Testbed for Development of a DSP-Based Signal Processing Subsystem for an Earth-Orbiting Radar Scatterometer[1]

Douglas J. Clark, James P. Lux, and Mike Shirbacheh
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Pasadena, CA 91109
818-354-3427, 818-354-2075, 818-354-6147
douglas.j.clark@jpl.nasa.gov, james.p.lux@jpl.nasa.gov, m.shirbacheh@jpl.nasa.gov

Abstract–A testbed for evaluation of general-purpose digital signal processors in earth-orbiting radar scatterometers is discussed. Because general purpose DSP represents a departure from previous radar signal processing techniques used on scatterometers, there was a need to demonstrate key elements of the system to verify feasibility for potential future scatterometer instruments. Construction of the testbed also facilitated identification of an appropriate software development environment and the skills mix necessary to perform the work.

A testbed was constructed with three Astrium MCMDSPs, based on the Temic TSC 21020 general purpose DSP. Commercial data conversion hardware and high-speed serial communication hardware was interfaced to the MCMDSPs to allow demonstration of the key interfaces between subsystem elements: DSP program loading, synchronization and communication between multiple DSPs, interface to the scatterometer radio frequency subsystem, commanding, and science data delivery to the instrument data handling subsystem.

A baseline set of requirements for the radar signal processing subsystem was established. From these requirements, signal processing algorithms such as digital filters and FFTs were developed using a combination of standard library functions and custom software. A software framework was developed to coordinate execution of the periodic signal acquisition and processing routines with asynchronous commanding and timekeeping functions. Emphasis was placed on developing modular software that would be applicable to a number of potential future instruments.

Performance of the DSP subsystem was evaluated in terms of measured vs. theoretical execution speed, timing accuracy, power consumption, and computational accuracy.

TABLE OF CONTENTS

1. INTRODUCTION

Measurement of Ocean Vector Winds by Radar Scatterometry

Radar scatterometry can be used to determine the wind speed and direction over the world's oceans by transmitting microwave pulses from an orbiting instrument, and measuring the power of the received signal backscattered (reflected) from the ocean surface. After downlinking the power measurements, ground-based processing algorithms convert multiple power measurements made at multiple incidence angles on a given ocean location to estimates of surface wind speed and direction [1].

To reduce the bandwidth of the data stream downlinked from the instrument, some processing is done in the instrument. The primary job of on-board signal processing is to produce the power measurements. Signal processing operations necessary to provide power measurements include vector multiplication, digital filtering, decimation, fast Fourier transform (FFT), and summation.

---

[1] 0-7803-7231-X/01/$10.00/© 2002 IEEE

Traditional and Future Approaches to On-board Signal Processing

Scatterometer instruments such as SeaWinds and NSCAT have performed the required signal processing using specialized hardware which was developed specifically for the scatterometry application [1] [2]. On-orbit performance of the NSCAT and SeaWinds/QuikSCAT instruments have demonstrated that such hardware is capable of performing the signal processing tasks required to produce high-quality ocean wind vector measurements [3] [4]. However, the appearance of general-purpose programmable digital processors (DSPs) in flight-quality versions leads to the possibility that future scatterometers could benefit from the flexibility that they offer [5]. Thus, use of programmable DSPs are being considered for the next scatterometer instrument.

The Need for a Testbed, and Testbed Findings

Since the use of programmable DSP represents a departure from previous scatterometer architectures, the approach has been met with a certain amount of skepticism. This is understandable, given the fact that use of "heritage" hardware is a typical approach taken to reduce project technical risk. Without prior experience in implementation of programmable DSP-based subsystems, programmatic risk, specifically to cost and schedule, are also potentially increased.

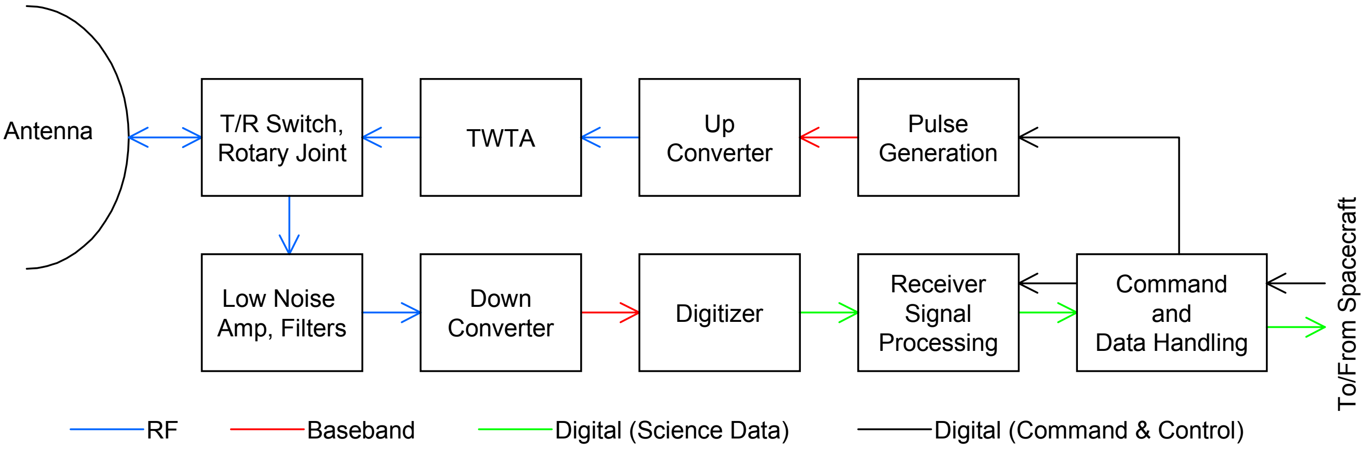

Figure 1. Typical Scatterometer Instrument

A testbed was constructed to provide an environment in which potential programmable DSP-based scatterometer instrument architectures could be breadboarded, integrated and evaluated. This testbed contains several programmable DSPs, data converters, data and command interfaces, and radio frequency (RF) hardware. Working in the testbed gives the software developers experience in working with the peculiarities of actual RF and digital hardware, rather than blindly coding to a (hopefully correct!) software specification.

Implementation of an end-to-end scatterometer breadboard in the testbed has not only shown the feasibility of the use of programmable DSPs for the scatterometer signal processing tasks, it has allowed the team to gain experience in the process of developing DSP hardware and software. The testbed experience has thus reduced the technical and programmatic risk of future programmable DSP-based scatterometers.

2. SCATTEROMETER INSTRUMENT

A simplified block diagram of a typical scatterometer instrument is given in Figure 1. The command and data handling subsystem (C&DH) controls the generation of the baseband radar pulse, and controls the receiver signal processing tasks. It also handles the command and science data interfaces between the instrument and the spacecraft.

In the transmit chain, the up converter performs a frequency translation of the baseband radar pulse (<40 MHz) to Ku band (13.402 GHz). The traveling wave tube amplifier (TWTA) amplifies the signal to the required transmit level (about 120 W). The rotating antenna is used for both transmitting and receiving the radar signal.

The NSCAT instrument used interrupted continuous wave (CW) pulses about 5 ms long. SeaWinds uses linear FM chirped pulses from 0.5 to 2.5 ms long. Linear FM chirped pulses are planned for use on future instruments as well. Between transmissions, the transmit/receive switches are configured to direct the radar echo from the ocean surface to the down converter. The typical pulse repetition interval (PRI) is 5.4 ms, which is determined by the orbital height of 800 km, so that the nadir (undesirable) echo arrives *during* the next transmit pulse. The scattered echo (signal of interest) will arrive *between* transmit pulses.

In the receiver chain, the echo is down converted to baseband. The signal is then digitized and processed. The processed science data is packetized by the C&DH and forwarded to the spacecraft for subsequent downlink to the ground.

3. SCATTEROMETER SIGNAL PROCESSING TASK

The signal processing tasks on the radar transmitter are primarily concerned with the generation of the linear swept FM chirp at baseband. Transmission of a linear FM chirp, combined with the dechirp done by the receiver, produces a baseband signal in which frequency is related to the distance from the instrument to the earth. This can be used to improve the resolution of the backscatter measurements [2].

The radar receiver calculates a power spectrum which is further processed to provide estimates of the power in a series of frequency bands, which correspond to range bins.

Transmitter

In the SeaWinds scatterometer, generation of the linear swept FM chirp is done using numerically controlled oscillator (NCO) techniques. A future instrument containing a programmable DSP could generate the numerical representation of the linear swept FM chirp using the equation:

$$I[n] = A\cos\left(\frac{\omega_0 n}{r} + \frac{\alpha n^2}{2r^2}\right) \qquad (1)$$

where n represents the sample number, $I[n]$ is the value of the nth sample, A is the amplitude, ω_0 is the initial frequency in radians/s, and α is the angular acceleration in radians/s^2, and r is the sample rate in sample/s. This data stream could then be sent to a digital-to-analog converter (DAC) to produce the baseband chirp[2]. Equation 1 could be modified to accommodate for non-linear group delay in the transmitter. Equation 1 can also be used to generate the dechirp signal used in the receiver echo processing.

Receiver

Measurement of the power of the backscattered radar signal involves several signal processing steps. Performance of part or all of the receiver echo processing tasks by programmable DSP is being considered for a future instrument. A baseline concept for signal processing using programmable DSP was developed. Exact details of the echo processing requirements of a future instrument have not yet been established, but it is felt that the system of Figure 2 is representative of future polarimetric instruments.[3]

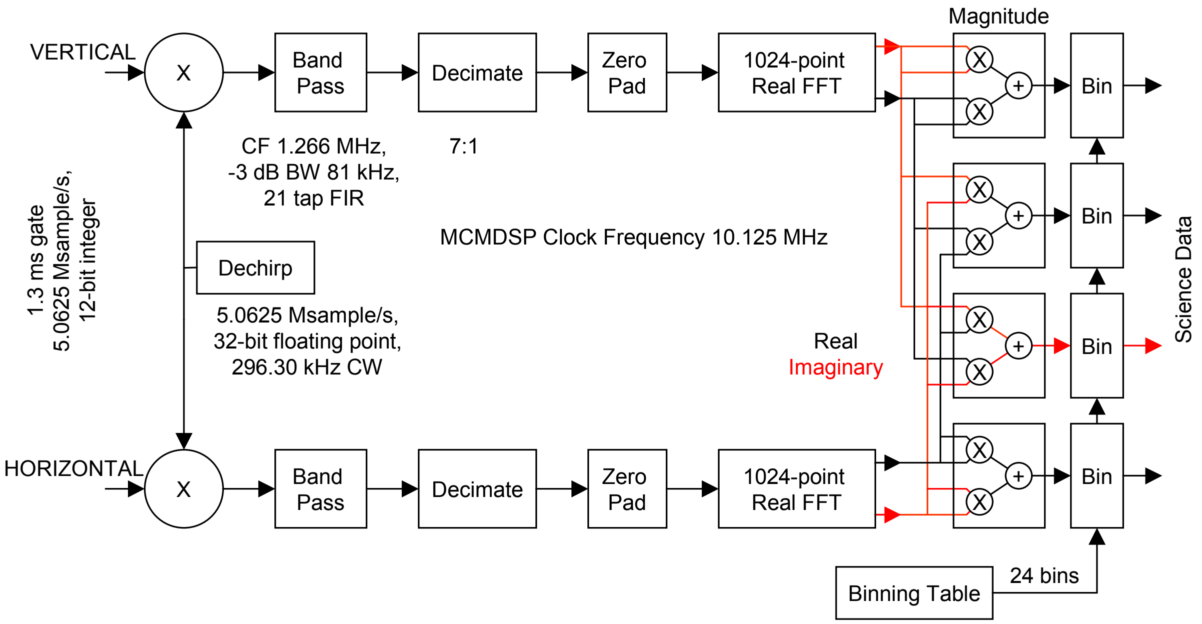

Figure 2. Baseline Concept for Polarimetric Echo Processing

4. DEMONSTRATION ELEMENTS

It was felt that confidence in the approach of using a programmable digital signal processor for the signal processing functions of a future scatterometer instrument could be gained by demonstration of a number of key technical and programmatic elements. These include:

---

[2] The frequency ω_0 is different for each pulse, to compensate for the Doppler shift caused by the movement of the instrument relative to the earth. Thus, all echoes are received at the same frequency.

[3] A polarimetric scatterometer is an instrument in which both vertically and horizontally polarized components of the radar echo are processed. Such an instrument has the potential for improved wind retrieval performance. [6]

Technical Elements

Signal processing– Can the programmable DSP perform the appropriate signal processing steps?

Startup of the programmable DSP– What issues are involved in starting the DSP and loading the application software?

Utilization of multiple programmable DSPs– How can the total signal processing job be efficiently divided among several programmable DSPs? How can these DSPs coordinate the signal processing tasks?

Commanding of the programmable DSP– How can the host processor control the signal processing done by the programmable DSP?

Delivery of science data– What interface is appropriate for the transfer of the science data to the host processor?

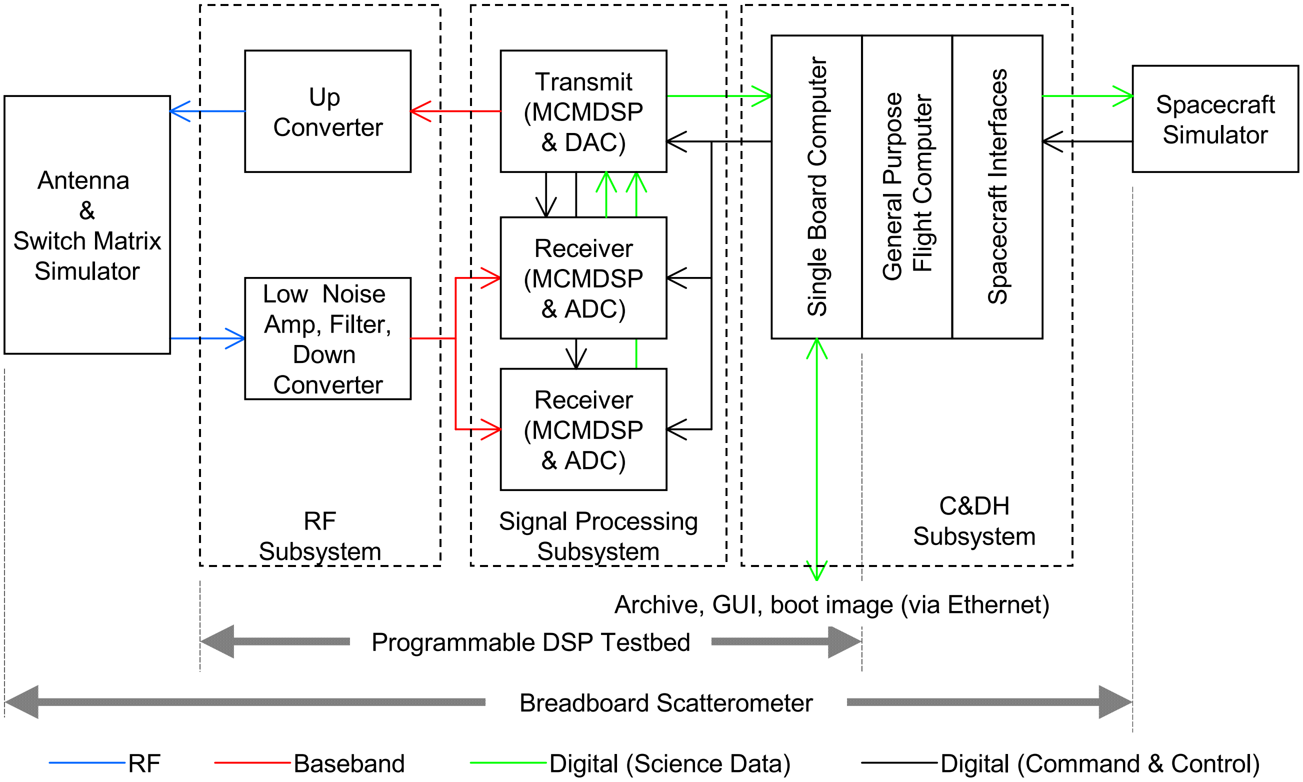

Figure 3. Programmable DSP Testbed and Breadboard Scatterometer

Interface between signal processing and RF subsystems– What interfaces are appropriate to the RF subsystem?

Programmatic Elements

Software development environment– What tools and practices are necessary to manage the development of software for a programmable DSP? How can such software be debugged?

Skills mix– What skills are necessary among the team members? How can DSP programming skills be acquired?

5. PROGRAMMABLE DSP TESTBED

A programmable DSP testbed was constructed to allow investigation of these key technical and programmatic elements. This programmable DSP testbed forms part of a larger ongoing effort, construction of a breadboard scatterometer. The programmable DSP testbed, in the context of the breadboard scatterometer is shown in Figure 3.

The breadboard scatterometer development task combines the efforts of JPL's Avionics group, traditionally responsible for scatterometer command and data handling subsystems, and the Radar group, who have handled the radar transmitter and receiver subsystems of the instrument. It is anticipated that co-location of the avionics and radar personnel during the construction of the breadboard will result in better partitioning of the instrument functionality between the radio frequency, signal processing, and command and data handling subsystems. Use of commercial off the shelf hardware (COTS) allowed more emphasis to be placed on overall instrument system architecture and functionality and development of appropriate interfaces between subsystems, rather than getting into the details of circuit design for each component of the system.

Other elements of the breadboard scatterometer, not covered in detail in this paper, include the RF subsystem [7] and the C&DH, to be developed in the upcoming months.

Signal Processing Subsystem

The programmable DSP-based signal processing subsystem under evaluation consists of three Astrium multi-chip module digital signal processors (MCMDSP) mounted on evaluation boards. Available in a spaceflight-qualified version, the MCMDSP contains program and data memory, serial and parallel input/output ports with first-in-first-out (FIFO) buffers, and debugging ports that make it an attractive candidate for a future scatterometer [5]. Each MCMDSP and its evaluation board is mounted in a holding fixture (Figure 4). Custom-built transition boards provide the drivers and receivers for the RS-422 and LVDS interfaces. A commercial data conversion board provides two channels of digital-to-analog conversion (DAC) and two channels of analog-to-digital conversion (ADC)[4]. A functional block diagram of the components of the holding fixture is shown in Figure 5. The data conversion direction (DAC or ADC) is controlled by configuration data loaded into a field programmable gate array (FPGA) on the data conversion board. Two of our MCMDSPs are configured with ADCs for use as receivers, the third MCMDSP is configured with DACs for use as a transmitter. In the future, we may reconfigure the data converters during operation under software control.

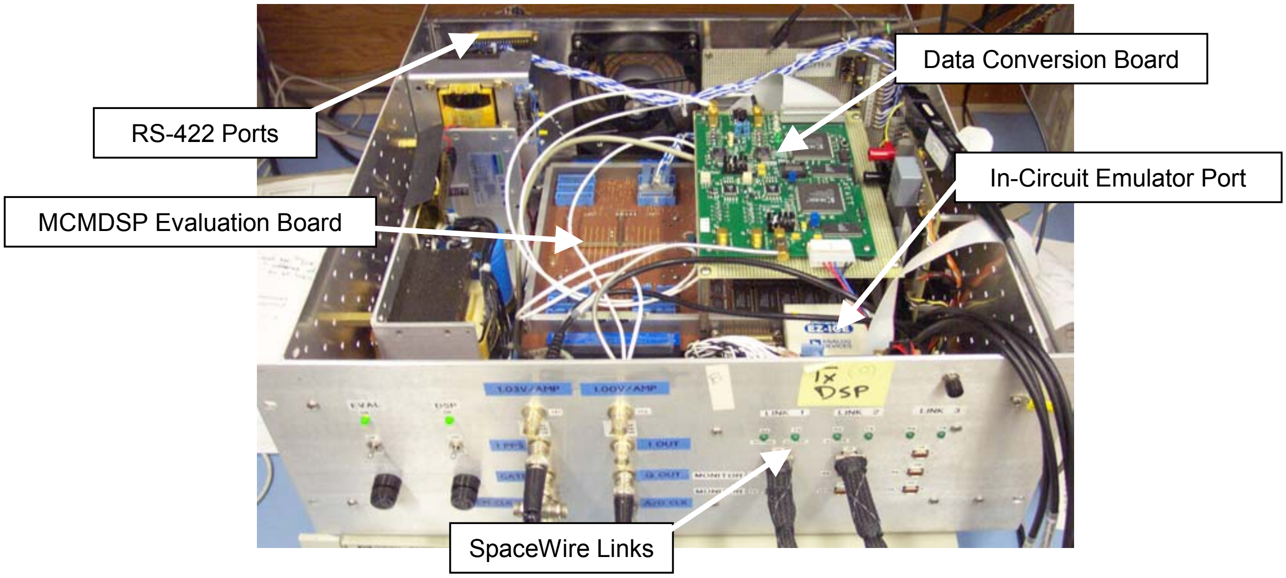

Figure 4. MCMDSP Holding Fixture

The MCMDSP contains the radiation tolerant Temic TSC21020E programmable DSP, which is functionally identical to the ADSP21020, a commercial part from Analog Devices[5]. The 21020 can execute two arithmetic operations per clock cycle. The MCMDSP is capable of 40 million floating point operations per second (MFLOPS) when operated at its maximum clock speed of 20 MHz.

Transmitter–

Quadrature sample pairs (I/Q) are clocked out to a pair of DACs at a sample rate of 10.125 MHz to generate linear swept FM chirps at baseband (3.5 MHz in the baseline concept). Even at 40 MFLOPS, the MCMDSP can not "keep up," with direct calculation of Equation 1 and its *sine* counterpart, so a look-up table approach is used. The MCMDSP calculates the values of $I[n]$ and $Q[n]$ for several thousand values of n at system initialization, before actual radar operation commences. The values of $I[n]$ and $Q[n]$ are stored in a look-up table. During operation of the radar transmitter, the $I[n]$ and $Q[n]$ values are retrieved from the look-up table, and passed directly to the DACs at 10.125 Msamples/s.[6] The pulse-to-pulse Doppler compensation can be performed by changing the index to the lookup table.

---

[4] Data conversion boards are model GVA-200 by GV & Associates, Inc., Ramona, California.

[5] Before we obtained the MCMDSP evaluation boards, commercial ADSP21020-based DSP boards were used for initial algorithm development (model IXD7232 from Ixthos, Inc., Leesburg, Virginia).

[6] The transmit sample rate of 10.125 MHz was chosen to be a convenient sub-multiple of the stable local oscillator used in the RF subsystem, and to be high enough to make the DAC output filter simple and non-critical to design.

Since all values of *I[n]* and *Q[n]* are precalculated, the speed of the transmit DSP is not critical, as long as the values can be retrieved from the look-up table at a rate sufficient to feed the DACs (10.125 Msamples/s).

Receiver–The scatterometer operates at a pulse repetition interval of about 5 ms. About 1.3 ms of this time is spent actually sampling the baseband received signals. Early estimates of processing time indicated that signal processing for each echo would take longer than the remaining 3.7 ms. Thus, the signal processing task is divided among several receivers, with each receiver performing all the processing for a given echo as shown in Figure 2. All signal processing is done with 32-bit floating point precision. All receivers run identical software.

The baseband echo signals (horizontal and vertical) are each sampled at a rate of 5.0625 Msample/s. Because of the violation of the Nyquist criterion, the signal of interest at 3.5 MHz is aliased to 1.5625 MHz by the sampling process. This data set (about 6,600 samples) is mixed (multiplied) with a dechirp waveform[7] at 296.30 kHz to produce the vector product, which contains the familiar sum and difference frequencies at 1.8588 MHz and 1.2662 MHz.

The difference frequency at 1.2662 MHz is selectively favored by a 21-tap finite impulse response (FIR) digital bandpass filter, centered at 1.266 MHz. To reduce computational load and increase the resolution of the upcoming FFT, the data stream is decimated by a factor of 7, producing a data stream effectively sampled at 723.21 kHz.

This data stream is zero padded to 1,024 samples, and processed by a real FFT function which produces 1,024 real and imaginary frequency components (cells). The first 512 real and imaginary cells are multiplied in the combinations shown to produce the polarimetric backscatter power measurements. A binning operation sums the power in a number of adjacent frequency cells to produce the science data product.

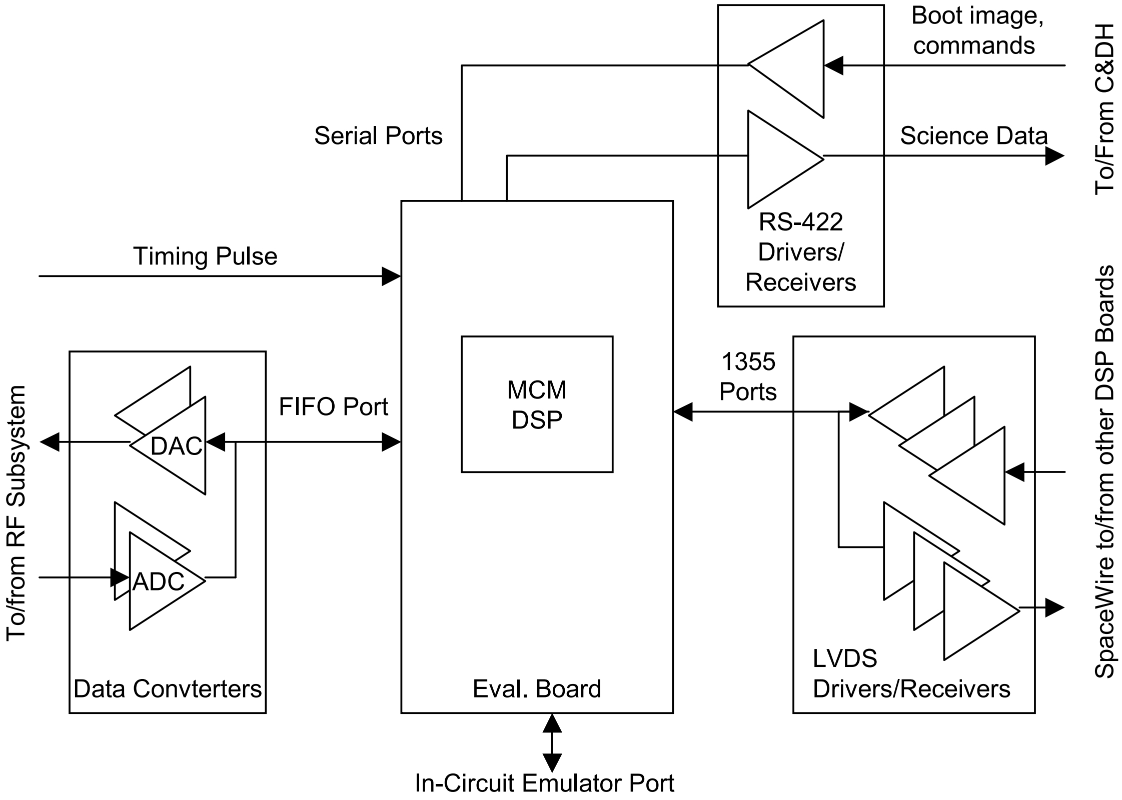

Figure 5. Holding Fixture Components

Power consumption of the DSP is a measure of how many simultaneous operations are being executed, and thus can be used as a diagnostic tool. Figure 6 shows the power dissipation of one receiver MCMDSP during processing of one echo. The complete data acquisition and signal processing takes about 13 ms. Thus, to "keep up" with echoes at a PRI of 5 ms would require at least three receiver DSPs[8]. The observed execution time of 2.2 ms for the two

---

[7] The signal of interest is actually at about 37 MHz, but for convenience, the receivers are currently being evaluated using continuous wave (CW) test tones at 3.5 MHz. Thus, we are using a CW signal in the dechirp stage (i.e. $\alpha = 0$ in Equation 1).

[8] The signal processing time would decrease proportionally from that shown in Figure 6 as the MCMDSP clock frequency is increased. However, even at the maximum clock frequency of 20 MHz, more than one MCMDSP would be required.

1024-point real FFTs agrees with that predicted by the developers of the rfft2.asm algorithm (see "DSP Software Development Environment," below).

Qualitative evaluation of the receiver results (binned output) and quantitative evaluation of the individual signal processing elements (dechirp, filter, FFT, bin) have given us confidence that the programmable DSP approach can perform the scatterometer signal processing tasks.

Startup of the GP DSP

Since the MCMDSP contains no non-volatile memory, it is necessary to perform program memory initialization (boot up) after application of power. This is accomplished via a synchronous serial interface using RS-422 signaling levels.

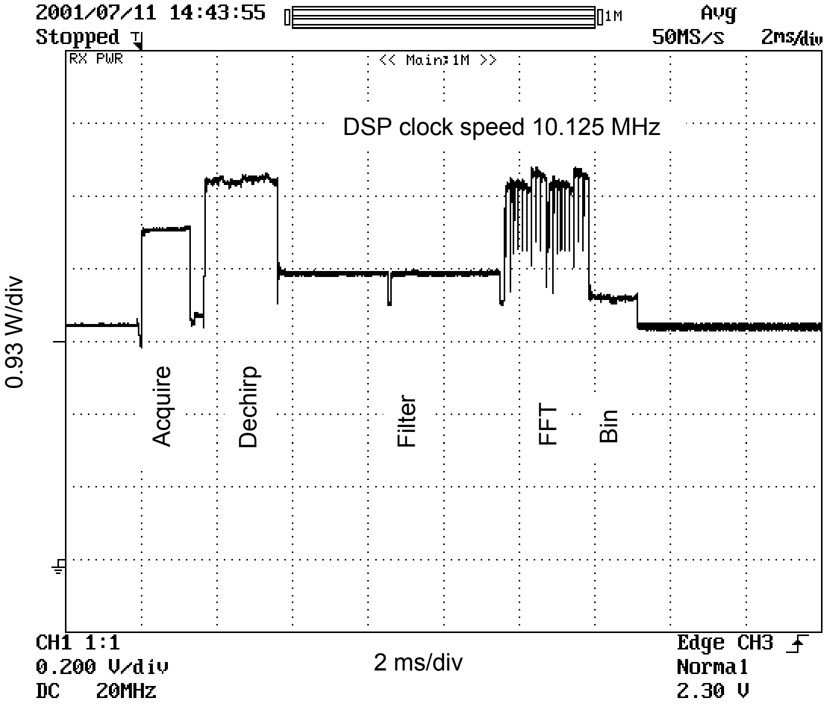

Figure 6. MCMDSP Power Dissipation

Software running on a Motorola 68060-based single-board computer (SBC) equipped with an Industry Pack digital interface module reads the DSP boot image from a file server via Ethernet, and then toggles the enable, clock, and data lines to send the boot image to the MCMDSP. The MCMDSP stores the incoming boot image in program memory. At the conclusion of the load process, the processor begins execution of the program. The ability to boot over a serial link in this manner is a feature of the DSP peripheral controller (DPC) chip in the MCMDSP.

Our current boot image for the MCMDSP transmitter, about 134 Kbytes, is sent in about 1.15 s, a rate of about 120,000 bits/s. It is anticipated that the transfer rate will be improved in an actual instrument implementation when dedicated interface hardware will control the enable, clock, and data lines, rather than the software-based "bit-banging" technique used in the testbed.

Utilization of Multiple Programmable DSPs

Because the signal processing requirements for the transmitters and receivers are greater than that which can be handled by a single MCMDSP, it was necessary to show that the task could be divided among several programmable DSPs, and that the communication between processors necessary to coordinate the task is possible.

Task partitioning was done as follows: A single MCMDSP performs the transmit signal processing, and schedules the receivers. Additional MCMDSPs perform the receive signal processing, with each doing the complete processing for a given echo. Having each receiver perform the processing for both the horizontal and vertical polarization components from a given echo eliminates any need for the receiver MCMDSPs to share data among themselves. Our current testbed has two MCMDSP receivers, but the number could be increased as necessary to accommodate the desired PRI [5].

Communication between the MCMDSPs is done over SpaceWire serial links. Popular in the European spacecraft community, the SpaceWire standard uses low-voltage differential signaling standard and is based on the IEEE-1355 standard [8].

The transmit DSP is the system master; it sends commands to each receiver to tell it when to acquire and process each echo. After processing each echo, the receiver sends the science data via SpaceWire to the transmit DSP, which forwards it via an asynchronous serial interface to the C&DH.

Timekeeping between the MCMDSPs is established and maintained by a combination of a periodic pulse signal fed simultaneously to hardware interrupt lines on all MCMDSPs, and time messages sent between the MCMDSPs over the SpaceWire links. Time is kept independent of the pulse repetition interval. Relative timing precision of events between the transmitter and receiver DSPs is generally maintained to +/- 7 cycles of the processor clock. At a processor clock frequency of 10.125 MHz, this is about +/- 700 ns.

Commanding of the Programmable DSPs

The C&DH subsystem sends commands to the transmit MCMDSP via the same synchronous serial RS-422 interface used to perform program memory initialization. Commands are not sent from the C&DH to either of the receiver MCMDSPs. Commands currently implemented include changing the PRI, and turning on or off the transmit pulses.

Science Data Delivery

The science data product from the DSP subsystem is sent from the transmit DSP to the controls subsystem via a 1.25 Mbaud asynchronous serial interface. The C&DH uses an Industry Pack serial interface module on a Motorola 68060-based single board computer to receive the data. For archival and analysis purposes, software running on the SBC directs the data to a file server via an Ethernet connection. The SBC also transmits the science data over Ethernet via user datagram protocol (UDP) to a personal computer. LabVIEW-based graphical user interface (GUI) software running on the personal computer displays the science data in histogram format for immediate interpretation by engineering personnel.

Interfaces Between Signal Processing and RF Subsystems

In the existing SeaWinds instruments, flight software running on a general-purpose computer in the C&DH subsystem is synchronized to a pulse signal synchronous to the radar PRI. The C&DH subsystem sends a serial command to the pulse generator circuitry before every radar pulse. The design of the C&DH flight software is thus somewhat complicated because it has periodic duties operating on several time scales: commanding of the pulse generator synchronized with the PRI (~5 ms), formation of the science data packet every 100 pulses (~500 ms), and collection of engineering data (instrument temperatures, voltages, etc.) once per second.

Use of programmable DSPs to perform the instrument functions synchronized to the PRI (pulse generation, echo processing) will significantly reduce the complexity of the software required for the C&DH subsystem. In the breadboard scatterometer, the interfaces between the signal processing subsystem and the RF subsystem are analog baseband signals, and the signal processing subsystem handles the pulse-synchronous tasks.

DSP Software Development Environment

The Temic TSC21020E DSP contained in the MCMDSP is functionally identical to the Analog Devices ADSP21020. Thus, the development tools used to support use of the ADSP21020 in the commercial marketplace can be used to develop and test software on the MCMDSP.

The majority of our DSP software is written in C, using the Analog Devices ADSP-21000 Family C Tools. This toolset includes the G21K compiler, based on the Free Software Foundation's GCC compiler. A set of C runtime library functions for floating point, digital signal processing and C operations is also included. A C source-level debugger (CBUG) and PC-based software simulation of the ADSP21020 are also included, although we seldom use these tools, because we can readily test our code on the target hardware.

For optimum performance, it was necessary to write some of the code in 21020 assembly language. Interrupt service routines that interact with the ADCs and DACs, as well as the dechirp, filter, decimate, magnitude, and binning signal processing functions are all written in assembly. The assembly code for the 1024-point real FFT function, optimized for the 21020, was obtained from the on-line Analog Devices Technical Support Applications Handbook[9].

Tools useful for debugging the application software include the Analog Devices EZ-ICE in-circuit emulator. Attached to the JTAG port on the MCMDSP evaluation board, it allows the software developer to examine all of the major components in the MCMDSP hybrid, including the TSC21020E registers, data and program memory, the IEEE-1355 interface controller, and the digital I/O controller. The EZ-ICE can also be used to load application software, and set breakpoints to halt execution when certain memory spaces are accessed. All of the Analog Devices tools run

---

[9] Source code for the real FFT function rfft2.asm and many other signal processing and mathematical functions is available at http://www.analog.com/techsupt/software/dsp/app_note/21k_books.html.

under Windows 95 on a personal computer (PC). Because the ADSP21020 is no longer widely used in new commercial designs, the development tools have not been updated to operate on newer versions of the Windows operating system. As one might imagine, finding PC hardware and software support for Windows 95 is becoming an increasing challenge.

Other PC-based software tools used in the development of MCMDSP software include QED-1000 digital filter design software[10] used to calculate coefficients for the digital FIR filter, and MATLAB,[11] used to simulate portions of the signal processing code for verification of accuracy of the DSP algorithms.

Configuration control of the source code for the DSP software is managed by WinCvs[12]. This program provides an organized method for coordinating contributions of multiple software developers to a software project.

Skills Mix

Although several members of the hardware and software engineering team were involved in the development of the NSCAT and SeaWinds instruments, the team's collective experience with using a general-purpose programmable DSP and developing the necessary software was somewhat limited. System engineering was a collaborative effort between engineers from the radar and avionics groups. The overall DSP software architecture was led by an engineer with previous DSP experience, while detailed coding was performed by two software engineers with no prior DSP experience. Early in the program, one of our software engineers and one of our test engineers attended a week-long technical course on the 210xx family of processors, presented by Analog Devices. Although a certain amount of trial and error experimentation was necessary, the experience gained at that class helped us to get over several of the hurdles that might have hindered our initial efforts: How to write, compile, load, and execute code on the target hardware, how to write 21020 assembly code, how to write interrupt service routines. A third software engineer, working part-time, developed the C&DH software.

Hardware engineers, also assigned part-time to the project, designed and tested the physical layer interface circuits for the RS-422 and SpaceWire interfaces. Two additional radar hardware engineers designed and built the breadboard scatterometer RF subsystem. Coordination between the personnel working on the breadboard scatterometer was handled at weekly all-hands meetings, and at splinter sessions conducted in the lab on an as-needed basis.

6. DSP TESTBED DEMONSTRATION OUTCOME

In July of 2001, after approximately one and one-half years of work, a technical presentation of the DSP subsystem of the breadboard scatterometer was given to a JPL review board consisting of technical personnel both from within and outside the scatterometer project. Although the review board recommended that other options for signal processing (such as field-programmable gate-arrays) be further investigated with respect to mass, power, and cost, they agreed with the DSP testbed team that a general-purpose programmable DSP, such as contained in the MCMDSP, could perform the scatterometer signal processing task.

Future Plans

Plans related to the DSP hardware development include completing the integration of the DSP-based signal processing subsystem with the breadboard RF subsystem to allow end-to-end evaluation of system performance. Design, fabrication, and test of a prototype circuit board, containing the components contained in the evaluation fixture (MCMDSP, data converters, physical layer interfaces) may also be pursued. Such a prototype would allow investigation of thermal issues associated with use of the MCMDSP, and allow early identification of any signal integrity (noise) problems related to increasing the clock speed to 20 MHz or co-location of the data converters with the MCMDSPs.

On the software side, plans include development of a numerical simulation of the entire signal chain, to facilitate signal processing tradeoffs and verification of end-to-end numerical accuracy. The issue of fault detection and protection will also be addressed.

7. CONCLUSIONS

A signal processing subsystem, based on a general-purpose programmable MCMDSP, was developed. This system was demonstrated in a testbed and found to be capable of performing the scatterometer signal processing task. The experience gained by the collaborative avionics/radar testbed team will allow them to develop the signal processing subsystem for a possible future instrument.

ACKNOWLEDGEMENTS

The work performed in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

---

[10] QED-1000 is a product of Momentum Data Systems, Inc., Fountain Valley, California.

[11] MATLAB is a product of The MathWorks, Inc., Natick, Massachusetts.

[12] WinCvs is available at http://www.cvsgui.org/download.html.

The authors would like Kouji Nishimoto, Minh Lang, and Alex Bachmann for DSP programming and software testing; Ben Wilkinson for C&DH programming; Dorothy Stosic for RF subsystem design; Yeghiya Mahjoubian for digital filter design and numerical modeling; and Rich Steffke and Steve Petree for hardware interface designs.

We would also like to thank Astrium, specifically Pierre-Eric Berthet, for the extended loan of the MCMDSP hybrids on the evaluation boards.

REFERENCES

[1] F. M. Naderi, M. H. Freilich, and D. G. Long, "Spaceborne Radar Measurement of Wind Velocity Over the Ocean-An Overview of the NSCAT Scatterometer System," *Proceedings of the IEEE, Vol. 79, No. 6,* June 1991, pp. 850-866.

[2] Michael W. Spencer, Chialin Wu, and David Long, "Improved Backscatter Measurements with the SeaWinds Pencil-Beam Scatterometer," *IEEE Transactions on Geoscience and Remote Sensing, Vol. 38, No. 1,* January 2000, pp. 89-104.

[3] Wu-yang Tsai, James E. Graf, Carroll Winn, James N. Huddleston, Scott Dunbar, Michael H. Freilich, Frank T. Wentz, David G. Long, and Linwood Jones, "Postlaunch Sensor Verification and Calibration of the NASA Scatterometer," *IEEE Transactions on Geoscience and Remote Sensing, Vol. 37, No. 3,* May 1999, pp. 1517-1542.

[4] David G. Long, Andrew S. Fletcher, and David W. Draper, "SeaWinds Wind Retrieval Quality Assessment," *IEEE International Geoscience and Remote Sensing Symposium 2000 Proceedings, Vol. 3,* pp. 1036-1038.

[5] Alex Bachmann, Douglas Clark, James Lux, and Richard Steffke, "Multiprocessor Digital Signal Processing on Earth Orbiting Scatterometers," *2001 IEEE Aerospace Conference Proceedings,* March 10-17, 2001.

[6] Michael W. Spencer, James N. Huddleston, and Bryan W. Stiles, "Advanced Design Concepts for a SeaWinds Scatterometer Follow-On Mission," *2001 IEEE Aerospace Conference Proceedings, March 10-17,* 2001.

[7] Dorothy K. Stosic and James P. Lux, "Ku-Band Receiver and Transmitter for Breadboard DSP Scatterometer," *2002 IEEE Aerospace Conference Proceedings,* March 9-16, 2002.

[8] Alex Bachmann, "SpaceWire on Earth Orbiting Scatterometers," *2002 IEEE Aerospace Conference Proceedings,* March 9-16, 2002.

Doug Clark *is a member of the Flight System Engineering Group within JPL's Avionics System Engineering Section. He earned Bachelor's and Master's degrees in Physics from San Diego State University and California State University, Los Angeles, respectively. At JPL he has been involved in the test and integration of command and data handling subsystems for scatterometers, and is currently contributing to the system design for future instruments and spacecraft.*

James Lux *is in the Spacecraft Telecommunications Section at JPL. Scatterometer design is only one of Jim's diverse interests, which range from man-made tornadoes to solar eclipses to high performance digital radio links.*

Mike Shirbacheh *is Project Element Manager for the SeaWinds Command and Data Handling subsystem. Mike is a member of the Avionics System Engineering Section of the Avionics Systems and Technology Division of the Jet Propulsion Laboratory.*