Autonomous Telemetry Collection for Single-Processor Small Satellites

Dave Speer
Northrop Grumman Electronic Systems
Space Technology & Services
4276 Forbes Blvd.
Lanham, MD 20706
301-286-8499
dave.speer@ngc.com

Abstract

.For the Space Technology 5 mission, which is being developed under NASA's New Millennium Program, a single spacecraft processor will be required to do on-board real-time computations and operations associated with attitude control, up-link and down-link communications, science data processing, solid-state recorder management, power switching and battery charge management, experiment data collection, health and status data collection, etc. Much of the health and status information is in analog form, and each of the analog signals must be routed to the input of an analog-to-digital converter, converted to digital form, and then stored in memory. If the micro-operations of the analog data collection process are implemented in software, the processor may use up a lot of time either waiting for the analog signal to settle, waiting for the analog-to-digital conversion to complete, or servicing a large number of high frequency interrupts. In order to off-load a very busy processor, the collection and digitization of all analog spacecraft health and status data will be done autonomously by a field-programmable gate array that can configure the analog signal chain, control the analog-to-digital converter, and store the converted data in memory.

Introduction

For many spacecraft, there are requirements for the down-linked health and status telemetry to include information on various voltages, currents, temperatures, pressures and other signals collected from many locations within the vehicle. To collect this telemetry there is typically a network of analog signal switches which "funnel" down to the input of an analog-to-digital (A-to-D) converter. To sample and digitize the signal for any one of the various analog channels, the following series of micro-operations will likely be needed. The subsystem must first configure the network of switches to select the analog signal of interest, then wait for the analog signal to slew and settle, then start the A-to-D conversion, then wait for the A-to-D conversion to complete, and then read the data from the output of the A-to-D converter (ADC). To sample N signals, this series of micro-operations must be repeated N times.

In larger spacecraft architectures for which a distributed data system (like MIL-STD-1553) is used, the task of collecting analog health and status telemetry is often assigned to a set of smaller processors that are locally embedded in the various subsystems or remote terminals. In many cases these locally embedded processors are relatively lightly loaded, and they can use wait loops or interrupts to handle the time delay while the analog signal settles or while the analog signal is being converted to digital data. The wait loop and interrupt techniques may work when the local processor does not have much else to do. But if the local processor has many tasks other than analog telemetry collection, and if periodic samples of a large number of analog quantities are needed while other tasks are executing, then a software implementation of the delays for the A-to-D conversion micro-operations will probably be complex, difficult, inefficient or even impossible.

This paper describes a mostly hardware-based approach to the collection of analog health and status telemetry for a spacecraft that has only one microprocessor. This approach requires a minimum of software overhead and is being implemented by the command and data handling (C&DH) subsystem as part of the Space Technology 5 (ST-5) mission. In order to put the ST-5 analog telemetry collection approach into perspective, the paper starts with an overview of the ST-5 spacecraft and an overview of the C&DH subsystem's interfaces and memory architecture. Following that, there is a description of some fundamental hardware-software timing concepts and a review of the arbitration scheme used for direct memory access requests. The details of the autonomous analog telemetry collection process include discussions of analog channel number assignments, the use of tables in a static random access memory, and the ability to start the collection of analog telemetry either immediately after a request or at a predetermined exact time. Some of the system's built-in test features are reviewed at the end. After the paper's conclusion, there is a list of acronyms for easy cross-reference.

ST-5 Spacecraft Overview

As part of NASA's New Millennium Program, the ST-5 mission will flight validate a variety of new technologies focused on reducing size and power consumption for micro-satellite and constellation mission applications. Included are a miniature X-band transponder (XPDR), a cold gas micro-thruster, two different variable emittance controllers (VEC), and a radiation-tolerant +0.5 volt CMOS logic encoder. In addition to the New Millennium technologies, other spacecraft subsystems include the power system electronics (PSE), a lithium-ion battery, solar arrays, a digital sun sensor (DSS), and a propulsion system. The main science instrument is a high-resolution magnetometer (MAG).

Each of the three identical 25-kilogram spin-stabilized spacecraft has a total power budget of approximately 22 watts. The outside shape is a right octagonal prism, or 8-sided flat "cylinder," approximately 0.6 meters in diameter and 0.3 meters tall. The spacecraft spin axis is the axis of symmetry or axis of rotation of the cylinder, and the spacecraft spin rate will be about 20 revolutions per minute. The top and bottom "decks" are octagons, and eight small solar panels cover the outsides of the eight sidewalls of the cylinder. A brick-shaped "card cage" enclosure mounts between the two decks. The two slots in this card cage house the C&DH electronics board and the power system electronics board.

C&DH Subsystem Overview

The flight computer (C&DH board) is designed around a single radiation-hardened processor. Memory types include electrically-erasable programmable read-only memory (EEPROM) for non-volatile program storage, dynamic random access memory (DRAM) for real-time program execution and solid-state data recording, and static random access memory (SRAM) for temporary storage of digitized analog health and status data (and magnetometer science data). The C&DH design incorporates three Actel RT54SX32S field-programmable gate arrays (FPGA). The design will allow the processor (flight software) to have continuous read-write access to all memory and all FPGA registers, with a direct memory access (DMA) controller for shared access to the SRAM space. The analog telemetry collection "machine" and the DMA controller will be implemented by a portion of one of the FPGAs.

The C&DH will implement direct digital and analog interfaces to all of the other spacecraft subsystems and components. There will not be a distributed data system that uses a standardized serial bus such as MIL-STD-1553 or MIL-STD-1773. The single processor will be connected to all other spacecraft subsystems and will do all of the data acquisition and control tasks associated with attitude control and maneuvering, command uplink and telemetry downlink, science data processing, battery state-of-charge management, power switching, technology validation data collection, health and status data collection, fault detection and correction, etc. By doing the majority of the analog health and status telemetry collection functions in hardware, the software required to handle multiple, asynchronous data streams will be simpler, and the average processor interrupt frequency will be greatly reduced. Figure 1 is a block diagram showing the main ST-5 subsystems and components, along with their C&DH interfaces. For more details on the ST-5 mission and/or C&DH design, see reference[1].

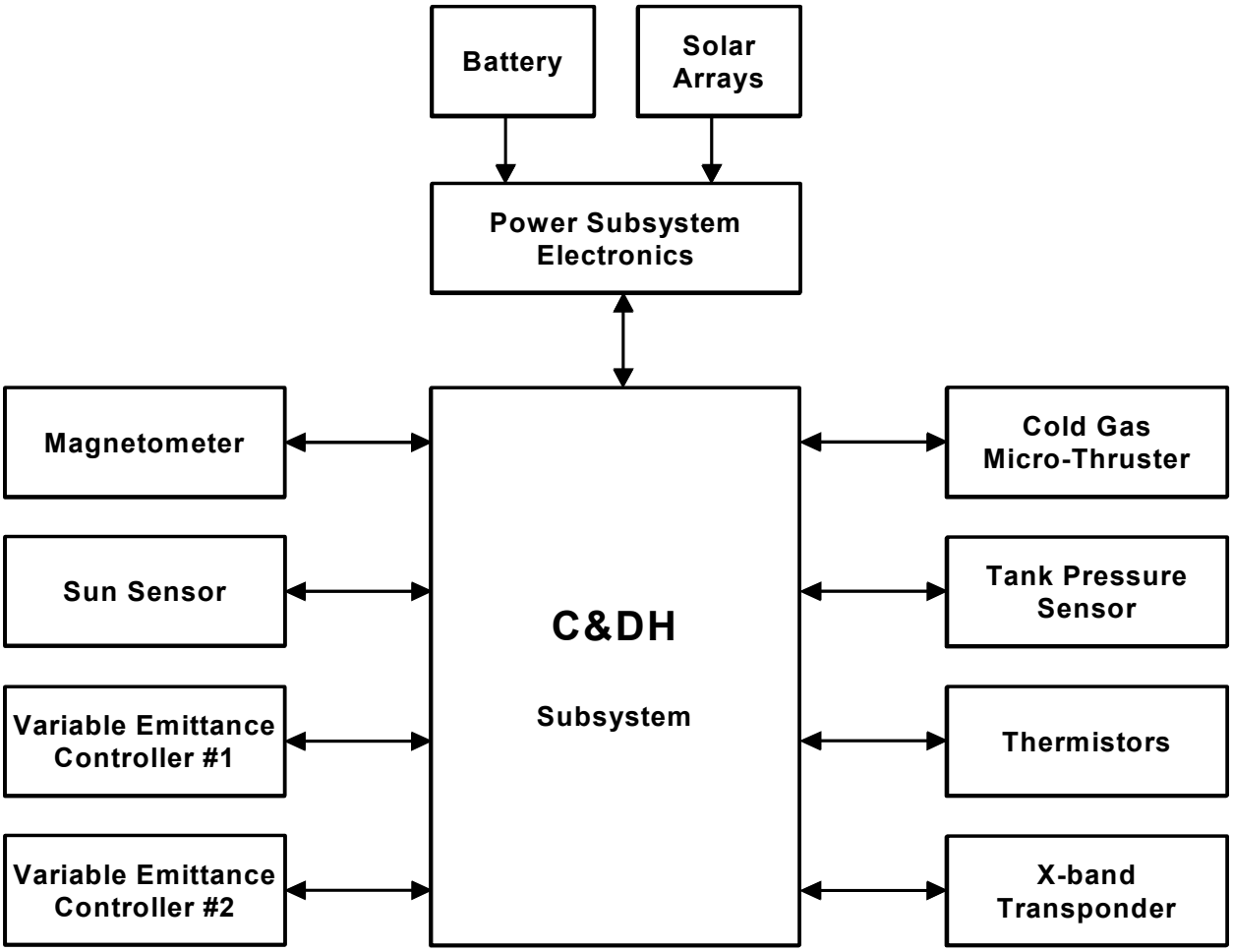

Figure 1: Block Diagram of ST-5 Subsystems and Components, with C&DH Interfaces

Fundamental Clocks and Timing

Many of the data acquisition and control tasks performed by the ST-5 flight software are executed repetitively on one second centers. There is a 1 Hz pulsed signal that interrupts the flight software each second, and this pulse is used to keep the software execution synchronized with the operation of the hardware. Two additional clock signals are derived from the 1 Hz pulse and kept exactly in phase with it. The first is a 1 kHz clock that divides each second into 1000 milliseconds (numbered 0 to 999). The second is a 12 kHz clock that divides each millisecond into twelve 83.3 microsecond "DMA time slots" (numbered 0 to 11). The 1 kHz clock is used to control the sampling and digitization of a sequence of analog signals on regular 1-millisecond boundaries. The 12 kHz clock is used to schedule the various micro-operations associated with the A-to-D conversion process, and to time-multiplex the DMA controller with the collection of science data from the magnetometer.

Figure 2 shows the 1 Hz clock pulses that interrupt the processor at the start of each 1-second interval, along with the division of each second into milliseconds. Figure 2 also illustrates the numbering of each millisecond, and the further division of each millisecond into 12 equal DMA time slots, each 83.3 microseconds long.

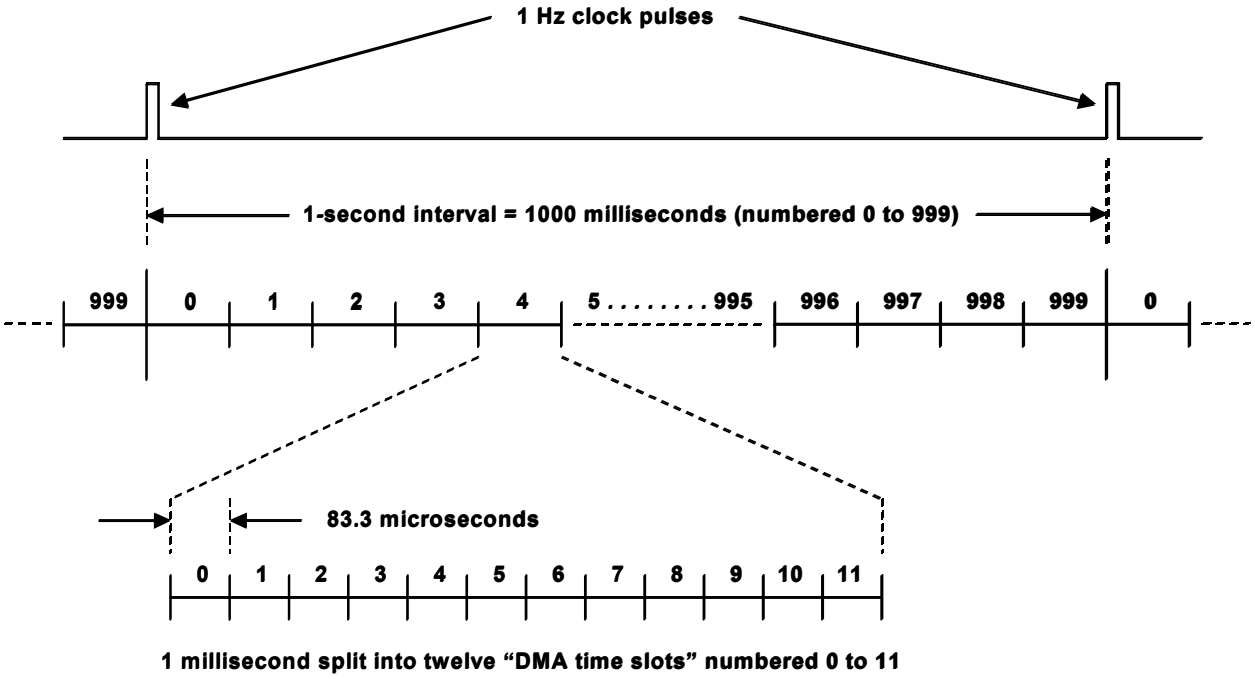

Figure 2: Fundamental Clocks and Timing Intervals

Direct Memory Access (DMA) Controller

Most of the time, the processor (flight software) will have continuous read-write access to the SRAM memory. Occasionally, however, the analog telemetry collection "machine" will also need temporary access to the same SRAM, and both cannot access the same SRAM part at the same instant in time. In order to handle the situation where an external device needs direct access to the system's memory, there is a request-grant arbitration scheme that allows the analog telemetry collection machine to request control of the system's address and data busses by sending a "bus request" signal to the processor. When the processor sees this request signal, it will finish the instruction that it is currently executing, send a "bus grant" signal to the analog telemetry collection machine, and allow that machine to have temporary but complete control over the system's address and data busses. After the analog telemetry collection machine receives the bus grant signal, and after it has done the DMA read from SRAM or the DMA write to SRAM, it will signal that it is done by turning off the bus request signal to the processor. When the processor sees that the analog telemetry collection machine is done, it will turn off the bus grant signal, re-assert control over the system's address and data busses, and resume its previous instruction flow. The analog telemetry collection machine and the DMA controller are implemented in a portion of one of the three FPGAs.

The hardware design and the number of times that the analog telemetry collection machine will need to access the SRAM is such that the amount of time that the DMA controller is accessing the SRAM will be a very small fraction of the total time (less than 0.1%). The DMA controller will not directly access the EEPROM or DRAM spaces, but each DMA cycle to or from SRAM might "pause" the executing flight software program for a very short time. Referring back to Figure 2, each millisecond is divided into twelve DMA slots numbered 0 to 11. The autonomous analog telemetry collection process described in this paper uses DMA slots 0 and 11. (The magnetometer uses slots 1, 4, 7 and 10, and slots 2, 3, 5, 6, 8 and 9 were reserved for two other science instruments.) Figure 3 shows the address (ADDR), control (CNTL) and data lines for processor accesses to FPGA registers and SRAM, with a similar set of lines needed for DMA controller accesses to SRAM.

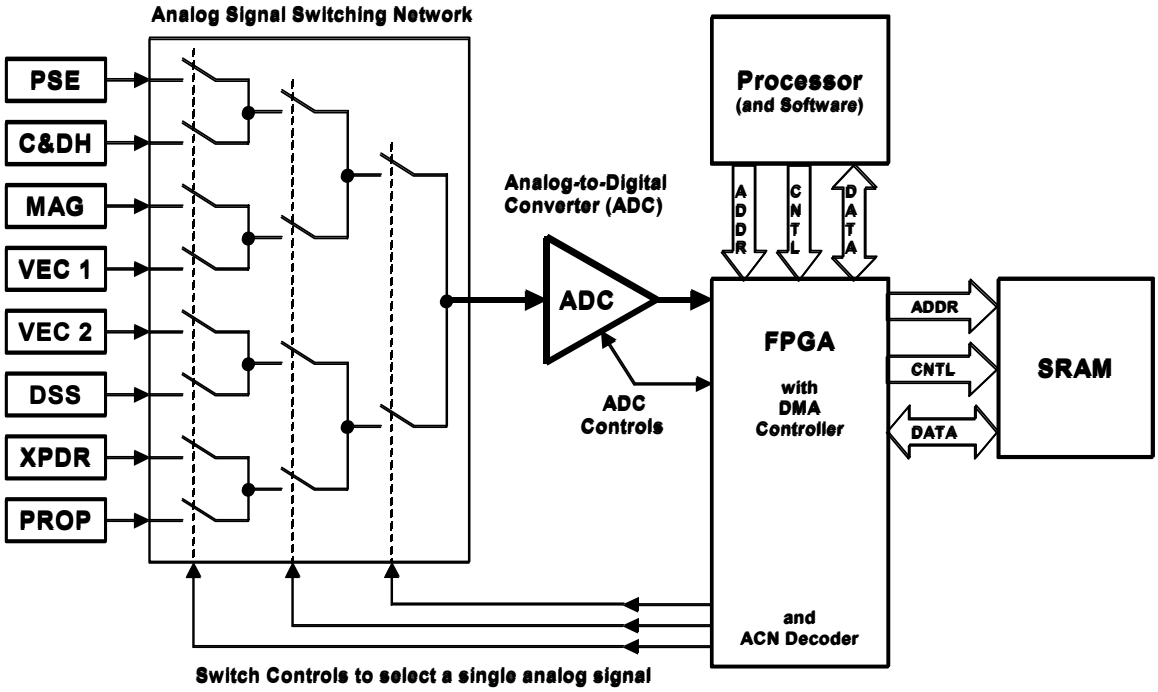

Figure 3: Analog Signal Selection by the ACN Decoder

Autonomous Analog Telemetry Collection

Analog Channel Numbers

There are a total of 71 different analog signals that can be sampled by the ST-5 system. Of this total, 33 are for thermistors to sense temperatures, and 38 are used to measure voltages, currents, pressure and other parameters. The sources of these signals are distributed throughout the spacecraft, and the signals can come from the PSE board (including the battery and solar arrays), the C&DH board, the magnetometer, the two variable emittance controllers, the digital sun sensor, the transponder, the propulsion system (PROP) components, or the spacecraft structure itself. There is a network of analog switches that "funnels down" to the input of a single A-to-D converter on the C&DH board, and this A-to-D converter will be used to digitize all of the analog signals on the spacecraft.

Each of the different ST-5 analog signals, regardless of whether it is a voltage, a current, a temperature or a pressure, has a unique 7-bit analog channel number (ACN) assigned to it, and each assigned ACN value corresponds to a single and unique analog signal source. The ACN values for the thermistor channels range from 25 to 31 and from 48 to 73. The ACN values for the other analog signal channels range from 0 to 12, 16 to 24, and 32 to 47. Because there is a one-to-one correspondence between each ACN and each analog signal source, there is also a one-to-one correspondence between each ACN and each unique configuration of the analog switch network that "connects" the selected analog signal source to the input of the A-to-D converter. This one-to-one correspondence is made by a decoder that translates each ACN value into a unique setting of the control lines that configure the network of analog switches.

Figure 3 shows the various analog signal sources on the left, feeding into the network of analog switches that routes one analog signal to the input of the A-to-D converter (ADC) at any given time. The ACN decoder is inside the FPGA, and it generates the switch controls that configure the analog switch network.

The ACN assignments to the various signal sources were made to minimize the network of analog switches and to simplify the logic that decodes each ACN and drives the select lines that control the analog switch network. The ACN values of 13, 14, 15, and 74 through 127 do not correspond to any of the analog signal sources described above. When these unassigned ACNs are decoded, they will "connect" the ADC input to analog ground, and will return a digitized data value of zero if put into an analog Collection Control Table.

Control and Data Tables in SRAM

The design of the autonomous analog telemetry collection system is based on the use of matched pairs of tables stored in the shared SRAM, where the first table in each pair is for collection control, and the second table in the pair is for storage of the digitized data. Since the number of memory locations in the SRAM is very large compared to the number of different analog signals in the system, the length of the tables and the number of pairs of tables is almost unlimited. However, in order to simplify the addressing scheme used to select and access each table, the size of the tables and the number of table pairs will both be a power of two. For ST-5 there are four matched pairs of tables in the SRAM, and each of the tables has 128 entries in it. The four pairs of tables are numbered 0, 1, 2 and 3. The processor (software) normally writes to the Collection Control tables and reads from the Digitized Data tables. The analog telemetry collection "machine" inside the FPGA can only read from Collection Control tables and can only write to Digitized Data tables. See Table 1 below.

Before an autonomous analog telemetry collection can be started, the processor (software) must first pre-load a collection control table with the number of analog channels to be sampled and digitized, followed by a sequence of ACNs that defines which analog signals will be sampled and determines the order in which those analog channels will be sampled. This loading is done by the processor writing to the SRAM, and if one or more of the collection control tables will never be changed, it could be loaded only once during power-on initialization. Alternatively, any of these collection control tables could be modified by flight software either before or after they are used. The flight software should never modify the contents of an analog collection control table while it is being used.

Table 1: Matched Pairs of SRAM Tables for Collection Control and Digitized Data

SRAM Table Description	# of SRAM Locations	Hardware (FPGA) Access	Flight Software Access
Analog Telemetry Collection Control Table #0	128	Read	Write* Read
Analog Telemetry Digitized Data Table #0.	128	Write	Read Write**
Analog Telemetry Collection Control Table #1	128	Read	Write* Read
Analog Telemetry Digitized Data Table #1	128	Write	Read Write**
Analog Telemetry Collection Control Table #2	128	Read	Write* Read
Analog Telemetry Digitized Data Table #2	128	Write	Read Write**
Analog Telemetry Collection Control Table #3	128	Read	Write* Read
Analog Telemetry Digitized Data Table #3	128	Write	Read Write**

* Write access for table setup before collection starts (could be one-time only).
** Write access for optional zeroing after data readout.

Each of the analog collection control tables can have from 1 to 127 analog channel numbers (ACNs) in it, and those 7-bit ACNs can be in any order, with values repeated in any pattern, and with some values not used at all. The first location in the collection control table is reserved for the number (ranging from 1 to 127) of channels to sample and digitize in any given collection process. The remaining locations in the collection control table are for the sequence of ACNs. For the current ST-5 design, the time between samples of the successive ACNs in a single collection control table is fixed at 1 millisecond. This limits the maximum sampling rate for back-to-back samples of the same channel to 1 kHz. Integer sub-multiples of this 1 kHz maximum rate are possible by putting the same ACN in every other entry or every third entry of the collection control table.

For each of the four Collection Control Tables there is a corresponding Digitized Data Table that is a "mirror image" of the control table. Where the first location in the collection control table has the number of analog signals to digitize, the first entry in the corresponding digitized data table will have the actual number of analog signals digitized when the collection process completed. Where the collection control table has the sequence of ACNs that defines the sequence of analog signals to digitize, the corresponding digitized data table will have the digitized data from each of the those ACNs, in the exact same sequence. By having multiple collection control tables (which can be pre-loaded and then started in any order and at any time) the design is very flexible, and requires a minimum of flight software overhead. Figure 4 shows a Collection Control Table and Digitized Data Table for a matched pair.

Analog Telemetry Collection Control Table (written by software)

SRAM Location	7-bit Contents of that SRAM location
0	Table Length = Number of analog signals to sample (1 to 127)
1	First 7-bit ACN defines first analog signal to be sampled
2	Second ACN defines second analog signal to be sampled
3	Third ACN
4 – 125	Fourth through 125th ACNs
126	126th ACN
127	127th ACN

Analog Telemetry Digitized Data Table (read out by software)

128	Number of analog channels digitized by hardware (1 to 127)
129	Digitized data for first analog signal from First 7-bit ACN
130	Digitized Data for second analog signal from Second ACN
131	Digitized Data for Third ACN
132 – 253	Digitized Data for fourth through 125th ACNs
254	Digitized Data for 126th ACN
255	Digitized Data for 127th ACN

Figure 4: Analog Collection Control Table and "matching" Digitized Data Table

Analog Collection Start-Up

After pre-loading the collection control table(s) with a table length and a sequence of 7-bit analog channel numbers, the collection process is ready to be started. If any one or all four of the collection control tables have been loaded (and they can all be completely different) then any one of the four can be started.

When the flight software is ready to start any one of the collection control tables, it only has to write a 2-bit number (either 0, 1, 2 or 3) to the Analog Collection Start Address. This 2-bit number will select which of the four analog collection control tables will then be used to sample and digitize the next group of analog telemetry signals (refer back to Table 1). After flight software has written to the Collection Start Address, it can then go off and do other things while the analog data collection starts and proceeds automatically under FPGA control.

If the system is in immediate mode (described below), then the analog data collection will start at the beginning of the next millisecond interval. If the system is in time-delayed mode (also described below), then the analog data collection start-up will be delayed by the number of milliseconds in a 10-bit analog delay register. Once the analog data collection starts, the FPGA will begin reading through the selected collection control table and storing the digital data for each ACN at the matching location in the digitized data table.

Analog Telemetry Collection Modes

There are three different collection modes for the analog health and status telemetry. Two of those modes are called "automatic" and most of this paper is dedicated to describing how these automatic/autonomous modes work and why they are advantageous. The third mode is called "manual". The two automatic modes (described below) can be used in flight, and it is in the automatic modes that the flight software will have very simplified and streamlined top-level control over the analog telemetry collection operations, with no need to handle all the low-level micro-operations for analog signal selection and A-to-D conversion. The manual mode is available for use during integration and test on the ground, and the manual mode is described later in the paper, in the section titled Built-In Test Features. Any one of the three modes is selected by software writing to a 2-bit mode select register.

The two automatic modes (in which a table of digitized analog health and status data can be autonomously collected) are called "immediate" and "time-delayed." In immediate automatic mode, the collection process will start one millisecond after the flight software writes to the collection start address. The immediate mode is good for sampling signals that vary very slowly with time (like temperatures). In this mode it is also very easy for the software to start a collection during the current 1 Hz cycle, while reading the digitized data and starting another collection during the next 1 Hz cycle. In time-delayed automatic mode, the flight software can write to a 10-bit register that specifies the delay in milliseconds from the start of each 1-second interval. After the flight software writes to the collection start address in time-delayed mode, the start of the collection process will be delayed (from the 1 Hz pulse) by the number of milliseconds in the 10-bit delay register. The advantage of the time-delayed mode is that the analog signal samples can be scheduled to occur at a fixed repeatable time or during a signal transient of interest. An example might be the sampling of PSE load current when the X-band transmitter is turned on.

Figure 5 illustrates how the start of a 2-channel analog collection will depend on whether the system is in immediate automatic mode or time-delayed automatic mode. (A 2-channel collection is one for which the first entry in the collection control table is 2, and then there are only two ACNs following the first entry.) At the top of Figure 5 are the 1 Hz clock pulses and the millisecond intervals numbered 0 to 999. In immediate mode the 2-channel analog collection would start during the millisecond interval immediately following the processor (software) write to the analog collection start address. In this case the two analog signals would be sampled and digitized during milliseconds numbered 3 and 4. In time-delayed mode (and for a 10-bit delay equal to 995) the start of the 2-channel analog collection would be delayed by 995 milliseconds from the 1 Hz clock pulse. In this case the two analog signals would be sampled and digitized during milliseconds numbered 996 and 997, as shown at the bottom of Figure 5.

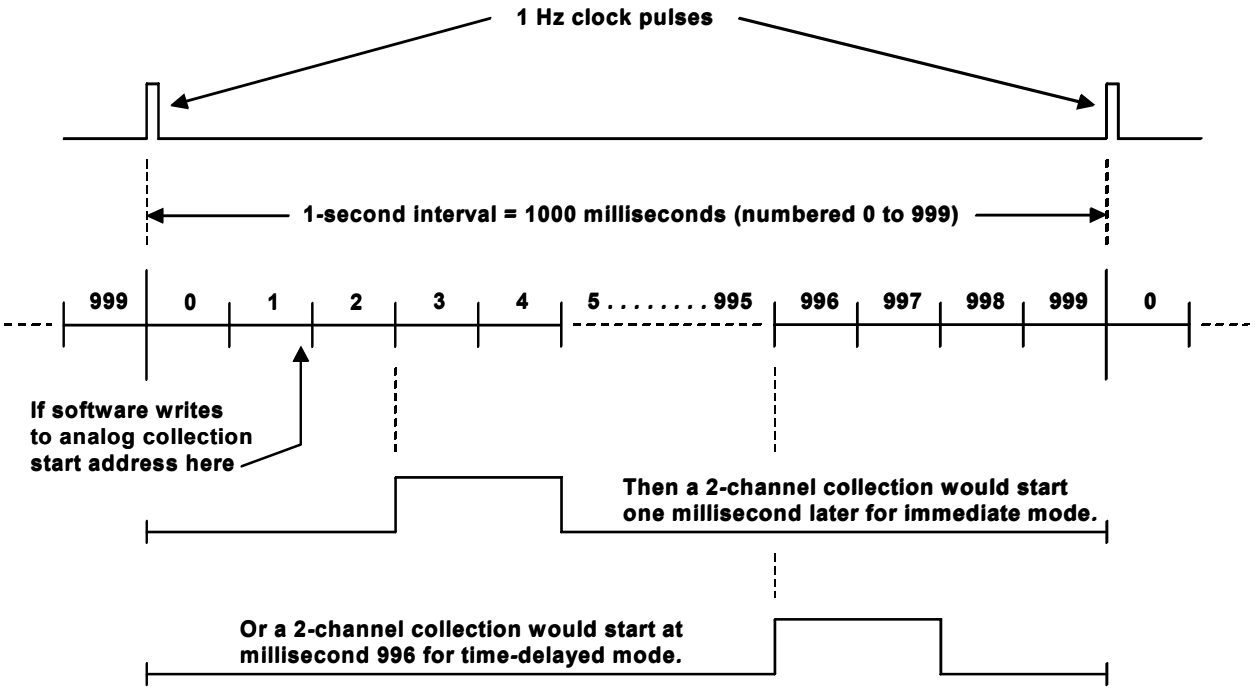

Figure 5: Analog Telemetry Collection Start-Up in Immediate Mode and Time-Delayed Mode

Analog Sampling Micro-Operations

Once a new analog telemetry collection process is started (either immediately or after a time delay) the process will first look at the number of analog signals to digitize (first entry in the selected collection control table) and then start selecting and digitizing analog telemetry signals according to the sequence of ACNs in the next N entries in the collection control table. The analog collection "machine" will generate the analog switch network control signals by decoding the ACN, then handle the start, done and readout signals for the A-to-D conversion, and then store the data from the ADC into the appropriate location in the digitized data table. The details of the required micro-operations are given below.

During the first millisecond of an analog data collection, the FPGA will request a DMA cycle to read the first entry in the collection control table and store it in a register that holds the number of analog signals to sample and digitize for this collection process. For the next N milliseconds, if another A-to-D conversion is required, DMA slot 0 is used for the DMA read cycle to fetch the next ACN from the SRAM collection control table. Once the ACN is latched in an FPGA register and decoded to set the configuration of the analog switch network, the analog signal is allowed to slew and settle during DMA slots 1 through 10. The A-to-D conversion is initiated at the beginning of DMA slot 11, and after the conversion completes there is a DMA write cycle to put the ADC output into the digitized data table.

After each DMA write cycle to the digitized data table the FPGA increments a counter which is compared to the number of analog signals to sample. If the counter value is less than the number if analog channels to sample and another A-to-D conversion will be required during the next millisecond, then the next DMA slot 0 will be used to read the next ACN from SRAM and the process repeats. If the counter value is equal to the number of analog channels to sample and this was the last ACN in the collection control table, then the next DMA slot 0 will be used for a DMA write cycle to put the number of ACNs converted into the first element in the digitized data table.

To monitor the progress of an automatic analog telemetry collection process, there is an FPGA register where software can read out the number of ACNs that have been digitized and stored in SRAM so far. This number will count from 0 to the number of channels defined by the first entry in the Collection Control Table. When hardware completes the analog telemetry collection process, this number will be transferred to the first entry in the Digitized Data Table. When the automatic collection process completes, the number of channels requested (written by flight software) and the number of channels digitized (written by hardware) should always be equal.

Completion and Acknowledge

When all of the analog telemetry signals defined by the collection control table have been sampled, digitized and stored in the matching digitized data table, the FPGA will set a status bit in a readable register to signal to the processor that the analog data collection has completed and the digitized data table is ready for readout. This status bit can be periodically polled by software to see if the data is ready, or the bit can be enabled to generate an interrupt that would provide immediate notification that the data is ready.

When the software either sees the 0 to 1 transition of the polled status bit or responds to the analog data ready interrupt, it will read from the digitized data table and collect all of the analog telemetry measurements at once. To acknowledge an analog data ready interrupt or to clear the polled status bit, the software will write to an Analog Data Ready Interrupt Acknowledge Address. Once the digitized data table has been read out and software has acknowledged the read-out, the software can re-start the same collection control table or start a different collection control table as required.

Built-In Test Features

To complement the automatic operating modes (immediate and time-delayed) which can be used in flight, there is a manual operating mode that can be used on the ground for integration, test and debug purposes. In manual mode, keyboard control is provided for all of the micro-operations associated with analog telemetry signal sampling. In manual mode, a test engineer can write directly to the ACN holding register to select any analog channel for a static measurement, write to another register to start the A-to-D conversion process, and then read from a register to get the digitized data directly from the output of the ADC. In manual mode, all of these operations are done through a set of diagnostics software that allows keyboard writes and reads to registers inside the FPGA.

In addition to the built-in signal probing capability that is provided directly by the FPGA device, the C&DH board has an externally accessible test connector where a variety of signals can be monitored with an oscilloscope, logic analyzer or other ground support equipment. At this test connector one can observe a variety of analog telemetry collection activities, including the least significant bits of the counter for number of analog signals digitized so far, the least significant bits of the ACNs as they are being sampled, the control signals for the ADC, and the operation of the DMA controller. While any of these signals are being observed, one can simultaneously look at the analog signal at the input to the A-to-D converter.

Summary and Conclusion

For the ST-5 spacecraft, the collection and digitization of analog health and status telemetry will be done autonomously by an FPGA-based sequencer and DMA controller. An automatic collection approach was developed to offload the processor (software) from performing all the low-level micro-operations such as writing each set of channel select bits to the analog switch network, waiting for the analog signal chain to settle, starting the A-to-D conversion, waiting for the A-to-D conversion to complete, and then storing the digitized data in memory.

Flight software will control the collection by loading a collection control table that defines the number of analog channels to digitize, which channels to digitize, and what order to digitize them in. After loading a collection control table the flight software will write a 2-bit number to a collection start address. This 2-bit number will select one of the four collection control tables that will define the sampling and digitization order for the next group of analog signals. After software has written to the collection start address, the analog data collection can actually start either immediately or after a programmable time delay. An FPGA will then handle all the micro-operations required for analog signal selection, analog-to-digital conversion, and storage of the digitized data into an SRAM data table. When the entire analog data collection process completes, the FPGA will inform the software that all the digitized data is ready for readout from SRAM. After software reads this data from SRAM, it can start another analog telemetry collection, using any one of the four pre-loaded collection control tables.

By having a number of collection control tables that can be pre-loaded and then run in any order and at any time, the design is very flexible in terms of the number of analog signals sampled, the order of the signal samples, and the frequency of the signal samples. Different collection control tables can be configured to match up with the various spacecraft operating modes or telemetry downlink rates, and the SRAM tables are easy to re-configure. The design requires a minimum of real-time software to support it, and the processor can either be interrupted when the autonomous collection completes, or poll a status bit that indicates "collection complete." The only constraint is that there cannot be more than one collection control table running at any given time because there is only one analog-to-digital converter.

Acknowledgement

The design work described in this paper was done at the Goddard Space Flight Center, for the ST-5 project, in the Flight Electronics Branch (Code 561), under NASA contract NAS5-99124.

References

1. Speer, D. and G. Jackson and D. Raphael, "Flight Computer Design for the Space Technology 5 (ST-5) Mission," Proceedings of the 2002 IEEE Aerospace Conference, Big Sky, MT, March 2002.

Acronym List

A-to-D	=	Analog-to-Digital
ACN	=	Analog Channel Number
ADC	=	Analog-to-Digital Converter
C&DH	=	Command & Data Handling subsystem
CMOS	=	Complimentary Metal Oxide Semiconductor
DMA	=	Direct Memory Access
DRAM	=	Dynamic Random Access Memory
DSS	=	Digital Sun Sensor
EEPROM	=	Electrically-Erasable Programmable Read-Only Memory
FPGA	=	Field Programmable Gate Array
Hz	=	Hertz
kHz	=	kilo-Hertz
MAG	=	Magnetometer
NASA	=	National Aeronautics and Space Administration
PROP	=	Propulsion (subsystem)
PSE	=	Power System Electronics
SRAM	=	Static Random Access Memory
ST-5	=	Space Technology 5
VEC	=	Variable Emittance Controller
XPDR	=	X-band Transponder