Calhoun: The NPS Institutional Archive

Faculty and Researcher Publications   Faculty and Researcher Publications

2001

Naval Postgraduate School PANSAT: Lessons Learned

Sakoda, D.

http://hdl.handle.net/10945/46465

NAVAL POSTGRADUATE SCHOOL PANSAT:
LESSONS LEARNED

D. Sakoda *, Aerospace Engineer
R. Phelps, Electronics Engineer
J. Horning, Computer Engineer
D. Rigmaiden, Electronics Technician
and J. Damerau, Ens., German Armed Forces

Naval Postgraduate School
Space Systems Academic Group
777 Dyer Rd., Rm. 125, Code (SP)
Monterey, CA 93943

ABSTRACT

The Petite Amateur Navy Satellite (PANSAT) was launched aboard the STS-95 Discovery Shuttle on 29 October 1998. PANSAT was inserted into a circular, low-Earth orbit at an altitude of 550 km and 28.45° inclination on 30 October 1998. PANSAT continues to operate and support the educational mission at NPS even after reaching its two-year design life. The research aspect also continues with the analysis of the accumulated telemetry data, in terms of how well the spacecraft operated over the mission design life. However, the store-and-forward mission using direct sequence spread spectrum was never realized.

This paper describes the successes of the PANSAT project in developing a functional, low-cost, small satellite; and the lessons learned in designing largely with industrial components. Specifically, the use of commercial, off-the-shelf (COTS) nickel-cadmium batteries, sparse, yet judicious use of radiation-tolerant devices, error-detection-and-correction (EDAC) system RAM, and software design issues are discussed. The inability to realize the spread spectrum communications aspect is also presented.

Introduction

Officer students, faculty, and staff at the Naval Postgraduate School (NPS) designed, developed, launched, and are operating a small satellite, the Petite Amateur Navy Satellite (PANSAT). The main objective

---

of the satellite is to provide hands-on educational experience for the officer students. Two curricula in Space Systems are offered at NPS, Space Systems Engineering and Space Systems Operations[1]. Since the first conceptual study of the spacecraft processor in March 1989[2], more than fifty graduate theses were published covering various topics.

A second objective was to demonstrate a low-cost, digital communications satellite using commercial, off-the-shelf components. Largely due to cost considerations, PANSAT was designed without space-rated components, such as batteries and processor. The satellite is also a "single-string" system with a number of single-points-of-failure in the design. Design trades were made, cost for reliability, and simplicity for capability. It was understood that PANSAT would be an experimental satellite and that system reboots would have to be accommodated, either due to environmental effects or possible software errors (onboard or up-loaded).

Another objective of PANSAT was to provide digital communications using spread spectrum modulation techniques offering a store-and-forward service to the amateur radio community. Although successful communications were made with the satellite to relay telemetry and upload software, spread spectrum communications were never performed with the satellite.

Space Systems Engineering

The Space Systems Engineering curriculum at NPS is a nine-quarter program with a core matrix of engineering, math, and science courses, as well as some management courses. Officer students receive a Master of Science in one of five disciplines: Astronautical Engineering, Computer Science, Electrical and Com-

1

puter Engineering, Mechanical Engineering, and Physics. In addition to completion of the core courses, specialization sequences are required to meet the requirements of the specific discipline. A written Master's thesis is also required of all graduates.

Space Systems Operations

The Space Systems Operations curriculum at NPS is an eight-quarter program with a broad technical overview in space sciences and engineering. The focus is more on the operations of space systems with an emphasis on information technology, operations analysis, and acquisition management. As with the engineering curriculum, a Master's thesis is required. Officer students in the Space Systems Operations curriculum receive a Master of Science in Systems Technology.

Spacecraft Description

PANSAT was designed with simplicity in mind. With the exception of three microswitches used as inhibit switches, there are no moving parts. The spacecraft was designed as a tumbler with no attitude control. This meant that the antenna would need to be omnidirectional, and the body-mounted solar panels would need to cover the spacecraft so that at any

orientation, energy conversion can occur. The space-craft is basically composed of solar panels, electronics, and the aluminum structure. Figure 1 shows the PANSAT system block diagram with the radio frequency (RF) section on the left, the digital control electronics in the middle and the electrical power subsystem (EPS) on the right. Although PANSAT uses two nickel-cadmium (NiCd) batteries, they are not redundant. Two batteries are required to allow for the number of charge/discharge cycles over the desired two-year mission life. PANSAT uses both silicon solar cells and gallium-arsenide (GaAs) solar cells. One GaAs solar panel is located on the launch vehicle interface where a small user volume was available. The higher efficiency GaAs cells were necessary because of the limited area, but still provides much less energy than the larger silicon cell panels. The digital control subsystem (DCS) is built around a military version, M80C186XL microprocessor and performs all the command and data handling functions for the satellite. Communications was designed for a simplex (or half-duplex) channel to minimize frequency utilization. PANSAT operates in the 70 cm UHF amateur radio band, centered at 436.5 MHz. A more complete discussion of the PANSAT subsystems is given by the author[3].
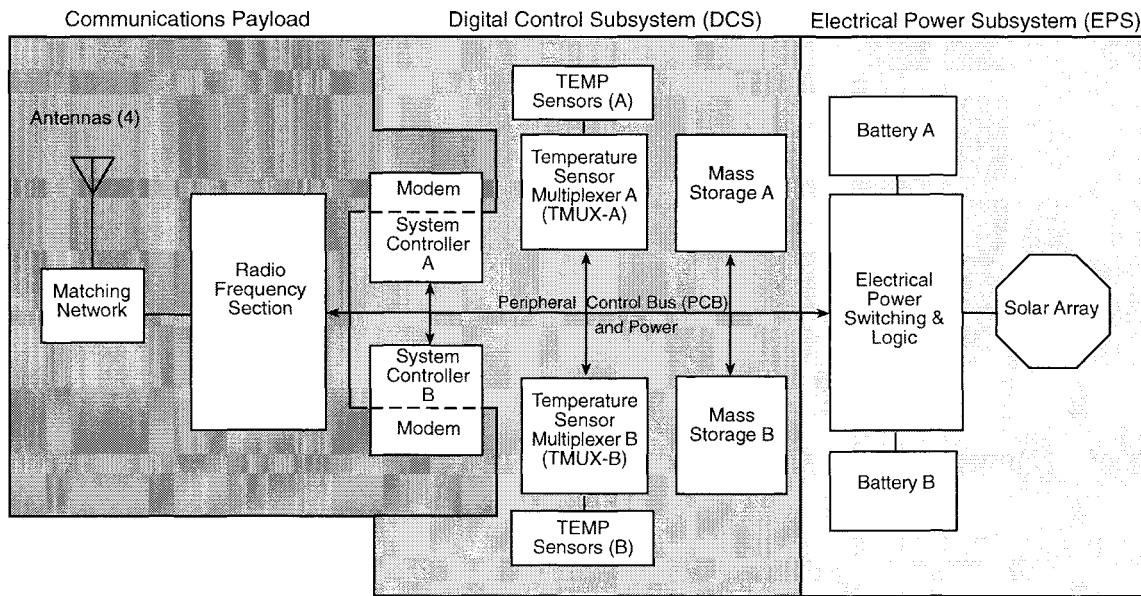

Figure 1. PANSAT System Block Diagram.

The PANSAT spacecraft was designed to fly as a secondary payload on the Shuttle as well as expendable launch vehicles. Designing for the Shuttle presented a number of challenges which were addressed[4]. Figure 2 shows a view of the PANSAT physical layout. Note that some solar panels have been removed to allow for clarity in the illustration.

Radiation Considerations

Initial PANSAT design requirements specified the spacecraft be built using radiation hardened electronics. Once it was determined that meeting this requirement would be prohibitively expensive, our focus turned to
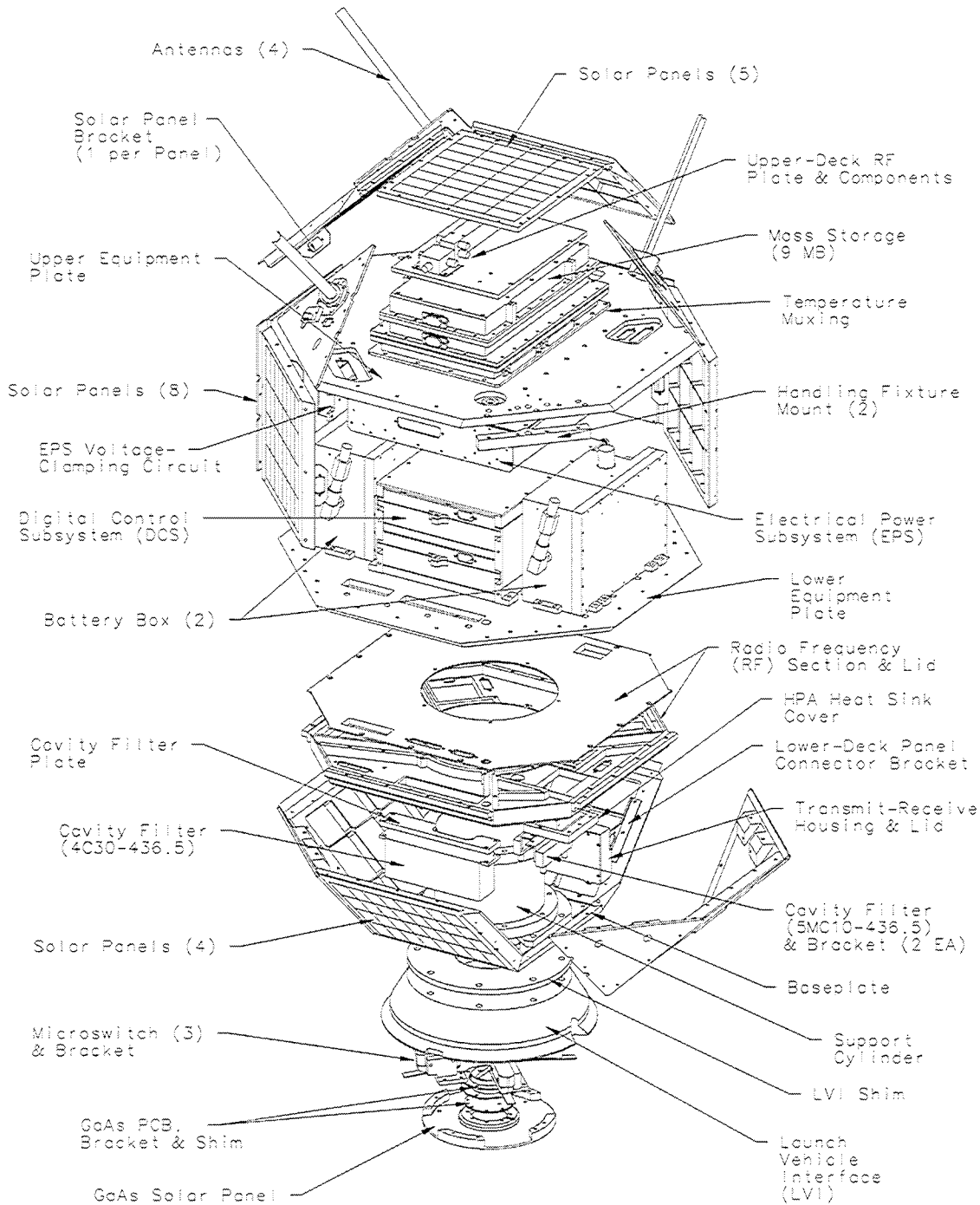

Figure 2. PANSAT Physical Layout.

3

American Institute of Aeronautics and Astronautics

identifying portions of the satellite where hardening against single event effects (SEE) was necessary for reliability and then also focused on operations methods for mitigating the radiation effects.

Given the two-year mission duration and a 550 km orbit, an evaluation of the radiation environment determined that radiation total dose effects were not a concern, leaving only single event effects such as upsets, transients and latchups as the only radiation effects to be addressed in the satellite design.

The approach to mitigating SEE's proceeded as follows. A radiation hardened Error Detection and Correction circuit was designed using the Harris (now Intersil) ACS630MS EDAC chip. The EDAC system can correct single bit errors (1-bit flip in a byte) in system RAM and detect two-bit errors. In the case of a two-bit error which is uncorrectable, the system controller will stop resetting a watchdog (WDOG) counter in the electrical power system. After approximately 90 seconds, the watchdog counter will power down the system controller with the corrupted memory and power on the backup system controller. The EPS will continue to power on the alternate system controller each time an uncorrectable error occurs.

In addition to bit flips in system RAM, it is possible for the processor and support chips to latchup. When latchup occurs, either communications with the EPS via the peripheral control bus (PCB) can be disrupted or the processor can go into a loop and fail to reset the WDOG counter. Again the response from the EPS when the WDOG timer times out is to power on the alternate SC.

The electrical power system is simple in design. It has no processor and the only digital components are large-scale integration circuits (LSI) used in the PCB interface, controlling power switching, the WDOG function, and selecting analog multiplexer channels. LSI digital logic circuits are less susceptible to SEE's than system RAM and the SC processor. Susceptibility is a function of transistor feature size and LSI logic has larger transistor features than the SC processor. Large enough, in fact that the EPS has not experienced any latchups or failures, to our knowledge.

Throughout the satellite are analog circuits used to multiplex, amplify and condition telemetry signals. These analog circuits are susceptible to single event transients, which add error to a signal. Most of the transients are of short duration and are averaged out; but on occasion, some transients are of very long duration. It was fortunate for PANSAT that the long duration transients have not affected operations.

Future satellite designs will need better methods of mitigating long duration transients in telemetry measurements. Latchup failures of digital circuits can be minimized by identifying components where radiation hardened versions would increase system reliability. In addition, radiation reports for COTS components are available on NASA, JPL and other web-sites that will help identify some of the better COTS components with regard to their tolerance to SEE's. Finally, as with PANSAT, software methods will need to be employed to minimize SEE effects.

Error-Detection-and-Correction (EDAC) Memory

The PANSAT spacecraft controller contains an Intel M80C186XL microprocessor. This microprocessor uses 64 kbytes of ROM and 512 kbytes of RAM. Error detection and correction circuitry provides single-bit error correction with dual-bit error detection for all of the microprocessor-addressable RAM. The error detection and correction is accomplished using a Hamming code to generate a check word (6 bits) for each data word stored in memory. Using a sixteen-bit data word requires six check bits. Thus, there are actually 768 kbytes of RAM implemented with three RAM devices where one device contains 256 kbytes used for the check bits.

The memory system operates as follows. When a data word is stored in memory, the associated check bits are also stored (after being generated). During a memory read operation the data word and corresponding check bits are retrieved from memory. New check bits based on the data word from the stored memory are generated and compared with the stored check word. If the two sets of check bits are identical, the data word is assumed correct. Correctable errors (one bit) are identified and corrected and the microprocessor is notified via an interrupt. Uncorrectable data words which are detected notify the microprocessor via an interrupt as well.

The PANSAT EDAC circuitry was first designed by Oechsel[5] and then modified and implemented for the actual flight processor by Horning[6]. The design is a memory bus controller using a commercially available EDAC IC, the ACS630MS. The controller implements a sequential state machine to generate the required control signals for the RAM (Mosaic MSM-8256 static RAM), provides transceivers and latches to isolate the RAM data bus from the microprocessor local bus, and coordinates the operations of the EDAC IC.

The EDAC system was implemented with a 16-bit memory system to reduce the number of RAM devices required for storing the check bits. This complicated the design but the trade off for reduced power was necessary. The M80C186XL is a 16-bit microprocessor, compatible with the 80186. The 16-bit bus allows both byte (eight bit) and word (16 bit) memory opera-

American Institute of Aeronautics and Astronautics

tions. This multi-mode memory interface requires both byte and word memory accesses. However, all data in RAM are stored as 16-bit words. Thus, a byte operation requires the adjacent byte to be accessed as well since the check bits are for 16-bit words only. Implementation of the EDAC system to allow byte-wide data accesses without needing to work with the adjacent byte would require 10 check bits, five for the lower eight bits and five more for the upper eight bits. This would require two RAM devices for the check bits. The state machine logic is simpler but the power saved from removing one RAM device more than compensates for the design complexity.

In addition, the write back control to RAM was designed to reduce the number of writes to RAM. The straightforward design requires every word accessed from memory to be written back to memory, regardless if there is an error or not. This insures that the memory controller automatically corrects the error in the data word that is sent to the CPU as well as the original data word in RAM. This write back occurs within the time envelope of the CPU read and write cycles and does not affect the timing. However, writing back after every memory access does increase the power consumed substantially since correctable errors seldom occur (e.g. about one per day). The write back control logic was designed to only write back when there is a correctable error.

Of all the ICs for the EDAC circuitry and the microprocessor only the ACS630MS is a radiation hardened device. All other circuitry (state machine logic, bus transceivers, microprocessor, etc.) are MIL-STD-883, non radiation hardened components. However, the PANSAT spacecraft controller computer board contains several ICs which are neither radiation hardened nor MIL-STD-883. All components are of at least industrial grade (e.g. -40° C to +85° C) quality.

Commercial Off-the-Shelf (COTS) Batteries

NiCd batteries have historically been a popular secondary power source for small satellite designs. A properly designed battery where cell capacities are matched and the proper charging scheme is utilized can result in a battery that will last 3000 cycles or more. Initial design requirements for PANSAT specified two NiCd batteries. This requirement was driven by a desire for redundancy. As the design progressed, it become apparent that two batteries were needed, not for redundancy, but to enable the satellite to meet the mission design life. For PANSAT's low earth orbit there are approximately 16 eclipses every day. For a satellite with one battery this would equate to approximately 2900 cycles in 6 months time, which is nearly

equivalent to the mission design goal. To lower the number of cycles per day, the batteries are operated as follows. One battery will charge for 4 orbits and the other battery will supply power in eclipse. Each eclipse depletes a battery capacity by approximately 10%. Operating the batteries in this method equates to operating each battery to a 40% depth of discharge before it is recharged. As a result battery cycles were reduced to two per day per battery.

To date, the NiCd batteries have performed well. Both batteries met the design mission life requirements of 2 years and are now in their third year of operation. Battery B is presently showing signs of wear and has had some cells short. The shorts subsequently burned through and the battery is working again though at a moderately reduced capacity. Battery A has performed well but a slightly reduced capacity shows it is wearing out. No answer is available for the difference in how each battery has performed.

Some difficulties in working with NiCd cells were the low energy density of the NiCd technology. Nine D-size cylindrical cells were used in each battery. After Shuttle-required safety measures were added to the battery design and combined with the difficulty of packaging cylindrical cells, the completed battery was large and heavy.

Software Design Issues

The goal of the embedded software, as implemented in ROM, for PANSAT is to provide simple and reliable control of the spacecraft in order to upload higher-level layers of software which are modifiable from a ground station. PANSAT ROM software is written almost entirely in C with some 80186 assembler for critical code and the initial bootstrap software. The ROM software is capable of autonomously operating the spacecraft while listening for commands from a ground station to either downlink telemetry or uplink new data or software.

Although PANSAT is successfully operating non-volatile Flash ROM, which is used to store telemetry, the ROM software was designed assuming this experimental storage might not function correctly. Thus, lacking any permanent memory, the software is designed to start up each time as if the spacecraft were just ejected. However, this software is capable of examining telemetry in Flash as well as the status of spacecraft hardware to determine if indeed the spacecraft may have been operating in orbit prior to the current restart. PANSAT experiences regular restarts which cause one operating system controller computer to shut down and the alternate one to startup. This phenomena is discussed later in this paper.

5

The spacecraft software bootstrap consists of hardware initialization (including a check to EDAC and resetting RAM to known values so as not to create EDAC errors when accessing RAM). The ROM data is relocated, floating point software emulation is setup (there is no floating point capability in the 80186 nor is PANSAT flying a coprocessor), and a C runtime environment is established for the remainder of the software to operate.

The main software consists of a master loop which does not operate concurrent software tasks. This simplified the design of the operating system, increased the complexity of individual components of the master loop (because of timing), and yet increased the reliability of the reset of the watchdog timer in the EPS. All software components of the master loop must operate correctly before the section of code which manipulates the watchdog timer. The master loop regularly performs the following. A/D conversion and telemetry saving begins the loop. Then, an iteration of the battery charge monitor is performed which maintains correct charge, discharge, and operation of the two batteries. The communication system is then checked for possible incoming data packets. Incoming data can occur at any time and is implemented using interrupt service routines and direct memory access. If incoming packets are present they are examined and processed. If a response from the spacecraft is needed then download packets are prepared and packet download begins. As in the uplink, direct memory access and interrupt service routines handle the download of packets invisibly for the master loop. The software then checks telemetry for anomalies and applies corrections if applicable. And finally, before the loop repeats, the watchdog timer in the EPS is reset. As in the communication code, the update of the system clock and maintenance of the EDAC were implemented with interrupt service routines.

Spacecraft Telemetry Analysis

EDAC Errors

As described earlier, one bit error correction and two bit error detection of the entire 512 kbytes of RAM available to the system controller microprocessor is implemented with the EDAC circuitry. Although single bit errors are automatically corrected upon reading or writing either the byte or word from or to RAM, there is no guarantee that the operating software will access all RAM locations within the time needed to ensure that a single bit error is corrected. In order to ensure that all

RAM locations are accessed regularly a RAM wash routine is run regularly by the system software.

As implemented in the PANSAT ROM, the RAM wash software operates as follows. The entire 512 kbytes of RAM are separated into 8192 blocks of 64 bytes each. A block wash is performed every 500 msec. Thus, the entire 512 kbytes are washed every 4096 seconds (68.27 minutes). An estimation of the number of SEUs expected in RAM was performed by Oechsel[5] based upon data from UoSAT-2, a microsatellite similar to PANSAT regarding its orbit and types of electronic circuits. Conservative assumptions indicated that the number of SEUs expected was to be 1.0×10^{-6} SEU/bit/orbit. This equates to an expected time between uncorrectable errors of 1.8 years assuming the RAM wash occurs once per orbit. The orbital period of PANSAT is 5739 seconds; and thus 4096 seconds per RAM wash is suitable.

Single bit errors cause a hardware interrupt to trigger an interrupt service routine which records the RAM location (byte or word address) where the error occurred within and the date and time of service. PANSAT experiences on average about one detectable single bit error in its RAM every day. At most, there have been four bit errors in one day; however, more than a single bit error per day is rare.

Since 4096 seconds RAM wash cycles are about two thirds of one orbit, using the date and time of the single bit error interrupt is useless when attempting to determine PANSAT's position over the Earth. After launch it was decided to modify the RAM wash software to implement RAM wash cycles at much higher frequencies. The software was changed to wash a block every 1/60 of a second. Thus, the entire RAM could be washed in 136.5 seconds. With an orbital period of 5739 seconds, 136.5 seconds is 0.149 rad (8.56°) which corresponds to a distance of 1035 km. This is the maximum error of location of the spacecraft at the time of the single bit error; and, on average, the error in distance is about 516 km. Using this more precise date and time of single bit error correction, the approximate location of the spacecraft was determined. Sixty-three data points were used with RAM wash cycles of 4096 seconds to produce Figure 3. Approximately 90% of the single bit error locations were near the region of the South Atlantic Anomaly[†] (SAA).

[†] The South Atlantic Anomaly is a local distortion of the geomagnetic field causing a high average of radiation level in a large area of the South Atlantic. This area is approximated at -22.5° to -24.5° latitude and 312.5° to 317.5° east longitude.
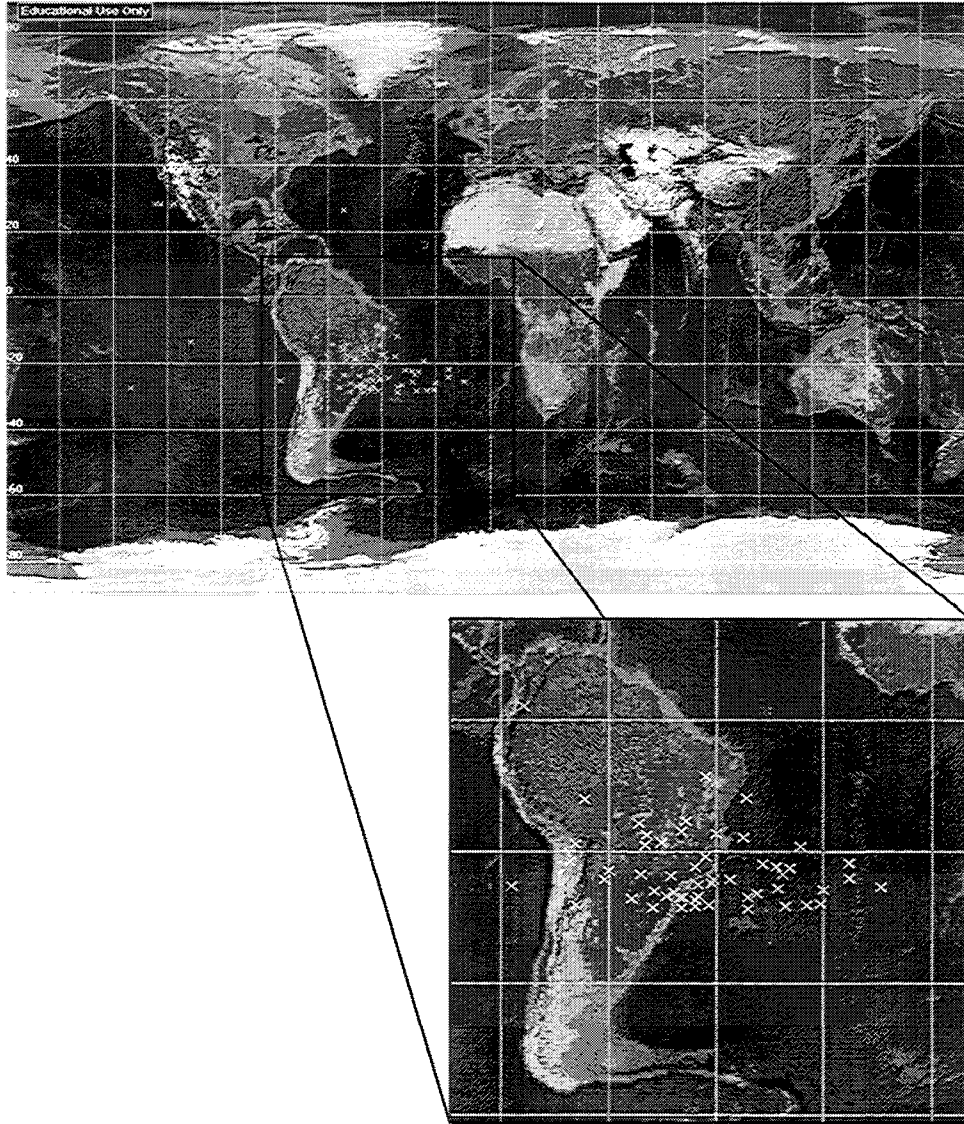
Figure 3. EDAC Errors Using Fast RAM Wash.

Spacecraft Restarts

PANSAT has experienced frequent computer restarts since launch. On average the spacecraft system controller operates continuously for about three days before it is restarted. The longest continuous operation of a system controller on PANSAT is 30 days. Because the elapsed time of a system controller is reset to zero each time the microprocessor is restarted, the spacecraft restart times can be translated to UTC. A restart implies that the spacecraft reset. The exact reasons for resets are not known. However, indirect implications due to the state of the spacecraft prior to reset and just after restart often indicate why a reset would have occurred. A system controller can cause itself to be shut down by failing to update a watchdog timer in the power system.

This timer will cause the operating system controller to be powered off and the other system controller to be powered on.

In general, there are several probable causes for a restart[7]. These are: transmission, software upload failure, commanded by NPS, communication state change, A/D failure, power glitch, or unknown software failure. Resets related to transmission are likely when the spacecraft is operating on battery power only and the battery is low in capacity. During transmission the spacecraft can use over 1.5 Amps of current and the switching peaks are over 2 Amps. Software upload failures are an operational problem from the NPS ground station. NPS can command the spacecraft to switch to the other system controller which will cause a restart. A communication state change can cause the

7

operating system controller to power down and thus allow the other to begin operating. This can occur after all communication possibilities with a modem and RF unit have been attempted without successful communication with NPS. A/D failures are possible with the calibration or sample buffer operation. When these are sensed the operating system controller will cause itself to shutdown. Power glitches of unknown reasons can cause the operating system controller to reset (not necessarily shut down). And, unknown software errors may cause a restart.

Transmission problems and command by NPS are the reasons most often determined to cause the restarts. Reducing transmissions while the spacecraft is in eclipse and the batteries are of low capacity has been achieved by modifying operations at the NPS ground station. NPS commanding the restart is purely experimental and has been minimized.

The other probable, but less likely causes are still of a concern; however, there is little that can be done to reduce these causes in the ROM software. Uploaded software images have reduced the possibility of the A/D failures and communication state changes by masking off the A/D errors and increasing the time required to make a communication state change. Regardless, each restart begins with operating ROM software.

Spread Spectrum Mode of Communications

The PANSAT digital modem contains a PA-100 spread spectrum demodulator chip first produced by Unisys Corporation. The PA-100 provides for flexible data rates, modulation types, processing gains, PN codes and loop control. External circuitry provides for clock generation, differential coding, automatic gain control and 70 MHz intermediate frequency (IF) generation. The 70 MHz transmit and receive IF signals connect to the RF section which contains the local oscillators, low noise amplifiers, high power amplifiers and power controller. The RF section feeds a matching network that drives the tangential turnstile antenna mounted on PANSAT to provide an omnidirectional pattern with nulls not to exceed 3dBi.

PANSAT is designed to provide half-duplex digital communications with a bit error rate not to exceed one error in one hundred thousand. The required signal to noise ratio was measured using a Noise Com precision carrier to noise (C/N) generator to set the signal level and a Hewlett Packard data error analyzer to verify the bit error rate. The measured C/N was used to verify the link budget calculations used to design the communications system. The link budget provides for a minimum margin of 3dB under worst-case conditions.

The modem was designed to operate in two different modes. The first is a 78k-bit narrow band mode used for telemetry downloads and program uploads. The second is a 9.8k-bit wide band mode used for spread spectrum store and forward messaging. In both cases the PA-100 modem chip requires software table downloads to setup, acquire, and track the incoming signal. The chip also requires constant monitoring for signal detection along with table updates to maintain a reliable link. The software overhead was also increased by the half-duplex communications link that PANSAT uses. During spacecraft integration it was determined that the final software to provide the spread spectrum mode of operation was not going to be ready in time to meet the launch schedule. At that time the narrow band mode was running so it was decided the required software could be uploaded after on orbit operations began.

PANSAT was launched as a secondary payload utilizing a cylindrical carrier, with spring eject mechanism. When PANSAT was integrated into the carrier the four antenna blades came into contact with the inside surface of the can. NASA required reforming the antenna blades to maintain the correct payload clearance. This operation, however, changed the radiation pattern of the antenna system to an unknown.

Once on orbit operations began, the narrow band mode proved to be reliable but unpredictable due to the antenna issue. Several modifications to the ground station antenna and amplifiers increased system performance but slow program uploads and short intervals between resets prevented spread spectrum operation.

Conclusions

The PANSAT small satellite is the first autonomous spacecraft developed by officer students at the Naval Postgraduate School. A number of challenges were overcome and lessons learned which will be addressed with the follow-on project to PANSAT. Although the spread spectrum aspect failed to materialize, the overall design and engineering trades successfully validated the overall design of the spacecraft, specifically, the use of commercial-off-the-shelf (COTS) components in a low-cost, single-string spacecraft design.

American Institute of Aeronautics and Astronautics

References

1. "Naval Postgraduate School Graduate Education in Space Systems Through Space Flight Experience," D. Sakoda, Proceedings of the 5th International Symposium on Small Satellites Systems and Services, La Baule, France, June 19-23, 2000.

2. *Design of a Reliable Computing System for the Petite Amateur Navy Satellite (PANSAT),* J. K. Hiser, Master of Science in Electrical Engineering Thesis, Naval Postgraduate School, Monterey, California, March 1989.

3. "Overview of the Naval Postgraduate School Petite Amateur Navy Satellite (PANSAT)," D. Sakoda, Thirteenth Annual AIAA/Utah State University Conference on Small Satellites, Paper SSC99-I-5, Logan, Utah, Aug. 1999.

4. "The Petite Amateur Navy Satellite (PANSAT) Hitchhiker Ejectable," D. Sakoda, 1999 Shuttle Small Payloads Symposium, NASA/CP-1999-209476, NASA/GSFC, Greenbelt, MD. Sept. 1999.

5. *Implementation of Error Detection And Correction (EDAC) In the Static Random Access Memory (SRAM) Aboard Petite Amateur Navy Satellite (PANSAT),* C. R. Oechsel, Master of Science in Electrical Engineering Thesis, Naval Postgraduate School, Monterey, California, March 1995.

6. *System Controller Hardware and Embedded Software for the Petite Amateur Navy Satellite (PANSAT),* J. A. Horning, Master of Science in Electrical Engineering Thesis, Naval Postgraduate School, Monterey, California, September 1997.

7. *Correlating Space Environment Phenomena and Satellite Operations to System Controller Resets in the Petite Amateur Navy Satellite - PANSAT,* J. Damerau, Special Topic Report, Naval Postgraduate School, Monterey, California, December 2000.