Design and performance of the Astro-E/XRS signal processing system

K. R. Boyce[a], M. D. Audley[a,b], R. G. Baker[a], J. J. Dumonthier[a],
R. Fujimoto[c], K. C. Gendreau[a,b], Y. Ishisaki[d], R. L. Kelley[a],
C. K. Stahle[a], A. E. Szymkowiak[a], and G. E. Winkert[a]

[a]NASA Goddard Space Flight Ctr., Greenbelt, MD

[b]University of Maryland, College Park, MD

[c]Institute of Space and Astronautical Science, Sagamihara, Japan

[d]Tokyo Metropolitan Univ., Tokyo, Japan

ABSTRACT

We describe the signal processing system of the Astro-E XRS instrument.[1] The Calorimeter Analog Processor (CAP) provides bias and power for the detectors and amplifies the detector signals by a factor of 20,000. The Calorimeter Digital Processor (CDP) performs the digital processing of the calorimeter signals, detecting X-ray pulses and analyzing them by optimal filtering. We describe the operation of pulse detection, pulse height analysis, and risetime determination. We also discuss performance, including the three event grades (hi-res, mid-res, and low-res), anticoincidence detection, counting rate dependence, and noise rejection.

Keywords: X-ray astronomy, Pulse Detection, Digital Signal Processing, XRS, ASTRO-E

1. INTRODUCTION

XRS[1] is a high-resolution, high-throughput X-ray spectrometer to be launched on the Japanese ASTRO-E[2] mission early in 2000. It will have resolution of approximately 12 eV FWHM over the range from about 0.4 keV to 10 keV. As described in Reference 4, the XRS detectors are microcalorimeters, which measure the temperature pulse which occurs when an X-ray is absorbed by the detector. Each such pulse lasts on the order of 100 ms, and its height is approximately proportional to the X-ray energy. To measure the energy of an X-ray photon, then, we need to measure the height of the resulting pulse. In this paper, we discuss the on-board processing used to measure these pulseheights with the best possible resolution.

A block diagram of the XRS electronics is shown in Figure 1. The detectors are located inside the Front End Assembly (FEA) which is inside the Dewar. The Dewar contains an outer layer of solid neon (at 17 K) and an inner layer of liquid helium (at 1.3 K). Signals from the detectors are amplified by the Calorimeter Analog Processor (CAP) and then sent to the Calorimeter Digital Processor (CDP). The CDP detects pulses and extracts a best estimate of the pulseheight. This estimate is time-tagged and sent to the spacecraft for transmission to the ground.

The Adiabatic Demagnetization Refrigerator (ADR) and its controller, the ADR Control and Housekeeping Electronics (ACHE), keep the detectors at at fixed temperature of 60 milliKelvin. In this presentation we will discuss only the CAP and CDP.

2. ANALOG ELECTRONICS

The Calorimeter Analog Processor (see Figure 2) provides power and bias to the detectors, and amplifies the signals by 20,000 before passing them on to the digital processor. It also controls the temperature of the first-stage JFETs. The CAP is broken into two identical and independent sides, labeled A and B, each of which handles 16 channels.

The cold JFETs provide impedance buffering between the detectors (with a resistance of about 10 MΩ) and the wiring harness. They are inside the Front End Assembly, but must be kept at 130 K. This temperature is controlled by the CAP. To minimize heat conduction to the 60 mK stage, the FETs are contained in a nested pair of boxes,

---

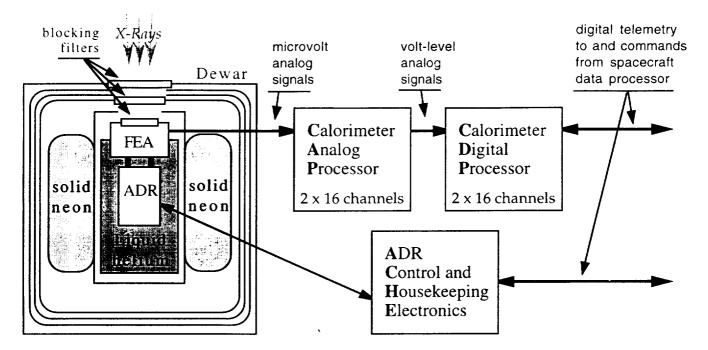Send correspondence to K.R.B.: E-mail Kevin.R.Boyce@gsfc.nasa.gov

Figure 1. XRS block diagram. The FEA (Front End Assembly) contains the detectors, two blocking filters, and the first stage of electronics (the cold JFETs). The ADR (Adiabatic Demagnetization Refrigerator) controls the temperature to 60 mK.

each of which is individually heat-sunk (the inner one to the 17 K solid neon Dewar, and the outer one to the 1.3K liquid helium).

There are 4 bias supplies, each of which supplies the bias voltage to 8 channels of detectors. Likewise, there are 4 V_{dd} supplies for the cold JFETs.

Each calorimeter channel has its own amplifier circuit, implemented on a hybrid circuit package. The amplifiers have an AC gain of 20,000 and a highpass knee of 10 Hz. Their output is a full differential signal (maximum 24 V), which is sent to the CDP. The temperature stability of the amplifiers is better than 200 ppm/°C. The signals are AC coupled, with a single pole at 10 Hz. This is necessary because the DC level of the input is about 0.5 V, so the DC gain can't be 20,000.

3. WHY PROCESS ON-BOARD?

The signals from the detectors (see Reference 4) are low-level voltage pulses, as shown in Figure 3 (In reality the pulses are negative-going, but the CDP inverts them for convenience.)

The signals from the XRS detectors contain useful information up to about 2kHz, so we need to sample at a minimum of 4kHz. Thus to transmit the data from all 32 channels with 14-bit resolution would require 1800 kBits/sec. However, the average telemetry bandwidth available to XRS is 10 kbits/sec, so clearly we need to convert the raw data to pulseheights on the satellite. The CDP performs this job, by detecting pulses in the data stream and analyzing them on board.

4. CDP OVERVIEW

The CDP is built as 32 independent channels, one per pixel, all running the same software. It also contains common circuitry to handle communication with the spacecraft. Figure 4 shows a block diagram of the CDP.

Each channel contains an Analog-to-Digital section with an antialias filter. Like the CAP, the CDP is built as two independent sides, each handling 16 channels.

Each channel is responsible for a single pixel in the detector array. The CDP processing detects pulses, processes them, and sends the data to the spacecraft. The data from each X-ray pulse contains the pulseheight, the risetime of

the pulse, a measure of the pulse shape, a time tag with 5 μs resolution, and the pixel number. It also includes flags to indicate event grade, whether the anticoincidence detector fired, whether the pulse is electrically clipped at the A/D converter, and whether it is a "baseline" pulse. Baseline pulseheights are created by processing a noise record as if it were a pulse, and are used to obtain a measure of the noise in the system.

4.1. Code

All channels of the CDP run the exact same code. When the CDP is initially powered up, each channel loads a copy of the code from the master card, and begins executing it. The master has two copies of the code: one in PROM and one in EEPROM. The code to be loaded by the channels is selected by a hardware-decoded command from the ground.

Normally we use the code in EEPROM. This allows us to update the code at any time. However, the power-on default is to use the PROM code. The code in PROM is there as a backup in case the EEPROM should fail. It is fully functional, but is missing some of the features in the latest EEPROM version.

5. PULSE DETECTION

The first job of the DSP is to detect pulses in the data stream. Detecting initial pulses is easy; the hard part is finding small secondary pulses on top of large ones. The reason is shown in Figure 5, which is a 200eV secondary pulse 2 milliseconds after a 10keV initial pulse.

As you can see, the secondary pulse is barely noticeable. However, it is still a valid signal which should go into the spectrum. Furthermore, if it were not detected, the calculated pulseheight of the 10keV pulse would be affected. Thus it is very important that the CDP can detect secondary pulses even when they are much smaller than the initial pulse.
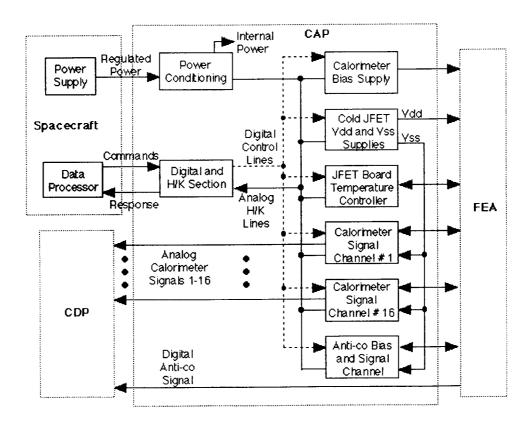

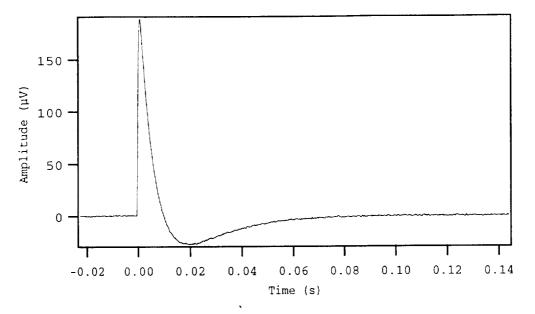**Figure 2.** Block diagram of the Calorimeter Analog Processor (CAP).

Figure 3. A typical pulse from one detector. The X-ray raises the temperature of the thermistor, thus lowering its resistance. This causes a drop in the voltage across it. For clarity, the pulse is inverted by the CDP. The pulse overshoots its baseline because of the AC coupling in the CAP.

5.1. Derivative

The pulse detection algorithm is based on using a smoothed derivative of the data. The derivative is calculated by convolving the data with a "boxcar derivative" function, as shown in part (A) of Figure 6. The number of samples (the length of the boxcar) is an adjustable parameter.

This convolution has the effect of simultaneously smoothing and differentiating the data. We want to use the derivative so that slow changes in the baseline level of the data don't affect the detection of pulses. The derivatives of the two pulses from Figure 5 are shown in part (B) of Figure 6.

The initial pulse is detected when the derivative exceeds a fixed threshold. This threshold must be set low enough so that even the smallest pulses will be detected, but high enough that the noise in the system will not cause false triggers.

5.2. Adjusted Derivative

Notice in Figure 5 that the single-pulse and double-hit derivative traces are nearly indistinguishable. In particular, the derivative doesn't go above (or even near) zero when the second pulse happens. So using the derivative doesn't directly help to identify double-hits. To do that we need to compare the actual derivative to the value we would expect from a single pulse.

Section (C) of Figure 6 is an expanded view of the time when the second pulse occurs. The data containing a secondary pulse is clearly distinguishable from the single pulse.

The CDP software maintains a copy of what the single-pulse derivative shape is. Then when a pulse occurs, the expected derivative shape is scaled so that the peak value matches the peak value of the measured derivative. This scaled derivative shape is subtracted from the calculated derivative to form the adjusted derivative.

The adjusted derivative is then used in place of the derivative for detection of secondary pulses. The adjusted derivative is compared to a threshold, and a secondary pulse is preliminarily detected when it exceeds that threshold.
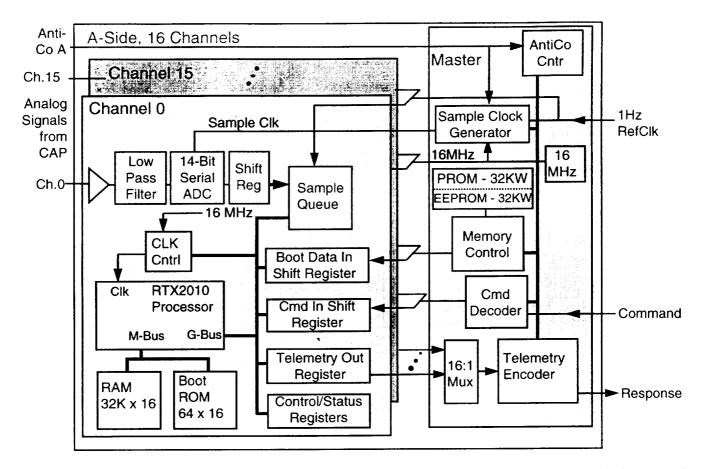
Figure 4. CDP block diagram. There are 2 completely independent "sides", labeled A and B, for redundancy. Each side comprises a "master" circuit which handles commands and telemetry and stores the DSP code in PROM and EEPROM, and 16 DSP channels. One channel is shown in the diagram.

6. ANALYSIS

6.1. Pulseheight Analysis

The job of the CDP pulse height analysis is to get the best possible estimate of the height of each pulse. To first order, the pulseheight is proportional to the X-ray energy, and the higher-order terms can be calibrated.

The zeroth-order PHA is just the peak value of the pulse minus the baseline it was sitting on. This is not a good measure of the pulseheight, though, because the noise in the result is equal to the noise per sample (assuming we can average out the noise on the baseline).
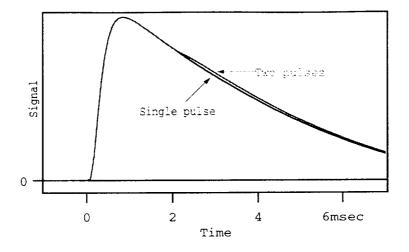
What we'd like to do is use all the samples in the pulse to calculate the pulseheight, thus reducing the noise by averaging. We do this by optimal filtering,[3] which is very similar to using a matched filter.

A matched filter is a template with the expected pulse shape, which is multiplied point-by-point with the data. An optimal (Wiener) filter is just a matched filter that has been adjusted based on the noise spectrum of the system, as described here.

We model the pulses as a form $A = H \times S(t)$, where $S(t)$ is constant. That is, we assume all pulses are scaled versions of a single shape. This assumption is incorrect, and the variation of pulse shape with energy is a large source of nonlinearity in the calculated pulseheight.

For a given pulse $D(t)$ we want to perform a least-squares minimization to find the best value of H:

$$\chi^2 = \sum \frac{(D(f) - H \times S(f))^2}{N^2(f)}$$

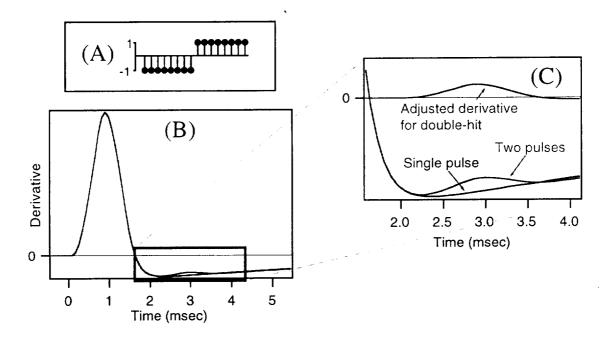**Figure 5.** A 200 eV pulse 2 ms after a 10keV initial pulse.

Figure 6. (A) The form of the boxcar derivative. (B) The derivative of a single pulse, and of the double pulse from Figure 5. (C) Magnified view of the derivatives where the second pulse occurs, including the *adjusted derivative*, which is the difference between the single pulse and double pulse derivatives.

where $N^2(f)$ is the power spectrum of the noise. Minimizing yields the optimal value for H:

$$H = \sum \frac{D(f) \times S(f)}{N^2(f)}$$

This expression is not in the best form for rapid data processing, since it is in the frequency domain. Because transforming between frequency and time domains is computationally expensive, we would rather not convert each pulse to the frequency domain to calculate its pulseheigh. So, transforming back to the time domain, we have

$$H = \sum D(t) \times f(t)$$

where $D(t)$ is the inverse Fourier Transform of $\frac{S(f)}{N^2(f)}$. Now we are left with N multiply-accumulate operations. where N is the number of points in the template (2048, in our case).

If the power spectrum of the noise were white, the optimal filter would be identical to the average pulse shape. Since our noise increases as we approach DC, our optimal filters typically have a region with negative weights before the pulse arrives. In effect, this region is measuring the height of the variable baseline. and subtracting it from the region under the pulse. Note that this means that we need information before the pulse arrives. If there are some peaks in the noise spectrum. due to pickup of extraneous signals. the S/N ratio will be small at those frequencies, and the filter will effectively "notch out" the noise, regardless of its phase.

To actually create the template, we need a measure of the system noise spectrum and the expected pulse shape. We collect this information in a three-step process. First we average a number (usually 100 to 200) noise spectra, each of which we get by Fourier Transforming a data record containing no triggers.

Next we need a source of X-rays, to determine the average pulse shape. These X-rays are provided by a radioactive source (^{4}1Ca) which is in the Front-End Assembly just above and to the side of the detectors. The X-rays from this source are all of the same energy, and so are also useful for long-term gain calibration.

Before collecting events to calculate the average pulse shape. we first make a 2-dimensional histogram of pulses in pulseheight-risetime space, using a quick measure of the pulseheight. The largest peak in this space is used to create a set of limits for collecting average pulses.

Finally, a number (typically about 100) clean pulses (with no pileup) which fall in the risetime/pulseheight limits calculated in step two are added to form the average pulse. The average pulse is then transformed, divided by the noise spectrum. and inverse transformed to create the template.

The spectra of a typical average pulse, noise, and the resulting template are shown in figure 7.

6.2. Event Grades

The above PHA method requires a template (2048 samples) worth of data with only one pulse in it. At our sample rate of 12288 Hz. this requires pulses to be more than 166 ms apart, so we need some way of handling pulses which are separated by less than a template length. In fact, the CDP uses two methods for handling such pulses.

If two pulses are closer than 2048 samples, but not "too close" (what that means will be described below), a shorter optimal template is used. This is known as a "mid-res" PHA. If they are too close even for that. the peak pulseheight ("low-res") is used. The events in the data stream are tagged with the event grade.

The main reason for making the template 2048 samples long is to more effectively reject interference at single frequencies (tones). The CDP also contains a shorter "mid-res" template. This template is typically 512 samples long, but can be shorter than the hi-res template by a factor of 4. 8, 16, 32, or 64. This template is created using the same data as the normal Hi-res template, in the following way.

First, the noise spectrum is averaged and decimated by the length factor. Then the average pulse is truncated to the appropriate length. The truncated average pulse is transformed, divided by the decimated noise spectrum. and inverse transformed back to the time domain.

In the absence of noise tones, it provides approximately the same resolution as the Hi-res template. If there are noise tones, then the Mid-res pulses will have lower resolution than Hi-res. We typically have many noise tones on the ground (mostly power-line harmonics and computer vertical sweep harmonics), but don't expect any on orbit.

For pulses which are too close for the mid-res templates, the CDP falls back to simply measuring the peak height from the baseline. This results in "low-res" events. The low-res events have a resolution about a factor of 2 worse than the hi-res events. However, when pulses are that close. the detector is still warm when the second one arrives, resulting in a lower amplitude for the same X-ray energy.

For this reason. the CDP also tags events with a "secondary" flag, which means that an event fell on the tail of a previous event. So if a pair of pulses arrives 10 ms apart. they will both be low-res, but only the second one
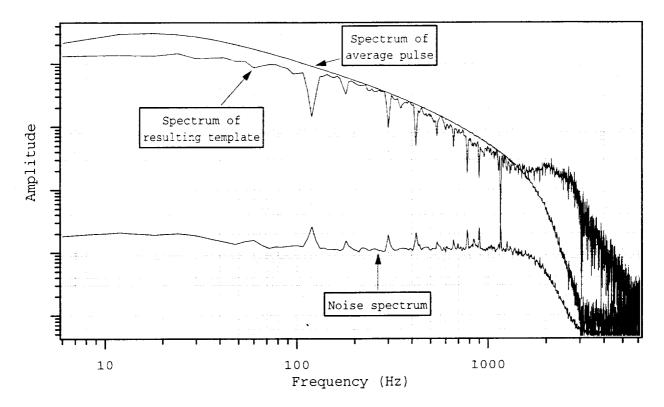
Figure 7. Spectra typical of template generation. The noise spectrum is an average of 200 power spectra of noise records (shown here as amplitude). The pulse spectrum is the amplitude spectrum of 100 pulses, which were first averaged in the time domain. Note that the template spectrum has dips at the frequencies where the noise spectrum has peaks (at multiples of the power-line frequency, since this is from ground testing). The small hump in the template spectrum at around 2 kHz is an artifact of the CDP processing.

	$(t_1) < 35.5$	$35.5 < (t_1) < 142$	$(t_1) > 142$
$(t_2) < 35.5$	LS	LS	L
$35.5 < (t_2) < 142$	LS	MS	M
$(t_2) > 142$	LS	MS	H

Table 1. Event grades as a function of time since previous pulse (t_1) and time until next pulse (t_2). H:Hi-res, M:Mid-res, L:Low-res, S:Secondary. Times are in milliseconds.

will necessarily be marked secondary. Secondary low-res events are certain to suffer from reduced gain. Secondary mid-res events also have a slight gain reduction, and we discovered during calibration that even hi-res events within about 200 ms of the previous event also have a very slight gain shift.

All events are time-tagged, which allows us to correct fairly effectively for the reduced gain of secondary events.

It is important to note that each individual event is analyzed with the best possible resolution; there are no resolution modes. In fact, once collecting science data, the CDP has no modes at all.

Table 1 indicates the grade for an event separated in time from a preceding event by (t_1) ms and from a subsequent event by (t_2) ms, all three events occurring on the same pixel. This assumes that the mid-res template is 1/4 the length of the hi-res template, which is the usual case.

The fraction of events in each grade can be calculated easily, assuming a Poisson distribution of photons, using the information in Table 1. The results are shown in Figure 8. Figure 8 also shows an estimate of the fraction of events which are missed entirely, because they are too close to the preceding pulse. If two pulses are too close to be detected individually (about 0.7 ms apart), they will be measured as a single event, with a pulseheight approximately

equal to the sum of the two individual pulseheights. Such events also have a very broad range of risetimes, since even a small offset in time leads to a noticeable S-shape in the rise of the combined pulse.
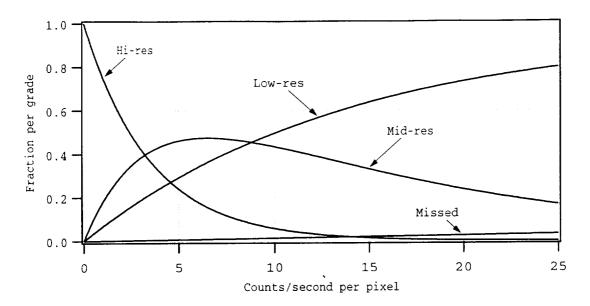

Figure 8. The fraction of events which fall into each grade, as a function of count rate per pixel. This applies to each pixel individually.

6.3. Risetime Determination

Along with pulseheight, time, and flags, each event includes a measurement of the pulse risetime, this is useful to identify certain classes of "bad" events.

As described in Reference 4, the detectors consist of a silicon body with an implanted thermistor, and a mercury telluride absorber attached via a small silicon spacer. When an X-ray is absorbed, it creates a non-thermal spectrum of phonons. However, for the thermistor to accurately measure the energy deposited, the phonon spectrum must be thermal.

Most X-rays are absorbed in the HgTe, where the phonons have a chance to thermalize before traveling through the spacer to the thermistor. However, some photons pass through the absorber and interact directly in the silicon part of the pixel. These events cause a smaller temperature rise for the same energy X-ray, and so must be excluded from spectral analysis. Fortunately, the pulse profile from such an event has a much faster risetime. We can use this fact to identify these events and exclude them from the analysis.

We determine the risetime by fitting a straight line to a fixed number of points centered at the 50% amplitude point of the rise. Normally we use 3 samples, but this can be changed by a command. We use a fixed number of points to eliminate a source of bias in calculating the risetime. If we use all the points from, say, 10% to 90%, then there are discrete jumps in risetime between pulses with N points on the rise and pulses with $N + 1$ points. This is because the rise is S-shaped, and so adding one more point to the calculation always increases the risetime.

6.4. Anticoincidence Flagging

In order to avoid contaminating the data with particle hits, we have an anticoincidence detector (see Ref. 4). The antico detector is a silicon PIN diode just under the pixel array. When it is triggered, the CDP receives a pulse. If an antico pulse occurs anytime within an adjustable window near the beginning of a detected X-ray pulse, that event is tagged with the Antico flag.

6.5. Pulse Quality

The CDP also provides a measure of pulse quality. This is calculated by scaling the average pulse to the same height as the pulse being analyzed, and then summing the mean-square differences between the two. This measure was originally intended to help distinguish between X-rays absorbed in the absorber and those absorbed in the silicon. However, it is of limited use for that: risetime is a much more useful diagnostic tool.

Pulse quality appears to be useful mainly for identifying noise spikes which trigger the detection algorithm. These happen exceedingly rarely, making the pulse quality measurement somewhat pointless. However, we have left it in as a diagnostic tool in case something unexpected happens on orbit.

7. CONCLUSION

The XRS on-board processing is responsible for detecting pulses and extracting a best estimate of their amplitude with an optimal filter. It also creates the optimal filter template based on the average noise spectrum and average pulse shape. The telemetered data for each pulse includes pulseheight, time, risetime, pulse quality, and event flags.

The digital nature of the processing allows easy adjustment to changes in the detectors. The resulting reduction in data rate does not degrade the results relative to the processing done off-line, yet allows the needed data to fit in the available telemetry bandwidth. This optimal pulse processing, coupled with XRS's new, high-resolution and high throughput X-ray detectors, will enable XRS to open a new window on X-ray astronomy.

REFERENCES

1. R. L. Kelley, M. D. Audley, K. R. Boyce, R. Fujimoto, K. C. Gendreau, S. S. Holt, Y. Ishisaki, D. McCammon, T. Mihara, K. Mitsuda, J. S. H. Moseley, D. B. Mott, F. S. Porter, C. K. Stahle, and A. E. Szymkowiak, "Astro-e high-resolution x-ray spectrometer," *Proc. SPIE* **3765**, p. this volume, 1999.

2. Y. Ogwara, "The astro-e mission," *IAU Symposium* **188**, pp. 75–78, 1998.

3. A. E. Szymkowiak, R. L. Kelley, S. H. Moseley, and C. K. Stahle, "Signal processing for microcalorimeters," *J. Low Temp. Phys.* **93**, pp. 281–285, 1993.

4. C. K. Stahle, M. D. Audley, K. R. Boyce, R. P. Brekosky, S. R. Breon, R. Fujimoto, K. C. Gendreau, J. D. Gygax, Y. Ishisaki, R. L. Kelley, D. McCammon, R. A. McClanahan, T. Mihara, K. Mitsuda, J. S. H. Moseley, D. B. Mott, F. S. Porter, C. M. Stahle, and A. E. Szymkowiak, "Design and performance of the astro-e/xrs microcalorimeter array and anti-coincidence detector," *Proc. SPIE* **3765**, p. this volume, 1999.