this proposal, new research has already been completed and peer reviewed as either a publication or presentation. A listing of that work is included here:

Chapter 2: 2019 Microwave Journal – Development of a Time Allocation Method using a Performance Algorithm Applied to a Department Manager as a Resource.

The paper referenced in this section is for submission with Microwave Journal for 2023 publication, estimated.

Chapter 4: 2014 ICEAA – Building a Complex Hardware Cost Model for Antennas

The information referenced in this section was accepted and presented in June 2014 at the ICEAA conference in Denver, CO.

Chapter 5: 2015 ICEAA – Development of a "Similar-To" Basis Of Estimate (BOE) generation tool used in conjunction with a Complex Parametric Antenna Cost Model

The information referenced in this section was accepted and presented in June 2015 at the ICEAA conference in San Diego, CA.

Chapter 6: 2017 Microwave Journal - Digital Control and a Delay Line to Frequency Lock an Oscillator

The information referenced in this section was accepted and presented in the with Microwave Journal Magazine, appearing as part of their new on-line tutorials in February 2020.

Chapter 7: 2022 JCAP (peer reviewed journal) - Foundation of Structured Architecture, System & Cost Modeling

The information referenced was peer reviewed and published in JCAP, April 2022. Early research materials and concepts were written as a paper and the abstract has been accepted for presentation in May 2020 at the ICEAA National Symposium. The information was presented in June 2020 at the southern California regional chapter meeting for INCOSE

Chapter 8: 2022 JCAP (peer reviewed journal) – A System Engineering Approach Using Sensitivity Analysis For Reducing System Cost

The information reference was peer reviewed and accepted for publication in JCAP, October 2022.

Contributing Author 2020 May ICEAA Symposium - Building a parametric model for asset management

Contributing Author 2020 May ICEAA Symposium - Forecasting software entitlement demand

Contributing Author 2019 Raytheon Innovation Challenge - Test analysis and reasoning system for automation and normalization

References

[1] D. Polidi, "Development of a Time Allocation Method using a Performance Algorithm Applied to a Department Manager as a Resource," Microwave Journal, 2022 (est.).

[2] D. Polidi and D. Bloom, "Building A Complex Hardware Cost Model For Antennas," in *ICEAA Cost Symposium 2014*, Denver, 2014.

[3] D. Polidi and D. Bloom, "Development of a "Similar-To" Basis Of Estimate (BOE) Generation Tool Used in Conjunction with a Complex Parametric Antenna Cost Model," in *ICEAA Cost Symposium 2015*, San Diego, 2015.

[4] D. Polidi and M. Crist, "Digital Control and a Delay Line to Frequency Lock an Oscillator," *Microwave Journal,* vol. 63, no. 3, pp. On-line Spotlight, March 2020.

[5] D. Polidi, M. Crist and V. Chandrasekar, "Foundation Of Structured Architecture, System & Cost Modeling," in *ICEAA Symposium 2020*, Dallas, 2020.

[6] D. Polidi, M. Crist and V. Chandrasekar, "Foundation Of Structured Architecture, System & Cost Modeling," *Journal of Cost Analysis and Parametrics,* vol. 10, no. 2, pp. 116-134, April 2022.

[7] D. Polidi, "A System Engineering Approach Using Sensitivity Analysis For Reducing System Cost," *Journal for Cost Analysis and Parametrics,* vol. 10, no. 3, pp. (est.) 2-17, October 2022.

[8] P. Lacomme, J.-C. Marchais, J.-P. Hardange and E. Normant, Air and Spaceborne Radar Systems: An Introduction, Norwich, New York: William Andrew Publishing, 2001, p. 373.

[9] P. Tait, Introduction to Radar Target Recognition, N. Stewart and H. Griffiths, Eds., London: The Institution of Engineering and Technology, 2009, p. 26.

[10] A. V. Alejos, M. G. Sanchez, I. Cuinas and M. Dawood, "Wideband Noise Radar based in Phase Coded Sequences," in *Radar Technoloogy*, G. Kouemou, Ed., Croatia, Intech, 2010, pp. 52-54.

[11] G. W. Stimson, Introduction To Airborne Radar, Second ed., Raleigh, NC: SciTech Publishing, Inc., 1998, p. 15.

[12] M. I. Skolnik, Introduction To Radar Systems, New York: McGraw-Hill, Inc., 1962, p. 5.

[13] J. F. Reintjes and G. T. Coate, Principles Of Radar, Third ed., New York: McGraw-Hill Book Company, Inc., 1952, p. 2.

[14] H. E. Penrose and R. S. Boulding, Principles and Practice of RADAR, London: Whitefriars Press Ltd., 1952, p. 96.

[15] D. G. Fink, Radar Engineering, First ed., New York: McGraw-Hill Book Company, Inc., 1947, p. 11.

[16] O. E. Dunlap Jr., Radar: What Radar Is And How It Works, Third ed., New York: Harper & Brothers, 1946, p. 86.

[17] L. Varshney, "Radar System Components and System Design," Syracuse Research Corporation, Syracuse, 2002.

[18] N. P. Group, "Radio Detection and Ranging," *Nature,* vol. 152, pp. 391-392, 2 October 1943.

[19] R. F. Graf, "Antenna," in *Modern Dictionary of Electronics*, Newnes, 1999, p. 29.

[20] D. Rutledge, "The electronics of radio," New York, Cambridge, 1999.

[21] APITech, "RF Amplifiers," *APITech,* 18 May 2020.

[22] L. D. Wolfgang, "The ARRL Handgook for Radio Amateurs Sixty-Eighth Edition," ARRL, 1990, pp. 12-10.

[23] I. Poole, "Double balanced mixer tutorial," Adrio Communications, 2012.

[24] T. Instruments, "Principles of Data Acquistion and Conversion," Texas Instruments, 2015.

[25] F. Bedell, "History of A-C Wave Form, Its Determination and Standardization," *Transactions of the American Institute of Electrical Engineers,* vol. 61, p. 864, 1942.

[26] P. R. Morris, "A History of the World Semiconductor Industry," Institution of Electrical Engineers, 1990, p. 18.

[27] ARRL, "The ARRL Handbook, 1968," ARRL, 1968, p. 50.

[28] D. G. Fink and H. W. Beatty, "Standard Handbook for Electrical Engineers Eleventh Edition," Mc Graw Hill, 1978, pp. 7-30.

[29] D. Crecraft and D. Gorham, "Electronics, 2nd Edition," CRC Press, 2003, p. 168.

[30] J. Borky and T. H. Bradley, Effective Model-Based Systems Engineering, Springer, 2019.

[31] B. Markgraf, "The Importance of a Work Breakdown Structure," *Small Business Chron,* 28 January 2019.

[32] Christopher Sichko, "Normalized Prices," United States Department of Agriculture, 2019.

[33] Mislick, Gregory K; Nussbaum, Daniel A;, "Cost Estimation: Methods and Tools. Ch5 Data Normalization," in *Cost Estimation: Methods and Tools. Ch5 Data Normalization*, Wiley, 2015.

[34] ICEAA-CeBoK, "Data Collection and Normalization, Unit II - Module 4," ICEAA Online, 2013.

[35] M. E. Porter, Competitive Advantage: Creating and Sustaining Superior Performance, Free Press, 1998.

[36] Federal Aviation Administration, NAS System Engineering Manual, Ver 3.1, Air Traffic Organization, 2006.

[37] TAMU, "Trade Studies: Finding the best alternative for system performance from available options," Texas A&M University, College Station, 2019.

[38] Felix, Art, "Standard Approach to Trade Studies: A Process Improvement Model that Enables Systems Engineers to Provide Information to the Project Manager by Going Beyond the Summary Matrix," in *INCOSE Mid-Atlantic Regional Conference*, 2004.

[39] Felix, A, "Standard Approach to Trade Studies," in *INCOSE 2004 - 14th Annual International Symposium Proceedings*, 2004.

[40] Cilli, Matthew V.; Parnell, Gregory S.;, "Systems Engineering Tradeoff Study Process Framework," *INCOSE,* p. 17, 2014.

[41] V. Belton and T. J. Stewart, "Multiple Criteria Decision Analysis," *Springer-Science+Business Media, B.V.,* 2002.

[42] C. Zopounidis and P. M. Pardalos, Handbook Of Multicriteria Analysis, Springer, 2010.

[43] T. L. Saaty, "How to make a decision: The Analytic Hierarchy Process," *European Journal of Operational Research,* vol. 48, pp. 9-26, 1990.

[44] T. L. Saaty and L. G. Vargas, "The Analytic Network Process," in *Decision Making with the Analytic Network Process*, Boston, MA: Springer, 2013, pp. 1-40.

[45] Forrester, Jay W.; Breierova, Lucia; Choudhari, Mark, "An Introduction to Sensitivity Analysis," *MIT System Dynamics in Education Project,* p. 64, 6 September 1996.

[46] S. Bingulac, "Minimum Sensitivity Adaptive Systems," in *IFAC Proceedings Volumes*, 1964.

[47] D. J. Pannell, "Sensitivity Analysis of Normative Economic Models: Theoretical Framework and Practical Strategies," Agricultural Economics, 1997.

[48] P. Glasserman and S. Tayur, "Sensitivity Analysis for Base-Stock Levels in Multiechelon Production-Inventory Systems," *Management Science,* vol. 41, no. 2, pp. 263-281, 1995.

[49] B. Iooss and A. Saltelli, "Introduction: Sensitivity Analysis," in *Handbook of uncertainty quantification*, Springer, 2017.

[50] L. Nurhadi, S. Boren and H. Ny, "A sensitivity analysis of total cost of ownership for electric public bus transport systems in Swedish medium sized cities," *Transportation Research Procedia,* vol. 3, pp. 818-827, 2014.

[51] A. MacGillivray, H. Jeffrey, M. Winskel and I. Bryden, "Innovation and cost reduction for marine renewable energy: A learning investment sensitivity analysis," *Technological Forecasting & Social Change: An International Journal,* vol. 87, pp. 108-124, 2014.

[52] PRICE Systems, "Processes & Best Practices: Design-to-Cost Target," https://www.pricesystems.com/cost-estimating-process-integrations/, 2020.

[53] COSYSMO, "Constructive Systems Engineering Cost Model," http://cosysmo.mit.edu/, 2020.

[54] Galorath, "SEER for Hardware, Electronics and Systems," https://galorath.com/our-products/, 2020.

[55] V. Fragoulakis and C. Mitropoulou, Economic Evaluation in Genomic Medicine, Academic Press, 2015.