Rapid Space Hardware Development through Computer-Automated Testing

Dallas S. Masters and Kimberly K. Ruud
Nonproliferation and International Security Division
Los Alamos National Laboratory
NIS-4, MS D448
Los Alamos, NM 87545
(505) 665-8018, 665-5134
dallas@lanl.gov, kruud@lanl.gov

Abstract. FORTÉ, the Fast On-Orbit Recording of Transient Events small satellite designed and built by Los Alamos and Sandia National Laboratories, is scheduled for launch in August, 1997. In the spirit of "better, cheaper, faster" satellites, the RF experiment hardware (receiver and trigger sub-systems) necessitated rapid prototype testing and characterization in the development of space-flight components. This was accomplished with the assembly of engineering model hardware prior to construction of flight hardware and the design of component-specific, PC-based software control libraries. Using the LabVIEW® graphical programming language, together with off-the-shelf PC digital I/O and GPIB interface cards, hardware control and complete automation of test equipment was possible from one PC. Because the receiver and trigger sub-systems employed complex functions for signal discrimination and transient detection, thorough validation of all functions and illumination of any faults were priorities. These methods were successful in accelerating the development and characterization of space-flight components prior to integration and allowed more complete data to be gathered than could have been accomplished without automation. Additionally, automated control of input signal sources was carried over from bench-level to system-level with the use of a networked Linux workstation utilizing a GPIB interface.

Introduction

FORTÉ Overview

Los Alamos National Laboratory (LANL) and Sandia National Laboratories (SNL) are developing for space flight, FORTÉ (Fast On-orbit Recording of Transient Events), an advanced radio frequency (RF) impulse detection and characterization experiment. Emphasis is on the measurement of electromagnetic pulses (EMP), primarily due to lightning, within a noise environment dominated by continuous wave (CW) carriers, such as TV and FM stations. Optical sensors will augment the RF system in characterizing lightning events. A principal goal is to develop a comprehensive understanding of the correlation between the optical flash and the very high frequency (VHF) emissions from lightning.

To help meet FORTÉ's low cost objective, development of the RF experiment hardware necessitated rapid prototype and engineering model (EM) validation and comprehensive testing, prior to and concurrent with, flight model fabrication. Early in the project, a path was chosen to use an inexpensive, off-the-shelf PC test and control system (TCS) for control of the RF hardware during bench-level testing. The TCS would also communicate with a suite of instruments via a GPIB interface, providing control of test input signals and high-speed data acquisition. Computer-automated tests would accelerate progression to flight hardware and allow more thorough testing and data archiving than could be reasonably achieved with conventional testing methods.

RF Experiment Payload

The FORTÉ RF experiment is designed to detect, discriminate, and record transient signals embedded in the noisy VHF spectrum and correlate these events with their optical signatures. To accomplish this, the RF system components are designed for maximum sensitivity in the band from 20-300 MHz. The RF experiment hardware consists of four sub-systems: Antennas, Back-Up/Primary Antenna Control, Receiver Box, and Trigger Box.

The antenna system consists of two linearly-polarized, 35-foot, log-periodic arrays (primary antennas) and two short dipoles (secondary antennas). Any combination of the four antennas is selected by the Back-Up/Primary Antenna Control (BUPAC) and connected to the Receiver Box. An antenna switch in the Receiver Box allows the two 20 MHz wide-band receivers (TATRs) and one 90 MHz ultra wide-band receiver (HUMR) to connect to either of the two antenna sources from the BUPAC. The receivers are highly configurable, with multiple tuning modes, two local oscillator (LO) sources, and variable attenuator settings.

The Trigger Box contains three separate detection and discrimination triggering circuits, one for each receiver. These triggering circuits are the heart of the FORTÉ RF experiment, capable of discriminating among a wide array of pulse signatures based on power, spectral, and temporal parameters. A trigger output pulse results in a command to the spacecraft's high-speed data acquisition system (DAS) to record the transient event of interest. The triggering parameters are highly configurable, and thus pose a challenge in testing all possible combinations of triggering modes.

This paper is organized as follows. The RF experiment payload is discussed, with details of the hardware involved in automated testing. The PC-based test and control system for controlling the RF payload hardware and instrument suite is then presented. A discussion of automated testing with the TCS from slice-level to system-level follows, with examples of specific tests conducted. Continued testing of the flight RF hardware on the integrated satellite is also addressed. Finally, a summary of the work is presented.

RF System Hardware Development

Development of the RF hardware was separated into two levels: engineering model (EM) and flight (figure 1). In addition to these levels of hardware, early prototype versions of the critical hardware (TATR and Trigger slices) were fabricated, and the designs verified before EM construction. The EM hardware was identical to the flight hardware in function and differed only in the environmental specifications of some components. EM hardware was fabricated and tested for baseline performance data before fabrication of the flight model. With EM baseline performance achieved, fabrication of flight hardware began concurrently with further testing of the EM model. If any later modifications to the design were necessary, they were first tested on EM hardware before implementing in the flight model. After fabrication, flight hardware was then subjected to the same battery of tests as performed on the EM model. Comparison of the their performance data revealed any differences between the EM and flight hardware, possibly caused by fabrication errors or defective components.

Modular Design Approach

The design approach to the Receiver and Trigger boxes was modular, with six autonomous slices forming each box. The slices contained circuit boards designed to perform one or more specific functions of the box. Housing function specific circuits in individual slices allowed each slice to be tested and characterized individually before integration and simplified debugging in the event of anomalies found at the box or system levels.

After testing individual slices, they were physically integrated to form the Receiver and Trigger boxes. Testing at the box-level repeated some applicable tests performed at the slice-level and added a new set of more complex tests. After the three RF sub-systems were tested at the box-level, they were integrated on the bench for a series of system-level tests. Previous tests were conducted to verify performance continuity, and a new set of system-level tests added. This method of repeating standardized tests at the three levels of hardware integration insured continuity of baseline performance during hardware development.

RF Sub-Systems

The RF experiment hardware developed and tested using the TCS were the BUPAC, Receiver Box, and Trigger Box sub-systems (figure 1). Each of the sub-systems was designed to have similar control interfaces through 8-bit registers. The registers were addressable with select lines and latched with write lines. With the exception of the Trigger Box, all registers were write-only, with the state of the hardware inferred from the last command. The following descriptions give more detail on the design of the hardware and how control of the individual interfaces was achieved.

BUPAC

The BUPAC selectively connects a primary antenna or secondary antenna to the respective input on the Receiver Box. The positions of two double-pole RF relays are controlled by setting two bits of a register shared with the Receiver Box and latching in the values by strobing a write line.

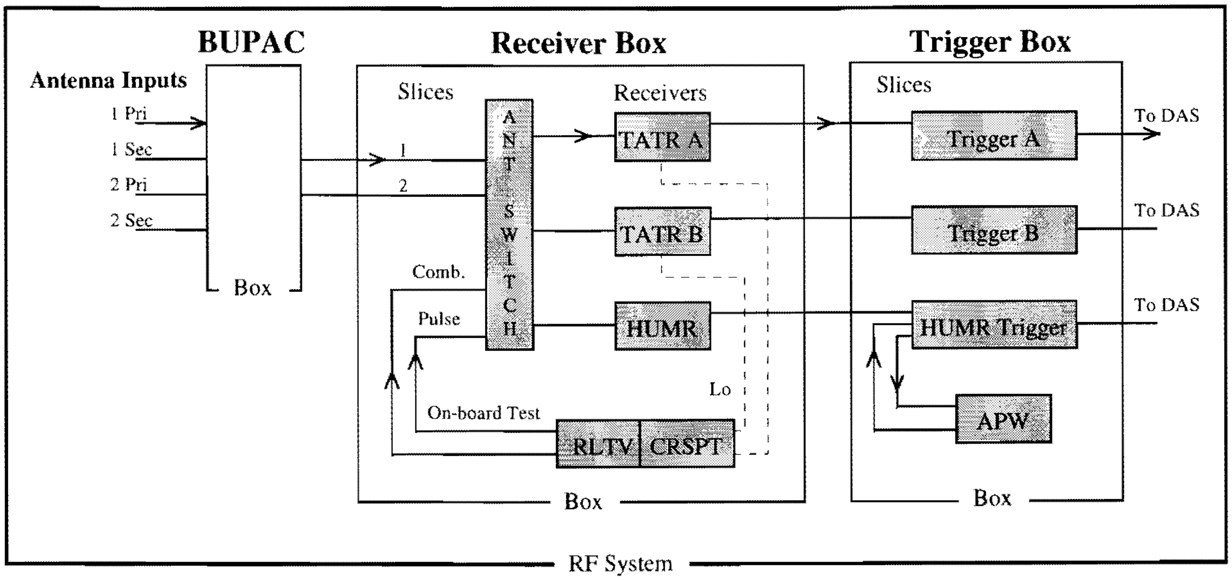


Figure 1. RF payload hardware showing slice, box, and system level components.

Receiver Box

The Receiver Box contains six slices: power supply, Antenna Switch, REF/LO/Test, and three receivers (TATR A, TATR B, and HUMR). The Antenna Switch routes either of the two antenna inputs to the receivers or couples an on-board test signal into the RF stream for on-orbit calibrations. The TATR reference oscillators, relay circuitry to switch between two LO sources, and the on-board test signal are housed in the REF/LO/Test slice. The TATRs are 20 MHz wide-band receivers, tunable from 4-320 MHz in center-frequency steps of 4 or 8 MHz. They employ superheterodyne up converters, with an IF baseband output from 2-22 MHz. Additionally, two attenuators can be adjusted from 0-41 dB and one of two variable LO sources selected for phase coherence measurements. The HUMR receiver preselects one of five possible bands to tune: 0-90 MHz, 110-210 MHz, 210-310 MHz, an all-pass band, and an intermediate 28-80 band. It has an IF baseband output from 0-90 MHz. The HUMR also contains a variable attenuator from 0-63 dB. Control of all Receiver Box slices is achieved via 8-bit registers to each slice, with the receivers having two select lines controlling the tuning parameters and attenuator levels.

Trigger Box

The Trigger Box contains a power supply slice, three separate programmable triggers (Trigger A, Trigger B, and HUMR Trigger), and two slices making up the Adaptive Pre-Whitner (APW). The triggers discriminate between narrowband and broadband events based upon programmable parameters. The APW was fabricated by Sandia National Laboratory and not available for testing with the TCS.

The Receiver Box's TATR A and TATR B outputs are coupled to Trigger A and Trigger B, respectively. Each trigger contains eight, 1 MHz-wide sub-bands from 3.0 -21.5 MHz for which independent threshold levels are set. Trigger coincidence windows, coincidence levels, carrier rejection parameters, and threshold levels are programmed via an 8-bit register. The power level in each sub-band is also measured and digitized on the same 8-bit register. The eight sub-bands each have a 3-bit address, and the different functions are selected by choosing one of three select lines.

The HUMR Trigger is a two channel, programmable level trigger coupled to the RF Box's HUMR and the APW. It has an adjustable trigger threshold level and broadband power monitoring circuits for each channel. Like the narrowband triggers, a single 8-bit register is used to both write threshold settings and read power monitor levels.

RF Test & Control System

The RF Test and Control System (TCS) served two roles in the development of the RF payload hardware: provide an interface for controlling the RF hardware and communicate with a suite of test instruments to facilitate automated testing (figure 2).

Computer

The TCS computer was a Compaq Portable PCIII (486DX/66). A portable computer was selected so the TCS could follow the hardware to off-site environmental and integration testing. The PCIII was chosen over smaller laptop models because it accommodated full-size and half-size DIO and GPIB plug-in cards needed for automated testing.

Digital I/O

For testing individual RF slices, a standard DIO-24 plug-in card by Keithley Metrabyte was used. Its 24 digital input/output lines were grouped into three, 8-bit registers (A, B, and C). Each register was configurable for either read or write, and the three registers made up a single "device". The 24 DIO lines were sufficient to test individual slice hardware during the prototype and EM phases of development.

When box-level testing began, the DIO-24 card was replaced by two National Instruments PC-DIO-96 plug-in cards. Each DIO-96 card contained 96 digital input/output lines and was configured to operate like four separate DIO-24 cards, each with an individual device address. Together, the two DIO-96 cards controlled up to eight devices from the PC. Individual control of the Receiver and Trigger boxes, each containing six slices, was possible, and limited simultaneous control of both boxes provided a means to test the partially integrated system on the bench.

GPIB

PC control of the bench instruments was via a National Instruments AT-GPIB/TNT plug-in card. It accommodated up to 31 GPIB instruments on a single bus and supported the IEEE 488.2 protocol. GPIB instruments were configured and controlled remotely from the PC and provided a means of high-speed data acquisition directly to the computer.

Software

The PC ran Microsoft Windows 95® operating system. Initially, the first EM slices were controlled with DIO driver software written in the C programming language.

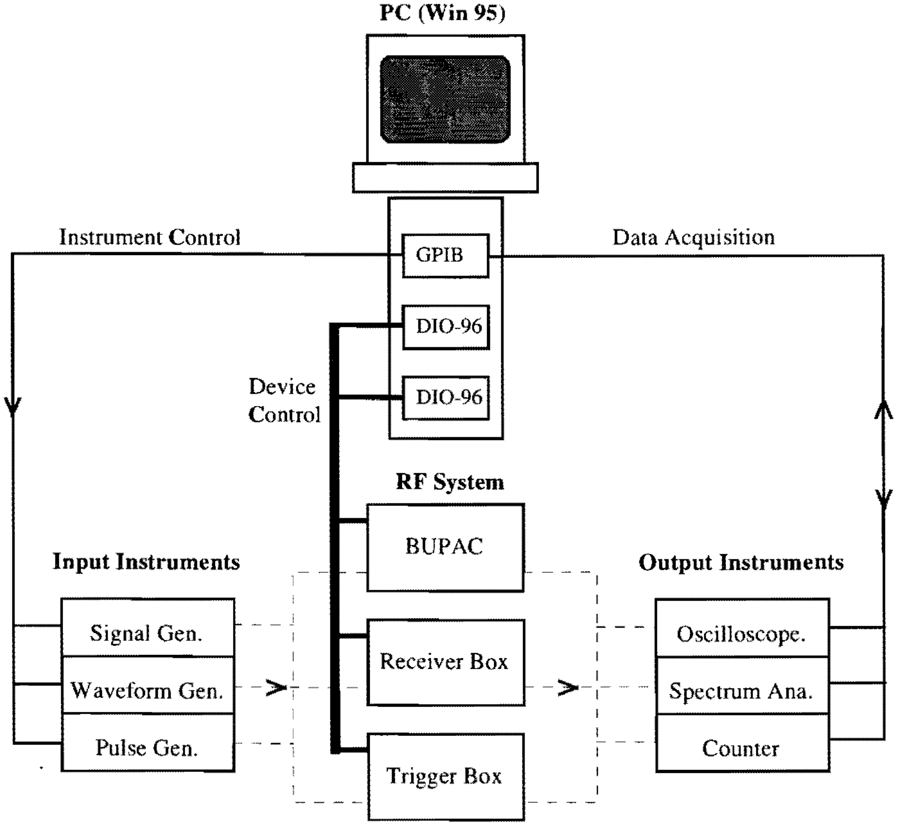


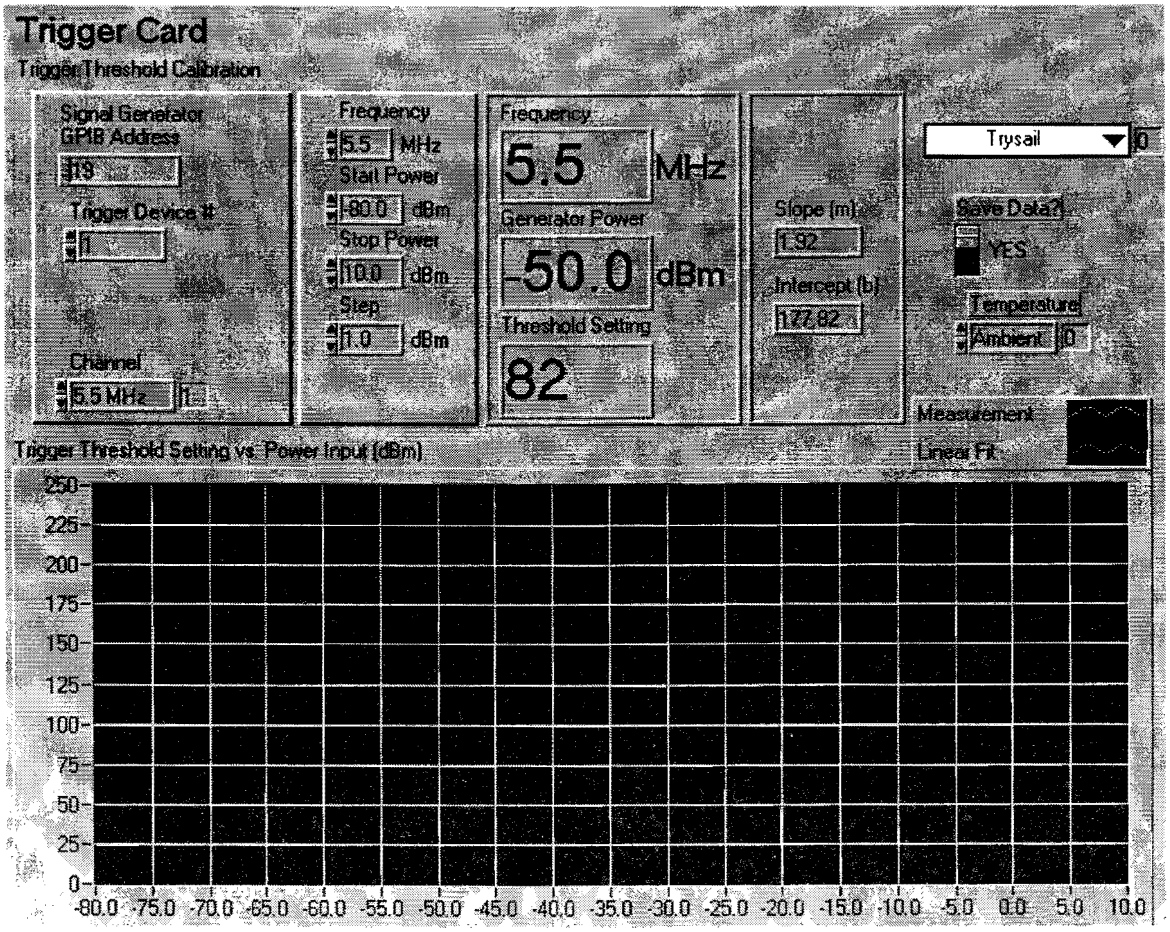
Figure 2. RF Test and Control System (TCS).

Development in C proved cumbersome, and hardware changes were not easily adapted to in software. The use of C was replaced by National Instruments LabVIEW® (Lab Virtual Instrument and Engineering Workbench) as the TCS software.

The LabVIEW® programming language is graphical in nature, with data flow and control governed by a simple wire diagram rather than lines of code. Data structures are similar to those in conventional programming languages, and constructs such as for, while, and if loops are fully implemented. Its proven use in engineering and science laboratories is well documented.[1,2,3]

A program in LabVIEW® consists of two parts: a front panel and a wire diagram. The front panel is a graphical user interface which can be populated with both input and output data structures. These structures take on forms closely associated with real instruments, such as knobs, switches, LCD displays, and graphs (figure 3).

The "code" behind the front panel is contained in the wire diagram. The wire diagram is a graphical schematic for controlling data flow through the program (figure 4). Data and programming structures are linked together with wires that accommodate multiple connections and automatically convert among different data types. Programming in LabVIEW® closely resembles drawing a flow diagram, and VIs are easily debugged or modified to incorporate hardware changes. Its modular organization allows incorporation of VIs into the wire diagrams of higher level VIs when building more complex programs.



Signal Generator GPIB Address: 19
Trigger Device #: 1
Channel: 5.5 MHz 1
Frequency: 5.5 MHz
Start Power: -80.0 dBm
Stop Power: 10.0 dBm
Step: 1.0 dBm
Frequency: 5.5 MHz
Generator Power: -50.0 dBm
Threshold Setting: 82
Slope (m): 1.92
Intercept (b): 177.82
Trysail 0
Save Data?: YES
Temperature: Ambient 0

Figure 3. LabVIEW Front Panel of Trigger Sub-Band Threshold Level Calibration VI.

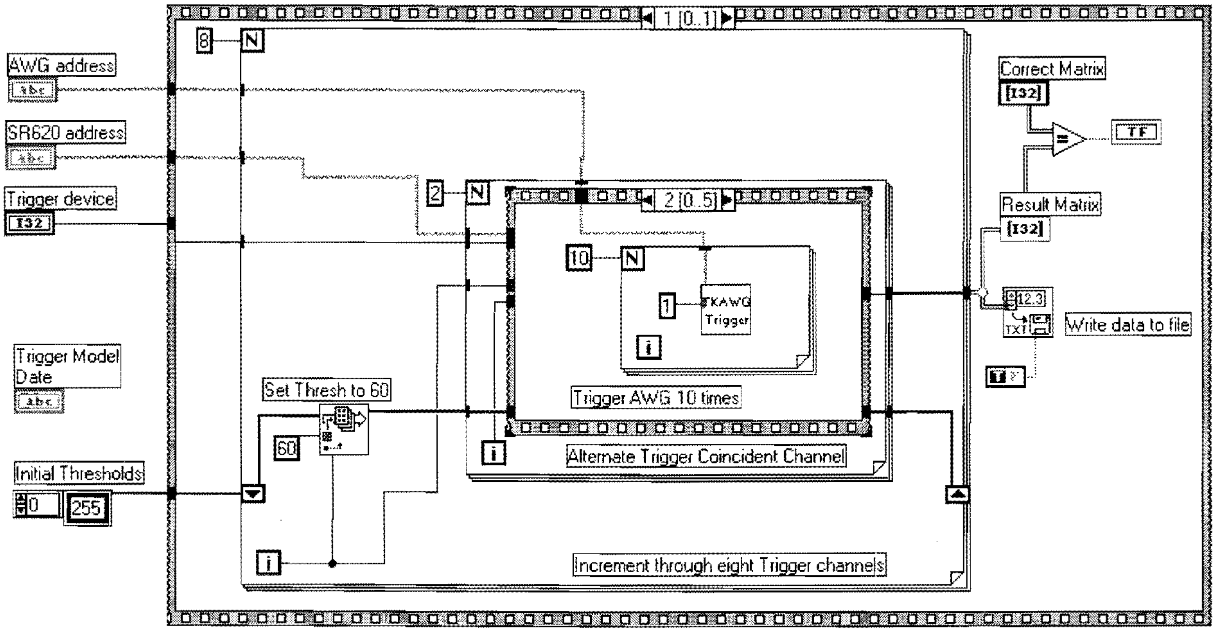


Figure 4. LabVIEW Block Diagram of Coincident Sub-Band Trigger Test VI.

The intuitive LabVIEW® graphical user interface and hierarchical virtual instrument (VI) structure allowed rapid software development and complemented the design of modular hardware. Any changes or redesign in one slice of hardware was adapted to easily within the sub-VI software environment without affecting operation of other hardware. Low-level VIs were reused and grouped into higher-level programs as the slice hardware was integrated at the box and system levels. Although the complexity of the hardware continually increased, the VI architecture allowed automated test procedures to be developed and adapted in short time.

Bench Instruments

A suite of bench instruments was used to produce input signals and measure output characteristics in testing the RF payload hardware. All instruments were GPIB capable and communicated remotely with the TCS computer. Three input sources were used: an HP8665B Signal Generator, Tektronix 2041 Arbitrary Waveform Generator, and Stanford Research Systems DG535 Pulse Generator. At the outputs, three instruments were used to measure and record waveforms and verify proper hardware functionality: an HP8652A Spectrum Analyzer, Tektronix TDS 540 Digital Oscilloscope, and Stanford Research Systems SR620 Counter.

LabVIEW® Virtual Instruments

Control of the bench instruments was made convenient by the availability of pre-written LabVIEW® drivers for the bench instrument suite. These VI drivers, provided by National Instruments and available for download off of their public web site, contained complete libraries of low-level GPIB commands necessary to control the respective instruments. Some high-level commands were implemented in the VI libraries, but customization was made simple by arrangement of the low-level VIs to form automated tests. These instrument VIs were integrated with the RF payload control VIs to form single interfaces to automated test programs.

Automated Testing

Automation of tests was pursued whenever it was deemed feasible and likely to accelerate the progression from slice-level hardware to an integrated payload (figure 5). As noted in earlier work with automated testing[4], upfront investments of development time were later compensated when automation provided a means for fast, repeatable, and comprehensive testing. Additionally, real-time data analysis was incorporated into test programs and offered quick looks at performance information.

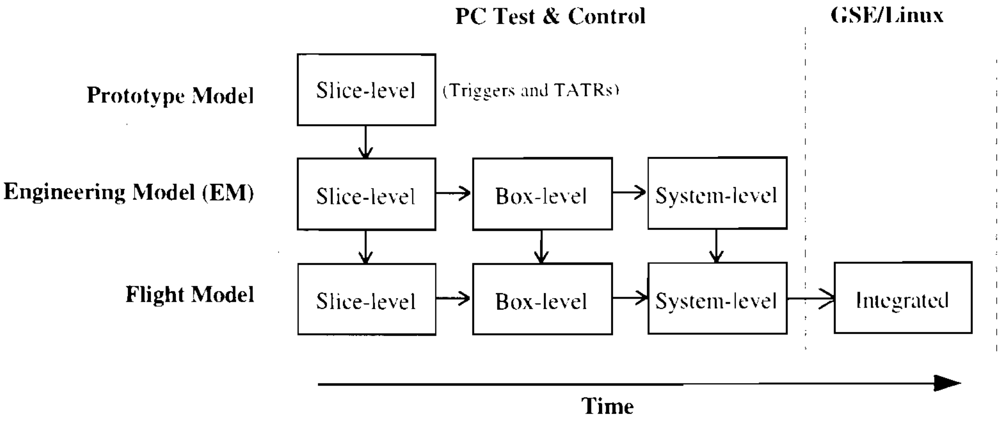


Figure 5. Organization and flow of automated testing sequence.

Slice-Level Testing

Individual hardware slices were tested on the bench using the TCS computer to control the slices via the DIO-24 plug-in card and communicate with the instrument suite via GPIB. Power to the slice was supplied from external bench sources. Tests at this level ranged from verifying relay operation in the Antenna Switch to measuring LO frequencies and power levels at the output of the TATRs. In simple tests such as verifying relay operation, automation was not required. Tests requiring multiple hardware configurations (e.g., LO frequency measurement) or a large range of input frequencies (e.g., Trigger transfer function gain) were automated. Simple loops alternately issued hardware commands, changed input signals, and recorded output measurements. Some slice-level tests, such as the example described below, were repeated at the box and system levels to verify performance had not changed as a result of integration.

Example: Trigger Threshold Calibration

An example of an automated slice-level test was the calibration of the sub-band trigger threshold levels in the Trigger slices. To accurately and rapidly calibrate the 8 sub-band thresholds, a test was designed to stimulate the threshold circuitry with an unmodulated carrier of known power and record the response of the trigger pulse output. For this test, the other triggering parameters were held constant, and only the setting of an individual sub-band threshold level influenced the trigger output.

The test used the HP8665B signal generator as the input source and the SR620 counter to record trigger pulse counts. The automated VI operated in a series of nested loops. The first loop incremented both the sub-band index from 0 through 7 and the HP8665B input frequency to match the sub-band center-frequency. The second loop incremented the HP8665B output power level from -80 dBm to 10 dBm. In the third loop, the sub-band thresholds were incremented continuously from 0 in unit steps until the threshold was above the signal power and the trigger count recorded by the SR620 equaled zero. When this condition was met, the value of the input power level and sub-band threshold were recorded to a file. The results bounded the linear operation range of the trigger amplifiers and yielded the calibration for setting trigger threshold values based on power input in dBm (see Figure 6).

Box-Level Testing

Once the Receiver and Trigger hardware slices were fully tested and characterized on the bench, they were assembled into their respective boxes. To control all six integrated slices from the TCS, two DIO-96 plug-in cards replaced the DIO-24 card. Power to the slices was now supplied from the incorporated box power supply slice. The box-level stage both reproduced tests conducted at the slice-level and introduced new tests to characterize performance of the box as a whole. Tests ranged from verifying the functionality of the Trigger coincidence window length to operating the TATRs simultaneously using different LOs. The automation of this last test illuminated a design flaw that was not apparent in slice-level or other box-level testing. By automating a test to characterize LO crosstalk as a function of the 4489 possible TATR tuning combinations, the crosstalk was found to be in less than 1 percent of the TATR tuning combinations. Without automation of the test, resolving the problem would

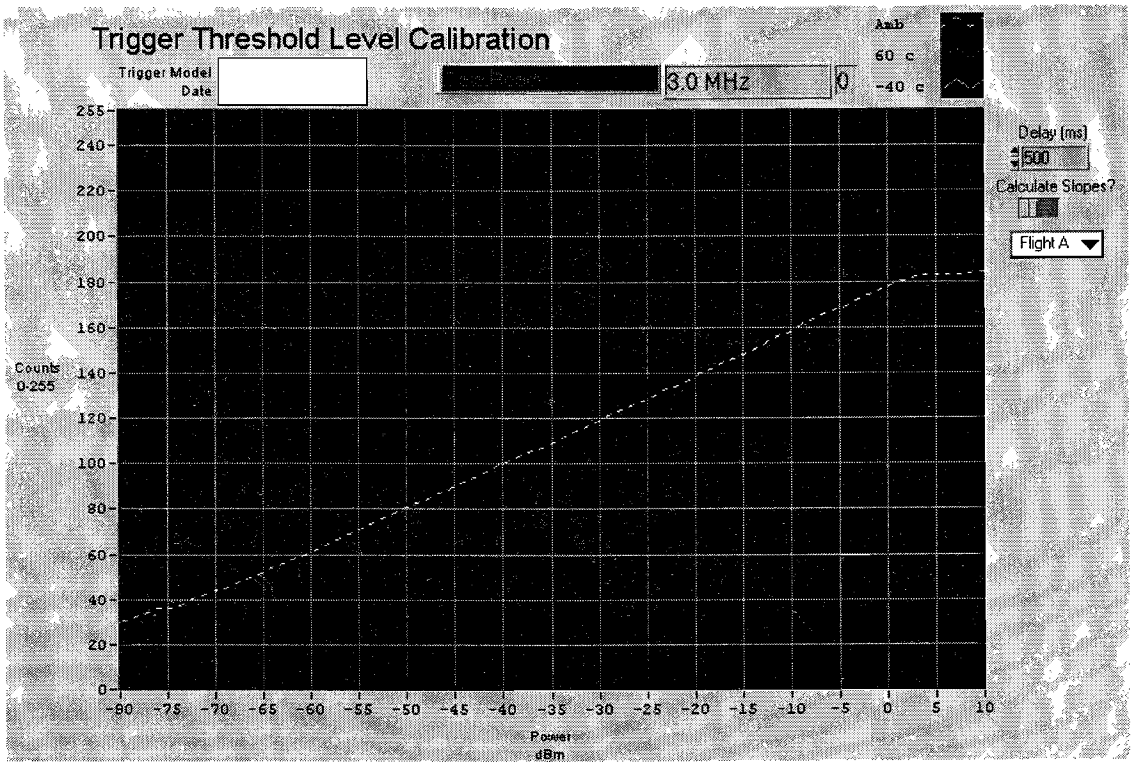


Figure 6. Display Trigger Sub-Band Threshold Level Calibration VI.

have used much more time and disrupted the development schedule.

Example: TATR S21 Gain Characterization

An example of an automated box-level test was the TATR S21 transfer function gain characterization. S21 refers to the two ports across which the gain is measured. For slice-level measurements, this was from the RF input to the IF output on the TATR slice. At the box-level, the signal routing through the Receiver Box now traversed the Antenna Switch and short lengths of connecting cable. Therefore, the box-level S21 gain was measured from the Antenna Switch to the TATR output on the Receiver Box.

The volume of configuration changes needed to measure the TATR S21 gain over all of its possible antenna switch routes, tuning bands, modes and input frequencies lent itself to automation. The TATR tunes input frequencies from 4-320 MHz in 75 bands (4MHz step tuning mode) or 38 bands (8 MHz step tuning mode). To measure the S21 gain of a single TATR for input frequencies spaced 1 MHz apart across all of the bands (assume 30 MHz wide-band for roll-off information) and in both tuning modes, a total of 3390 input configuration changes were required.

The test used the HP8665B signal generator as the input source and the HP8652A spectrum analyzer to measure power output. The automated VI consisted of two stages, a 4 MHz step tuning mode and an 8 MHz step tuning mode. Within each mode, two nested loops operated. The first loop incremented the TATR center tuning frequency from 4-320 MHz by the step amount and opened a file for data logging. The inner loop incremented the signal generator input frequency (constant power) in unit steps and recorded the power output measured by the spectrum analyzer. For each input frequency, the S21 gain was calculated and saved in the file. The results were S21 gain curves for the 113 possible tuning bands of the TATR.

System-Level Testing

After all three sub-systems had completed box-level testing, they were connected together on the bench for system-level testing. Because the dual DIO-96 plug-in cards allow control of only eight slices at one time, not all of the slices in each box could be controlled simultaneously from the TCS. This did not pose a problem since control of all slices was not necessary during the tests. Tests at this level ranged from verifying relay operation in the BUPAC to determining the probability of pulse detection of the integrated Receiver Trigger boxes.

Example: Receiver/Trigger Probability of Detection

An example of an exhaustive system level test was determining the probability of detection for the integrated Receiver and Trigger boxes. Because the quantity of adjustable parameters in the Triggers were large, predicting optimum settings for pulse detection was a complex problem. Predicting optimum settings for different background noise environments complicated it further. To find the maximum probability of pulse detection for a given background environment, an automated test was developed which varied the Trigger threshold settings, the quantity of coincident sub-band triggers, and the signal-to-noise ratio. The test was replicated for three TATR tuning bands (38, 58, 78 MHz center frequencies) and three expected background noise environments.

The automated test used the AWG2041 arbitrary waveform generator to continuously replay the background noise environment. The DG535 pulse generator issued a 10 ns pulse, which was combined with the noise. Two SR620 counters were used to count the total number trigger output pulses during the test and the number of trigger output pulses which occurred within a 2 µs gate after a pulse was input into the system. The test operated in a series of nested loops, generating all possible combinations of threshold settings, quantity of coincident sub-band triggers, and signal-to-noise ratios (SNRs). The first loop incremented the quantity of coincident sub-band triggers from 1-8, 8 being the most stringent triggering criteria. It also opened a file for data logging. The second loop varied the SNR by adjusting the TATR attenuator levels from 0-41 dB. In the third loop, all of the sub-band threshold values were incremented from 40-120. For each iteration of the third loop, 36 input pulses were commanded by the TCS. The resulting quantity of gated triggers and total triggers counted were saved with the current threshold levels, attenuator settings, and number of required coincident sub-band triggers.

These series of tests yielded the optimum settings for maximizing the probability of pulse detection in some expected FORTÉ noise environments. Even with automation, each of these tests required five hours to complete. Without automation, the tests would have been impossible to perform within a reasonable amount of time, and determination of optimum triggering parameters would have been left to analysis of on-orbit data.

Integrated Testing & Remote Instrument Control

Control of the RF payload moved to the UNIX-based flight ground support equipment (GSE) after the spacecraft was fully assembled, ending testing with the PC-based test and control system. To partially extend the automated test procedures beyond bench-level testing, a method of controlling both the RF hardware and input signal sources from the spacecraft GSE was pursued. This was accomplished by developing an instrument control interface that could issue commands from the GSE to a PC running the Linux operating system and outfitted with a GPIB card (figure 7).

Payload Testing with Linux & GPIB

Linux is an operating system developed to run UNIX on inexpensive PC platforms as alternatives to high-end workstations. It was initially a small project started by Linus Torvalds[5] at the University of Helsinki, but has subsequently become a very robust and popular operating system through the cooperative efforts of software developers connected by the Internet. In addition to being freely distributed under the GNU General Public License, the Linux Lab Project[6] supports the development of hardware drivers for most of the popular data acquisition and GPIB plug-in cards for PCs.

To control the bench instruments from the Linux computer, simple scripts were written in the Tcl programming language. The scripts sent timed commands to the input instruments via a National Instruments AT-GPIB board. Because the GSE and Linux PC communicated seamlessly over a network, these GPIB script commands were issued remotely from the GSE during testing of the integrated RF payload. The GPIB commands were timed to coincide with GSE configuration commands to the RF payload. Data acquisition was carried out in the flight DAS and downloaded to the GSE over the spacecraft communications link. This approach avoided breaking the integrated flight configuration and kept physical operations near the spacecraft to a minimum.

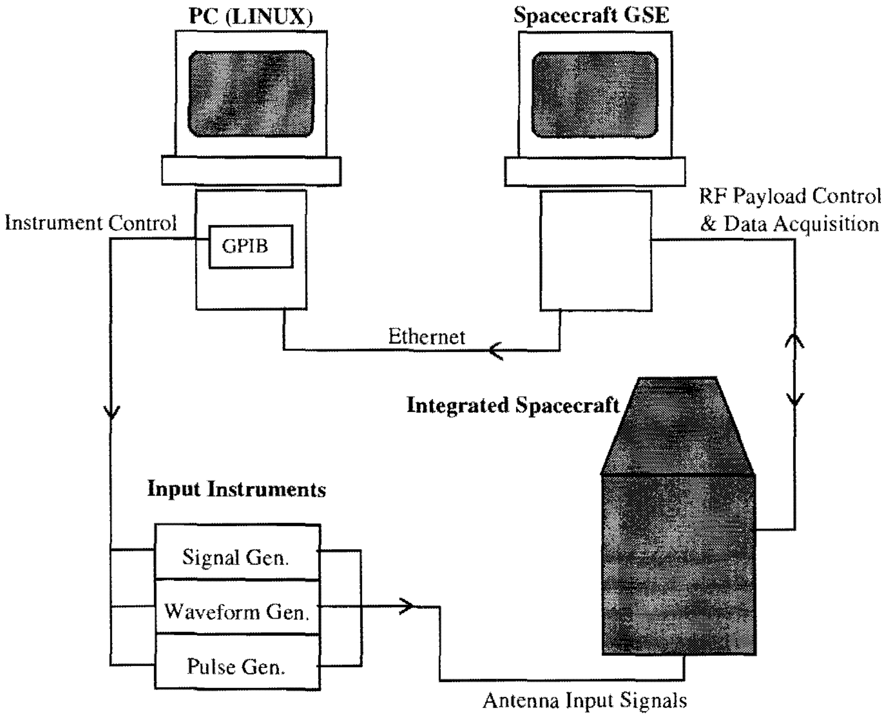


Figure 7. Remote instrument control after payload integration.

Summary

An inexpensive, PC-based test and control system has been examined which reduces the development time of one-of-a-kind space hardware by integrating control of the hardware and bench test instruments. The intuitive LabVIEW® programming environment accelerates the development of test and control software and is easily adapted to changing hardware designs. Automated test procedures provide continuity between the EM and flight stages of hardware development, and a modular approach in both hardware and software design simplifies testing and integration. The system allows more complete data to be taken than could be accomplished without automation, illuminating faults that otherwise might not be discovered. Finally, automated testing of hardware can be extended beyond integration on the flight spacecraft by using a Linux machine for remote instrument control.

Acknowledgments

The authors wish to thank Donald Enemark, David Guenther, Martin Shipley, Robert Franz, and Michael Carter for their contributions to this work. The FORTÉ Project is supported by the U.S. Department of Energy.

References

1. Cautero, G., G. Paolucci, B. Brena, R. G. Agostino, R. Tommasini, G. Comelli, and R. Rosei, "A LabVIEW®-based control system for a surface science experimental station," Measurement Science and Technology, Vol. 5,

2. Nafalski, Andrzej, T. Doan, Ö. Göl, "Application of virtual instrument concept in magnetic measurements," Journal of Magnetism and Magnetic Materials, Vol. 160, 1996, pp. 154-160.

3. Tondre, Holley, A. Bernal, J. Valle "VISTA: Virtual Instrument Standard Test Architecture," IEEE AES Systems Magazine, March 1995.

4. Brunschwyler, John, K. Kelly, K. Kubota, "A Non-Standard Production Approach to Satellite Constellations," Proceedings of the 10th Annual AIAA/Utah State University Conference on Small Satellites, Logan, Utah, September 1996.

5. "The Linux Bible", Yggdrasil Computing Inc., San Jose, CA, 1995.

6. The Linux Lab Project, URL:
http://obelix.chemie.fu-berlin.de/index.html

7. "LabVIEW® 4.0 User Manual", National Instruments, Inc., Austin, TX, 1996.

Biographical Sketch

Dallas Masters is a graduate research assistant in the Space Engineering Group (NIS-4) at Los Alamos National Laboratory. He received a Bachelor of Science in Engineering Science from Trinity University in 1995. His interests include spacecraft systems design, remote sensing, and test automation.

Kimberly K. Ruud is a technical staff member at Los Alamos National Laboratory. She received a Bachelor of Science in Electrical Engineering from South Dakota State University in 1994. In 1996 she received a Masters of Science in Electrical Engineering - Signal Processing and Communications from the University of New Mexico. Her interests include spacecraft attitude determination and control, digital signal processing and digital control system design.