Abstract— The Air Force Research Laboratory has recently expanded its ability to assess processing architecture options for next generation mission applications, providing traceability from Air Force mission requirements to the hardware and software implementations necessary to meet those requirements. Of prime concern has been the distinct challenge of quantifying the benefits at the spacecraft/mission level resulting from investments in space electronics technologies at the device and component level. Here, we present a satellite architecture modeling and simulation framework that incorporates a detailed accounting of the spacecraft’s energy chain of custody, while providing the flexibility to rapidly and easily change or modify spacecraft components. The model is shown to successfully yield the minimum energy configuration of a hypothetical surveillance satellite mission. This powerful new tool enables creation of a return on investment trade space, and ultimately provides the necessary quantitative analysis to guide/justify technology investment.

TABLE OF CONTENTS

1. INTRODUCTION ....................................................... 1
2. MODELLING METHODOLOGY ............................... 1
3. ENERGY AS A METRIC ........................................... 5
4. MODEL APPLICATION AND DISCUSSION ............... 7
5. CONCLUSION........................................................... 9
ACKNOWLEDGEMENTS ............................................ 10
REFERENCES............................................................. 10
BIOGRAPHY .............................................................. 10

1. INTRODUCTION

Radiation hardened component development requires an enormous cost and time commitment from the organizations that are spearheading the research, development and procurement of next generation technologies for space applications. As an acquisitions organization, it is the Air Force Research Laboratory’s responsibility to ensure that, to the best of their knowledge, resources are invested in technologies that will yield the greatest gains to the spacecraft missions. Unfortunately, the current methods for the identification of future technological capabilities required to support next generation DoD systems lack the quantitative sophistication and detail needed to compare system gains resulting from Size, Weight and Power (SWaP) and performance improvements at the device/component level. As a consequence, it is incredibly challenging to justify technology investment decisions.

Current research efforts at the Air Force Research Laboratory’s Space Electronics Branch have focused on characterization and identification of optimized computing architectures suitable for different classes of computations. This research is being actively pursued at different levels of granularity from benchmarking at the computation kernel level through development of device metrics and space dwarfs [1] to benchmarking at the mission application and algorithm level in a newly equipped hardware architecture laboratory [2]. These efforts support the identification and optimization of hardware requirements for future Air Force missions; however, they do not offer insight into impact of the technology at the spacecraft level, and they do not provide a method to compare the benefits of one technology insertion over another. In this paper, a modelling and simulation framework is presented that enables creation of a trade space which quantifies the gains at the spacecraft level resulting from power/efficiency improvements in dissimilar components. The paper is organized as follows: In Section 2, the framework for a detailed energy accounting model is presented, followed by the introduction of Energy as a Metric in Section 3. In Section 4, the model is used to identify the minimum energy configuration of a hypothetical surveillance mission, and its implications are discussed. The conclusions and future research objectives are given in Section 5.

2. MODELLING METHODOLOGY

There has been a long-standing interest in establishing quantitative approaches to answer abstract questions that relate to the impact of technology choices. An example of such a question might be: Is it better to invest in a method to improve the efficiency of solar panels by 0.5% or to invest in an approach that reduces the power consumption of all processor components by 50%? Since research budgets are finite, it is often necessary to make limiting choices from a field of perspective technology development possibilities. We should not be surprised to find that often these choices are made with little quantitative basis. It is sometimes only after significant time and money investments are made that we can retrospectively judge impact. Such motivations led to the pursuit of a general framework against which these (and other) questions could be answered objectively, providing a knowledgeable trail of logic behind these decisions.

Our modelling and simulation framework is rooted in the assumption that every action performed on the spacecraft expends some measurable amount of energy; therefore, energy is the metric by which gains at the satellite level...
should be quantified. To first order, the tool captures the “day-in-a-life” energy investment of a spacecraft, modelled as the sum of contributions from its individual components as the spacecraft is propagated through an orbit with complex, time-varying operational modes representing the typical behavior of the spacecraft performing its mission. To create the impact analysis trade space, the tool provides functions to modify or completely replace low level components, as well as modify the operational mode schedule prior to successive simulations. This functionality has been demonstrated in a C/C++ proof of concept version of the code and is currently being developed into a form more approaching a production level version suitable for direct use by end users.

While one of the motivations for creating this framework has been to provide an objective basis for technology investment decisions, we see a much broader set of possible useful cases, to include platform size, weight, and power (SWAP) optimization, mission energy analysis, and support of speculative constellation development. In each of these cases, a reasonably detailed understanding of energy usage can lead to simpler mission designs, based on platforms that could have lower launch cost due to SWAP minimization. Risk tradeoffs can also be better understood in terms of how lower risk options may (or may not) allow energy budgets to close in prospective platform architectures. Perhaps an exciting new (but risky) application-specific integrated circuit having tremendous performance benefits seems like a good decision in a spacecraft upgrade, but through an objective energy-based account it might be shown to have only a marginal added benefit based on workload vs. timeline considerations. It may, for example, complete on-board processing tasks in 10% of the time, but in situations where energy supply is abundant and the advanced processor spends much of an orbit in idle mode, the justifications for change may not be warranted, eliminating a source of added risk and cost.

The following outlines the modelling rationale and the associated methodology used to establish a chain connecting high-level satellite operational modes to a recursion of low level component mode settings (whose settings drive power consumption) and propagate the actions of a platform (e.g., satellite) or system and its concomitant energy usage through a “day in a life” on orbit.

**Architecture Model**

In a broad sense, the satellite is modelled hierarchically as a tree of nodes (i.e. Spacecraft, Subsystem, Component, etc.) as suggested in Figure 1. A *root* node represents the highest level, in this case a spacecraft platform. In principle, constellation of platforms can also be modeled in which the constellation represents the root node, and major platforms are subtrees. These nodes can be described at different levels of fidelity based upon one’s understanding of the subsystem, which is useful for rapid modeling in the presence of imperfect or incomplete knowledge. Unique nodes are derived from a base class with attributes of type, power and mass. Generally, these attributes are aggregated from the bottom of the hierarchy and percolated toward the top. The framework can be readily extended to encompass an arbitrary set of extended attributes that complement the energy-based accounting methodology (which is the primary emphasis of this tool). There are three fundamental node types; source, storage, and load. Source and storage nodes are used to capture the “supply side” of the energy balance, to include solar power generation and energy storage capacity characteristics of the spacecraft. These node types are modeled in a more rudimentary form compared to some industrial simulation tools (e.g., Spice or Spectre), as for our purposes, we do not believe a more detailed model adds useful insight to understanding energy use (e.g., consumption). Furthermore, we believe if useful, it will not be difficult to meld more powerful simulation frontends into the present framework in the future. Load nodes, which express the “demand side” elements of the accounting methodology, are the building blocks by which the satellite architecture model is constructed, and they play a fundamental role in power calculations. In the simplest case, satellite platform power would be modeled as a constant, independent of time, so that orbit energy could be calculated as the orbit duration (s) multiplied by the satellite power (W). In reality, a spacecraft has time-varying behavior, which necessitates a more sophisticated treatment to achieve more approximate orbit energy calculations. An accurate model must be able to capture the dynamics of the operational mode schedule and properly calculate changes in the bus and payload power requirements consistent with orbital / operating environment (to include the capture of energy from the sun and the geometry of other mission elements, such as ground targets, users, and operating stations).

We next describe the translation of methodology to implementation. To accomplish our initial implementation, the idea of operational mode mapping was formulated. To better understand this concept, it is useful to define a few terms. The spacecraft or platform model hierarchy is a directed tree (graph structure). A *parent* node is defined as a node that has one or more children or *child nodes* below it in the hierarchy. In Figure 1, “SubSys3” is a child node of “Spacecraft” and a parent node of “Comp4”. A *leaf* node has no child nodes below it in the hierarchy. As in any directed tree, a single distinguished *root* node (which has no parent) occupies the topmost level of the hierarchy.

The data structure representation of nodes promotes flexibility and extensibility. For example, there is no pre-ordained limit to representation depth. An entire subsystem can be modeled as a constant load or painstakingly detailed as a complex subtree containing thousands of components. This allows for quick abstractions in the case of incomplete knowledge (or in mocking up prospective architectures having components that do not yet exist). The models can be replaced iteratively as better, more detailed
representations are available. For economy in representation, node models can be defined once and re-used through a platform design, consistent with the practice used in many simulation tools. This approach is particularly convenient in the cases where it is desirable to examine the effects of replacing commonly occurring components with energy-efficient variants.

Every node in a hierarchy must have one or more local operating mode(s). The assignment is recursive, meaning that modes of high-level nodes must be mapped to corresponding modes of each child node. In the first-order model, this usually implies that only leaf nodes carry power assignments, which aggregate upwards as the hierarchy is ascended. Hence, the power value of any node is simply the sum of power values of all child nodes, and their values depend on the mode setting of the parent node. These settings ripple through the entire hierarchy. Thus, all nodes carry a ModeMap table (stored in our implementation as a linked list), which establish unambiguously the settings of modes to that of the parent node. The only exception is the spacecraft’s root node, where the ModeMap table is NULL (since the root does not have a parent). The root node will have local operating modes corresponding to top-level operational modes of the spacecraft, analogous to those specified in commanding the system during its day-to-day operations. Each of these top-level modes will trigger different mode changes all the way down the hierarchy; such that, the characteristics/variables of the individual nodes feed the recursive calculations used to derive the power for a given top-level operational mode. A simple example of “LocalMode” and “ModeMap” tables is illustrated in Figure 1 for one branch of the hierarchical structure. In this case, the spacecraft mode “α” returns 5 W, and mode “β” returns 1 W. Here the power is the sum of one component’s contribution. In a more complex system the upper level power is derived from the contribution of many components (there is no predefined limit, and the total recursion can conceivably involve thousands of nodes, even in a small spacecraft).

The accuracy of the power model based on this approach depends on the degree of fidelity used to model nodes and capture their mode mappings. It is clear that precise mode definitions are essential to the accuracy of the calculation, and these details can be difficult to obtain in practice, especially when component descriptions are restricted (due to proprietary or security limitations on the distribution of

Figure 1. Hierarchical decomposition of the spacecraft into a tree of nodes that carry attributes of type, power and mass. A component’s “LocalMode” table defines operating modes and power, while the “ModeMap” table defines how local modes map to the Parent’s operating modes. Simple example is shown for one branch of the hierarchical structure.
needed information. This highlights the utility of a dedicated architecture laboratory where the necessary accesses can be obtained so that space electronics components can be characterized in detail. It is also possible for these dedicated laboratories to broker “safer” sanitized and generic representations of platform descriptions that can be made accessible to a broader research community.

**Orbit Simulation**

Our discussion so far has focused on the creation of the architecture models where instantaneous power requirement can be derived from a set of top level operational modes that drive a cascade of mode settings down to leaf nodes that propagate particular power values upward (towards the system root node). In order to implement a simulation, it is necessary to propagate the representation of a spacecraft platform over a time interval, most usefully through one or more orbits, consistent with the actions a platform would take while executing its intended mission. Energy is estimated by integrating the time-varying power states of the architecture model using a discrete event simulator (DES) engine that steps the satellite model over time. The user creates a simulation file that describes the desired operating schedule of the spacecraft. Our initial implementation allows for definition of exceptions, which override the preprogrammed architecture mode mapping behaviour. For example, we permit the addition of rules such as: “always turn on the heater when the satellite is in eclipse”. The simulation file also allows the user to define satellite orientation power scaling coefficients. Scaling provides a simple method for capturing the dynamics of reduced power to the bus when the solar cells are not at normal incidence to the sun. While the initial implementation does not support detailed orbit modelling, the addition of a more advanced orbit simulator is highly desirable, as it can be used to derive information such as solar flux (for establishing power production from photovoltaic arrays) based on spacecraft attitude geometry and orbital trajectory. The software constructs three tables based on the simulation file: Mode Schedule, Orientation and Re-Mapping. Upon user definition of orbit characteristics, the DES engine determines the temporal sequence of discrete events where the operating mode, environmental condition or satellite orientation changes. Satellite power is assumed to be constant over the time intervals ($\Delta t$) that separate discrete events. This power ($W$) is calculated and multiplied by the appropriate $\Delta t$ (s) to get the energy for that time step. This is computed for the entire
orbit and results in the total orbit energy (J). To enable comprehensive trade studies, the software includes a global find and replace capability that allows the user to modify or swap-out components on an individual basis or as a group. The user can also modify the ModeMap table, operating mode schedule, or orbit characteristics with relative ease and re-run the simulation. In Section 4, an example will be presented where the total orbit energy was traded against the operational characteristics of the mission, comparing on-orbit processing versus transmission of data to the ground station.

Model Use-Case

The modeling framework presented here enables the unique capability to examine and quantify the impact of low level component decisions at the spacecraft level. This is a powerful tool for program managers who have to make decisions about which technology investments will yield the greatest returns. Figure 2 shows the use case decision diagram for a typical customer. An example scenario could be to seek the minimum energy option between reducing the power in a subsystem by 50%, versus increasing the bus regulator efficiency by 10%. The answer may appear trivial at first glance, however, many variables factor into the equation. One could make dramatic improvements in the power of a subsystem, but if that subsystem is only used for 3% of the orbit, it will have negligible impact at the system level. In contrast, one can envision marginal improvements in a component resulting in large net gains at the system level if many instances of that component exist in the architecture or if the component is heavily used.

3. ENERGY AS A METRIC

This section considers the utility of energy-based accounting in problems relating to spacecraft and space mission design. In general, a spacecraft is highly power constrained, limited by the available power extracted from the sun, produced from the solar panels and stored in batteries. Consider a spacecraft’s energy “chain of custody” (Figure 3) as the path traced by energy starting at its source (sun) to the endpoints that ultimately consume it within the spacecraft bus or payload components. Owing to various inefficiencies/loss mechanisms along this chain of custody (i.e., quantum efficiency of the solar cells, efficiency of the voltage regulators, resistive losses in interconnect, etc.), only a small percentage of incident photon energy is actually accessible by spacecraft components (and box, board, circuit-level drivers, and other packaging losses represent other reductions in the power that could otherwise be made available for useful work). Detailed consideration must be given to the erosion in energy at all points along the chain of custody, leading to clearer understanding that better informs which strategies have the most impact on the system as a whole.

At this point, we confront several obvious points:

- Energy supply must exceed demand over "all time" as far as mission is concerned.
- There is an energy associated with workload done by the payload, and necessarily the bus that carries it, in order to perform the mission.
The workload must be completed in a temporally useful deadline.

The last observation is important. Having a dramatically faster computer may only mean we can complete a given workload more quickly, with the possibility of being idle most of the time (missions do not always choose to exploit performance advantages). It may be better to make a computer do the same workload with 90\% less energy than simply doing it more swiftly.

Energy optimization of missions

We can make another general observation. Spacecraft missions (in particular) inevitably trade between one of four idealized options for the information they encounter (Figure 4). They must either: (1) process the data obtained from a sensor, (2) store the data in memory, (3) transmit the data off the spacecraft, or (4) discard the data (or never collect it). Each of these choices has an energy consequence, and disparities between them can drive new mission concepts. For example, if processing information is considered “free” from the position of energy investment (the infinite on-board processing case), then abundant amounts of computation should be employed in order to minimize the amount of information stored or sent to the ground. If storage is free (the infinite memory case), then all details pertaining to the spacecraft health or mission data should be stored for potential use at a later date. If transmission of information is free (the “infinite data pipe” case), then even the most insignificant fragments of information could be sent to the ground and there would be no need process or store anything on the spacecraft. In reality, none of these ideal scenarios are possible; however, we believe an energy optimized strategy can be identified for all missions. The energy slider framework presented in the following section examines such an optimization, an attempt to capture the dynamics of an on-board processing versus transmission trade study in order to demonstrate the utility of a detailed energy accounting approach to explore mission concepts.

Energy optimization and architecture alignment for spacecraft platforms

Even when missions are constrained, energy-based analysis can be valuable in making decisions regarding the choice of algorithms, processing architectures, or the design of networks and interconnect within a spacecraft. When a mission is rigidly defined, then so is usually the case that the workload that a spacecraft must perform is also constrained. In this case, cost and risk may be more important drivers than energy or performance.

In this regard, constrained missions care more about the “right-sizing” of processing, communications, and other resources. Disconnects, in either scarcity in overabundance, can result in mission infeasibility or overdesign. Energy-based analysis can still be a useful tool, because it can help confirm the viability of cost driven technology choices to meet the constraint that “workload must be completed in a temporally useful deadline” (since an energy deficit would at least identify a deficiency in energy supply). We may, for example, choose an array of less expensive processors instead of one expensive component. By evaluating them both in an energy-based framework, we may confirm that both of them meet the workload timeline criteria, allowing costs and risk factors to drive the best decision.

![Figure 4. Macroscale energy trade for processing.](image-url)
It is noted that while energy should be a fundamental consideration for system engineering studies, it is not the only metric that drives mission concepts or hardware decisions. Consider for example an exploratory processor study for a specific application: the traditional approach might be to assess the processor’s performance on a popular synthetic benchmark, then attribute that result to the suitability of the processor to meet the application requirements. This approach is broken in the sense that while performance on a synthetic benchmark may prove useful for measuring specific features, such as a processor’s ability to perform integer or floating point mathematical operations, it fails to offer reliable insights into the performance on the real applications of interest [3]. Recognition of this has led to the creation of the aforementioned architecture laboratory [2].

Technology comparison

The idea of energy-based accounting has broad applicability for comparing technology choices, such as the choice between two options in each of these contrived examples:

- A faster processor and an improved efficiency point of a power converter;
- a 32nm FPGA and a 65nm ASIC;
- a 3% improvement in solar cell efficiency and a decentralized power distribution architecture;

The principle involves performing an “A vs. B” assessment of the relative energy implied by the choices in question. Simply put, the option with the lowest magnitude of energy (kilowatt hours), when evaluated over representative mission lives in representative platforms (as many as necessary), is the “winner”. It can be argued that energy is not always the most important metric, but it defines a quantitative basis against which even wildly dissimilar technology options can be traded. It provides at least one objective approach for making decisions.

4. MODEL APPLICATION AND DISCUSSION

In this section, we discuss work to apply our initial implementation of the energy accounting model described in Section 2 to a hypothetical surveillance satellite mission. One objective is to identify the minimum energy configuration of the satellite as the amount of on-board payload processing is varied from “100% processing” (performing all complex image computations on-orbit), transmitting only processed results, to “100% transmission” (sending all of the raw unprocessed image data to a satellite ground station with no additional on-orbit processing). This analysis requires a detailed understanding of the transmission costs of moving data off the spacecraft, as well as a detailed understanding of the power requirements and temporal characteristics of the computation.

The architecture employed in this study is a 32-bit ARM big.LITTLE architecture, comprised of 4 ARM-15 cores and

Figure 5. (Top) Image processing algorithm execution flow diagram. (Middle) Energy profile and execution time of image processing algorithm on 32 bit ARM big.LITTLE architecture utilizing 8 cores. (Bottom) Minimum number of raw uncompressed data bits required to be transmitted for further target processing as a function of processing time and data reduction.
4 ARM-7 cores. This COTS computing platform is vastly more capable than legacy radiation hardened computers that are being flown today, but would surely fail to meet the environment requirements outlined in MIL_STD-883 [4] for a multi-year mission. Nevertheless, architectures such as this are becoming widely popular for use in short mission life satellites such as CubeSats. An image processing code for a generic surveillance application was optimized for performance on this platform. To capture the computational power requirements of this application for an “energy slider” assessment (the convex combination of processing fraction and transmission fraction), the application is decomposed into a sequence of key steps. In Figure 5 (top), the image processing steps are outlined. At the conclusion of each of these steps the accumulated time from program initiation is printed out. This yields the temporal profile of the application as it processes a dataset. Separately, the power draw is recorded at 10 samples/sec during program execution. Instantaneous power draw for this particular configuration ranges from 18 W down to 5 W; however, the time integration of the power profile results in a relatively smooth, straight, line suggesting that the average power is fairly constant. A least squares fit to the integration data followed by a time derivative yields an average power of 7.6 W. This is represented in Figure 5 (middle). The next step is to compute the transmission costs (energy) required to move data off the spacecraft. To do this, one must first know the minimum number of bits that must be sent to the ground at any point in the image processing sequence to complete target processing. For this particular analysis the initial raw data set is assumed to be 520Mb. Detailed investigation has suggested that the data can be reduced after the first feature extraction stage [5]. The level to which the data can be reduced and still yield reliable results varies as a function of image background, jitter and target magnitude. Details of these numbers are not presented here. For this analysis, data reduction between 2X and 50X is examined. Figure 5 (bottom) shows the notional data size following the intermediate algorithm steps, reducing to the size of the final result upon completion. Note that this is not the number of bits of data the code is operating on but rather the minimum number of bits that must be transmitted to the ground at that point of the calculation in order to yield a usable image. This study does not account for the energy required at the ground station to reconstruct the data from the limited set of eigenvalues and continue target processing. A future study could consider energy expended at the ground station; perhaps at a reduced cost from the energy used in space.

Leveraging the modelling and simulation framework introduced in Section 2, a previously verified model of the Modular Space Vehicle (MSV) bus architecture [6], outfitted with an experimental surveillance payload, is constructed to explore the impact of mission planning decisions on the spacecraft power budget. This model contains variable levels of subsystem detail down to the component level in some cases. The instantaneous power requirements for the architecture change as a function of operating mode, ranging from 293.4 W to 389.5 W. This difference is relatively modest; in a previously studied case the power varied by over a kilowatt. A simple operational mode profile is outlined in Table 1 to illustrate the energy slider trade space.

TABLE 1: Satellite Operating Mode Trade Space

<table>
<thead>
<tr>
<th>Time (min)</th>
<th>Operating Mode</th>
<th>Duration (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Standby</td>
<td>30</td>
</tr>
<tr>
<td>30</td>
<td>PayloadWarmUp</td>
<td>5</td>
</tr>
<tr>
<td>35</td>
<td>DataCollect</td>
<td>1 to 2</td>
</tr>
<tr>
<td>40</td>
<td>OnBoardProcessing</td>
<td>Variable 1</td>
</tr>
<tr>
<td>120</td>
<td>NominalRxTxOps</td>
<td>5</td>
</tr>
<tr>
<td>125</td>
<td>Data Downlink</td>
<td>Variable 2</td>
</tr>
<tr>
<td>135</td>
<td>System Idle</td>
<td></td>
</tr>
</tbody>
</table>

It is noted that the mode durations of OnBoardProcessing and Data Downlink are undefined. These are traded against one another as the mission parameters are adjusted. The accumulated energy calculations are shown in Figure 6, plotted on a scale from 0 to 100%, representing the percentage of processing performed on orbit versus on the ground. Early in the computation, the minimum number of bits required for transmission can be reduced using what is effectively a form of lossy compression. The level of reduction is explored in Figure 6 over a range from 2X (blue curve) to 50X (purple curve). The sum of the transmission and processing times are also presented. It is found that transmission of the raw data prior to any onboard processing is by far the least favorable option, requiring the most energy (1605.5 kJ) and greatest amount of time (253.9 s). Model results suggest that in most cases, running the
application to completion on the spacecraft yields the least total energy and takes approximately 67 s. An exception is identified when the minimum amount of data required for further target processing can be compressed by 20x or greater. In this case, transmitting the residual data after the eigenvalue decomposition in the feature extraction stage results in the minimum total energy requirements (1596.33 kJ), and reduces the accumulated transmission and processing times to ~17 s. It is important to consider that this does not include the time for additional target processing at the ground station.

The underlying satellite hardware was not changed in the above assessment of mission concepts and their impact the satellite energy footprint. It is noted, however, that the amount of energy gained by optimizing the mission planning (~10 kJ) is a relatively low percentage of the total orbit energy calculated by the model (~1600 kJ). Analyzing this further, it is found that the MSV bus functions account for the vast majority of the spacecraft power budget. In fact, the payload processor and radio subsystem only contribute a combined 4% to the total orbit energy of the spacecraft; the whole trade space is almost in the noise. Consequently, a separate trade study was conducted to examine the relative impact of two different bus optimization strategies. In one case, the Inertial Reference Unit (IRU), within the Attitude Control System (ACS) subsystem, was reduced to 10% of its original power. In a second case, the Applique Sensor Interface Module (ASIM), a critical component of the MSV architecture that serves as the interface between individual components and the spacecraft bus [6], was reduced to 10% of its original power. Selection of these components for optimization was not arbitrary. The IRU consumes greater than 20 W of power but appears only once in the satellite hierarchical structure, while the ASIM uses less than 4 W of power but appears 9 times in the satellite hierarchical structure. For reference, an optimized mission concept from Figure 5 was selected. A 20/80 mix of on-orbit versus on-ground processing, respectively, with a 20X compression yields an accumulated energy of 1596.64 kJ. In Figure 7, the resulting bus optimization analysis is shown. By reducing the power in the IRU to 10%, the accumulated satellite energy falls by 7.3% to 1481.45 kJ. Resetting the power in the IRU to its original value and reducing the power of the ASIMs to 10% decreases the total satellite energy by 11.9% to 1408.07 kJ. Considerably more power was pulled out of the IRU than any single ASIM but because there are 9 separate occurrences of the ASIM in the satellite architecture, the impact of improvements in the ASIM was more significant than improvements in the IRU.

In this study, a hypothetical imaging payload was grafted onto an oversized MSV bus and two distinctly different trade studies were performed; the first examined the impact of mission concepts on the satellite energy footprint. The second examined the impact of two hardware optimizations strategies on the satellite energy footprint. The actual results from these trade studies are essentially inconsequential; of prime importance is the development and demonstration of a modeling and simulation framework that allows satellite designers quantify the gains at the spacecraft level resulting from changes in hardware, choices in mission profile, or selection of software implementation.

5. CONCLUSION

A satellite architecture modelling and simulation framework has been created based on energy accounting (supply vs. demand) to study the use of a spacecraft’s energy supply during operation on orbit. By using energy as an “unassailable metric” (every action practically has an energy consequence), we can reduce arbitrary systems (platforms such as spacecraft, groups of spacecraft, even entire missions), when evaluated over a period of time, such as a day-in-a-life (one or more orbits), to a single scalar number. We argue this permits a tool for a variety of objective assessments, such as spacecraft architecture and mission optimization and technology investment analysis. Utilizing this capability, we created a generic reference system, namely a surveillance satellite architecture (with bus component details comparable to that used in the development of the actual spacecraft) to provide simple, illustrative examples, such as the use of an “energy slider” to examine the trade between on-orbit processing and data transfer to the ground and the effects of component substitution.

In the energy slider simulation, at one extreme, 100% of the computation is completed on the spacecraft and only the result is transmitted to the ground station. At the other extreme, 100% of the image processing data is transmitted to the ground station with the minimum possible on-orbit computations being performed. It is evident from the simulation results that for this architecture, on-board image

Figure 7. Bus optimization analysis examining the net energy gains that result from two different optimization strategies: (1) reducing the power in the attitude control system’s IRU to 10%, and (2) reducing the power in the Applique Sensor Interface Module (ASIM) to 10%.
processing acts to reduce the overall energy footprint of the satellite’s day in a life. The results of the simulation suggests that maximizing the amount of on-board processing is typically ideal; however, an energy minimum was identified about 20% through the image processing computation when the data could be greatly reduced (20-50X) after the first eigenvalue decomposition.

With the same reference architecture, two separate component substitution demonstrations were performed to show how technology choices might lead to a bus optimization strategy based on energy savings. As much as 11.9% decrease in total accumulated energy was observed by dramatically reducing power in a selected component.

Even the preliminary implementation and reference model simulation examples serve as evidence to the promising utility of an energy modelling framework to examine the impact at the spacecraft level resulting from mission concepts and hardware decisions at the component level. The framework is clearly extensible, both to a larger number of platforms, as well as non-space entities (such as the combination of a spacecraft and other participating ground elements in a mission). One could, for example, develop a “fleet metric” for technology investment analysis. Such a fleet metric would provide a weighted summation of the energy contributions of multiple constellations (using “fleet” as a root node in a mega-graph structure), each propagated in a number of day-in-a-life sub-simulations. The fleet metric would serve useful in examining the broader savings of eclectic component comparisons, where the stronger benefits of a component affecting a single spacecraft is more effectively balanced by components that seem less promising to a single spacecraft, but more useful across a larger pool.

ACKNOWLEDGEMENTS

The authors would like to thank the Air Force Office of Scientific Research for support of this project, and Dr. Art Edwards (Air Force Research Laboratory) for useful discussions on energy in space computing platforms.

REFERENCES


BIography

Jesse K. Mee (Member, IEEE) received the M.S. degree (distinction) in Electrical Engineering (EE) and the Ph.D. degree (distinction) in EE with a concentration in Optoelectronics from the University of New Mexico in 2010 and 2013, respectively. His M.S. thesis examined reliability physics of microelectronics for space application, concentrating on the reliability degradation phenomenon known as Negative Bias Temperature Instability. In his doctoral research, he undertook an extensive investigation of time-domain characteristics of monolithic quantum dot passively mode-locked lasers for high data-rate transmission architectures. Dr. Mee has been working for AFRL Space Vehicles Directorate since 2008 and is the Technical Area Coordinator for the Space Electronics Technology Program’s Systems Studies Group where he leads a collaboration of military, civilian, contractors & academic S&Es under an ambitions spacecraft architecture modeling and simulation project. He has led approximately 20 in-house and contractor research efforts on satellite optical backplane, flexible power electronics for space, data compression and advanced microelectronics. Dr. Mee is on the UNM Electrical and Computer Engineering Department’s Advisory Council, and is the president of the Phillips Research Site Junior Force Council.

Andrew C. Pineda (Member, IEEE) received the B.S. degree in Chemistry from Harvey Mudd College, Claremont, CA in 1982, the A.M. degree and Ph.D. degree in Chemistry and Physical Chemistry from Harvard University, Cambridge, MA in 1984 and 1994, respectively. From 1997 to 2012, he was a Sr. Research Scientist and an adjunct faculty member at the University of New Mexico affiliated with the Center for High Performance Computing, the Department of Chemistry and the Department of Electrical and Computer Engineering.
After working with AFRL as a contractor from 2012 to 2015, Dr. Pineda recently joined the AFRL Space Vehicles Directorate. Dr. Pineda has co-authored over 20 papers on electronics materials-related topics ranging from defects in SiO$_2$, optical properties of III-V materials, molecular electronics, nanotubes, chalcogenides and dielectric phenomena. His research interests include electronic structure theory of electronics materials and high performance computing.

Second Lieutenant John Guthrie is a Project Engineer working at Kirtland Air Force Base. He achieved his Bachelor of Science in Electrical Engineering at the University of California, Davis and is currently working on a Master of Science in Computer Engineering at the University of New Mexico. Second Lieutenant Guthrie has previously worked at the Lawrence Livermore National Laboratory creating models of materials reacting under extreme temperature and pressure. He specializes in energy assessment models of spacecraft as well as cyber security analysis of satellites.

James C. Lyke (Senior Member, IEEE) received the B.S. degree in electrical engineering at the University of Tennessee, Knoxville, TN, USA in 1984, the M.S. degree in electrical engineering at the Air Force Institute of Technology, Wright-Patterson Air Force Base, OH, USA in 1989, and the Ph.D. degree in electrical engineering from University of New Mexico, Albuquerque, NM, USA in 2004. He was in active duty military service with the U.S. Air Force from 1984 through 1995. Since 1990, he has supported the Air Force Research Laboratory (AFRL), Space Vehicles Directorate (AFRL/RV), Kirtland Air Force Base, NM, USA, including its precursor organizations (Weapons Laboratory, 1990-1991, and Phillips Laboratory, 1991-1998), in a number of capacities. He is currently technical advisor to the AFRL Space Electronics Branch (Space Vehicles Directorate) and an AFRL Fellow since 2008. He has lead over 100 in-house and contract research efforts involving two- and three-dimensional advanced packaging, radiation-hardened microelectronics, and scalable, reconfigurable computational and systems architectures, with recent emphasis on modularity and the rapid formation of complex systems. He has authored over 100 publications (journal and conference papers, book chapters, and technical reports), four receiving best paper awards, and he has been awarded 11 U.S. patents. Dr. Lyke is an Associate Fellow of the American Institute of Aeronautics and Astronautics (AIAA) and serves on the AIAA Computer Systems Technical Committee. He was selected as recipient of the Federal Laboratory Consortium award for Excellence in Technology Transfer in 1992 and twice for the U.S. Air Force Science and Engineering Award in Exploratory and Advanced Technology Development (1997 and 2000).