Verification of the Astro-E Hard X-ray Detector based on newly developed Ground Support Equipment

J. Kataoka[a,b], M. Nomachi[c], T. Takahashi[a], G. Kawaguchi[b],Y. Terada[b], T. Murakami[a], C. Tanihata[a,b], Y. Uchiyama[a,b], K. Yamaoka[a,b], Y. Fukazawa[b], T. Kamae[b], K. Makishima[b], M. Tashiro[b] and H. Kubo[d]

[a]Institute of Space and Astronautical Science, 3-1-1 Yoshinodai,

Sagamihara, Kanagawa, JAPAN 229-8510

[b]Department of Physics, University of Tokyo 7-3-1 Hongo, Bunkyo-ku,

Tokyo, JAPAN 113-0033

[c]Research Center for Nuclear Physics, Osaka University, 10-1

Mihogaoka, Ibaraki, Osaka, JAPAN 567-0047

[d]The Institute of Physical and chemical Research, 2-1 Hirosawa, Wako,

Saitama JAPAN 351-0198

ABSTRACT

We report the first results of the ground test of the Hard X-ray Detector (HXD) on board the Astro-E mission, by means of the newly developed Ground Support Equipment (GSE). Astro-E will be launched in 2000 by a Japanese M-V rocket. In order to verify the detector system during the limited time before launch, fast and versatile GSE is necessary. For this, we have developed a flexible test system based on nine VME I/O boards for a SUN workstation. These boards carry reconfigurable Field Programmable Gate Arrays (FPGAs) with 50,000 gates, together with 1 Mbyte SRAM devices tightly coupled to each FPGA device. As an application of using this GSE, we have tested the performance of a phoswich unit of the Flight Model of the HXD. In this paper, we present a schematic view of the GSE highlighting the functional design, and the results of our ground test of the HXD-sensor under the high count rate environment ($\sim$ 10 kHz/unit) expected in orbit.

Keywords: reconfigurable FPGA, Ground Support Equipment, Hard X-ray Detector, Astro-E mission

1. INTRODUCTION

The Field Programmable Gate Array (FPGA) was first introduced in the 1980s as a single-purpose technological unit, a modest device with no function beyond processing digital logic. FPGAs resemble traditional mask-programmed gate arrays, but differ in that they are programmed by the end user. Owing to the recent advances and technical innovation in the field of electronics, this adaptable technology has virtually revolutionized the fields of computation and digital logic, being the backbone for large scale systems. However, their flexibility sometimes sacrifices speed and density to mask-programmed devices. Furthermore, it is a great problem that once an architecture is dedicated to silicon, changes and corrections are not made easily. Thus, a system using such FPGAs was believed to be economically less advantageous, especially for the proto-typing stage which usually requires redesign and reprogramming for changes in the circuit.

The advent of a new class of logic devices – the static memory (SRAM) based FPGA – has greatly expanded viability in the field of FPGA.[6,7] Since the devices use static memory cells as the programming technology, a configuration program can be loaded into each FPGA when the system is powered up, and it disappears as soon as the device is powered down. With these devices, changes can be made to a system's logic functions simply by reconfiguring the FPGAs resident in the system. A user can design exactly the special hardware for a given task

J.Kataoka : E-mail: kataoka@astro.isas.ac.jp

without having to construct new hardware for each application. This increases the cost-effectiveness of using FPGAs, and brings maximum reliablity to design processes. Careful checks and measurements using test-circuits can be made easily at the proto-typing stage, instead of running complicated simulations with numerous input parameters for the circuit. Today, SRAM based FPGA devices with more than 200,000 gates have become available.[1]

In addition to the technical progress for the device itself, there are significant advances in the enviroment for designing FPGAs. One of the more remarkable advances is the development of Hardware Description Language (HDL). HDLs and synthesis tools can greatly decrease design time, improving time-to-market. A description based on HDLs is easier to understand than some schematic for a very large design in FPGA gate format. An engineer can later pick up the design and learn it more quickly and painlessly. The concept of HDL is to construct a sequence of actions to be performed, defining a logic entity or a specific task, by means of *state diagrams* or *state machines*. Today, there are several kinds of HDLs developed by different corporations: AHDL,[2] VHDL[18] and Verilog-HDL.[17]

The Hard X-ray Detector (HXD[8,13]) on board Astro-E, the fifth Japanese X-ray satellite to be launched in 2000, consists of a $4\times4=16$ modular assembly of identical well-type phoswich counters[5,8] and silicon PIN diodes.[9,10,12] The data from the sensor part (HXD-S) are processed by the analog electronics (HXD-AE) and the digital electronics (HXD-DE), specifically designed for the modular configuration of the HXD.[14] Because of the complexity of the detector system, careful verification is necessary before launch. However, the HXD-DE is only available at the final stage of system integration. To simulate the various functionalities of the HXD-DE and to enhance its capability for faster data acquisition, we have developed the Ground Support Equipment (GSE) based on a modular assembly of nine VME I/O boards (Figure 1). The prime motivation of the development of the GSE is to test the HXD-AE thoroughly and to perform calibration of the HXD-sensors. The kernel of this system is a reconfigurable SRAM based FPGA, carried on each board. We have designed all the circuits with the Altera Hardware Description Language (AHDL). We give an overview of the detector system of the HXD in §2, and the configuration of the GSE in §3, presenting the schematic view of functionalities we have designed. In §4, the performance tests using the newly designed VME I/O board are summarized. In §5, we show the first results of the ground test of the HXD by means of the GSE developed in this paper, and we present our conclusions in §6.

2. DETECTOR SYSTEM OF THE HXD

2.1. Data Flow in the HXD

The HXD-AE receives the photon event data from the HXD-S (Figure 1). Each sensor consists of a deep well of BGO inorganic scintillators with GSO and PIN diodes embedded therein.[8–10,12] Six channels of signals per unit (2 from the PMT and 4 from PIN diodes) are fed into succeeding analog electronics on the HXD-AE. Together with the BGO shield counters placed to surround these phoswich counters, the HXD has 116 signal channels in total. The maximum data acquisition rate from all sensors amounts to 4 kHz (64 kByte/sec). The HXD-DE formats and processes the digitized data. It reacts to requests for services, and provides the primary interface with the satellite data processor for commands and telemetry.[14] The final data are then transmitted to the ground stations under the control of the HXD-DE. Data from various environmental monitors are also critical in the HXD system, because of the temperature dependence of the scintillation light yield[15] and difficulties for background subtraction in orbit. We acquire these monitor data periodically through the HXD-AE and send to the ground stations.

The HXD-AE consists of one Analog Control Unit (ACU) and eight signal-processing boards. The latter is organized from four Well Processing Units (WPUs) and four Transient Processing Units (TPUs).[14] The ACU handles the power supply to the sensors and WPU/TPU boards, and monitors the housekeeping data about power supplies, timing and temperatures on sensors. The signals from four neighboring phoswich counters are processed in one WPU board, while the signals from five BGO shield counters are processed in one TPU board.

The HXD-DE controls data acquisiton and arranges data into a fixed format. To cope with the high event rate, each AE board is capable of sending data by the Direct Memory Access (DMA) mode[14](see, also §3.2). After the accumulation of 64 events in the memory, an interrupt signal is sent to the CPU as a request for data processing.

2.2. Data Types

The data from the HXD-AE are classified into two groups: *observational data* and *monitor data*. The *observational data* from each WPU board includes pulse height information, existence of any hits on the other sensor units, and the arrival time of each photon event. The data length is 16 byte. The monitor data from the WPU consist of scaler

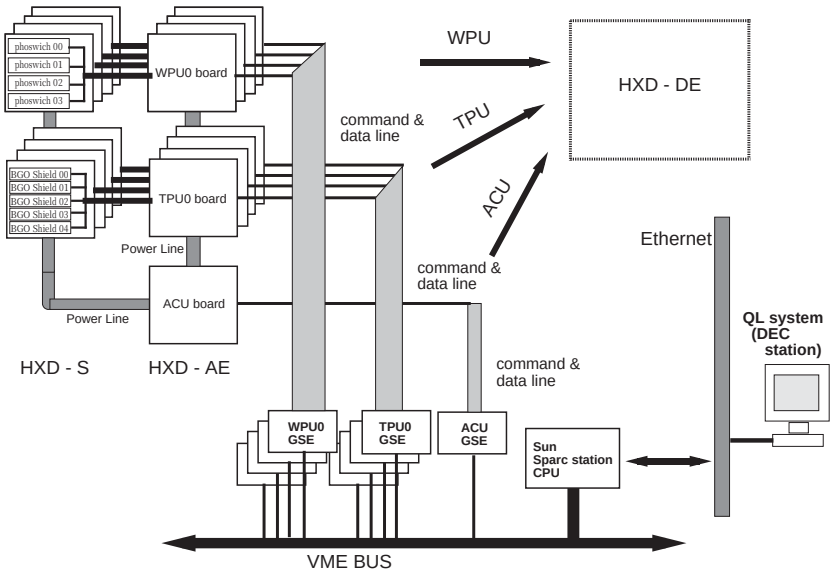

Figure 1. The block diagram of the ground test system of the HXD. The GSE presented in this paper can simulate all the functionality of the HXD-DE.

counts for the PMT anode triggers and PIN diodes. It also includes the output of scalers that monitor the deadtime of the analog circuits.[14] For the TPU board, the summed-up signals of each BGO shield counter are used to produce two sets of pulse height distributions. One has only four energy bins but is renewed every 16 msec, and the other has 64 energy bins and is renewed at a 512 msec interval. The former will be included in the *observational data* called γ-ray burst data, to detect various flares and other bright transient phenomena. The pulse height information and light curves accumulated in 128 seconds will be packed into 1 kbyte length blocks. The latter, called Transient data, is part of *monitor data* and enables continuous monitoring of transient phenomena with finer energy bins.

3. DEVELOPMENT OF THE GROUND SUPPORT EQUIPMENT

The GSE is designed to handle data from the HXD-AE (Figure 1). It consists of nine VME I/O boards associated with nine AE boards (Figure 1). Four WPU-GSEs control data acquisition from four WPU boards, and react to requests for services by sending commands to each AE board. The rest of the boards, four TPU-GSEs and one ACU-GSE, are also shared to control associated AE boards. This modular, flexible system is implemented on the VME bus. A SUN sparcstation (SPARC CPU-5CE/32) reads data stored in each VME I/O board. We also use another workstation (DEC α PC 164, 433 MHz), connected via a network, for further analysis and recording the data.

3.1. Design of the VME I/O Board

To cope with the high data acquisition rate of the HXD-AE, we designed a new VME I/O board (6U double-height) that carries a reconfigurable FPGA (FLEX10K50) and 1 Mbyte SRAM device (HM628128), tightly coupled to each FPGA device (Figure 2). Figure 3 shows a photograph of the VME I/O board developed in this work. For the interface between the VME bus and the FLEX device, we used a Programmable Logic Device (PLD) 22V10.

The FLEX10K family[1,11] is Altera's latest family of CMOS devices and contains many features suitable for programmable hardware systems. FLEX10K50 is an SRAM based FPGA with 50,000 gates and more than 310 I/O pins are available to the user.

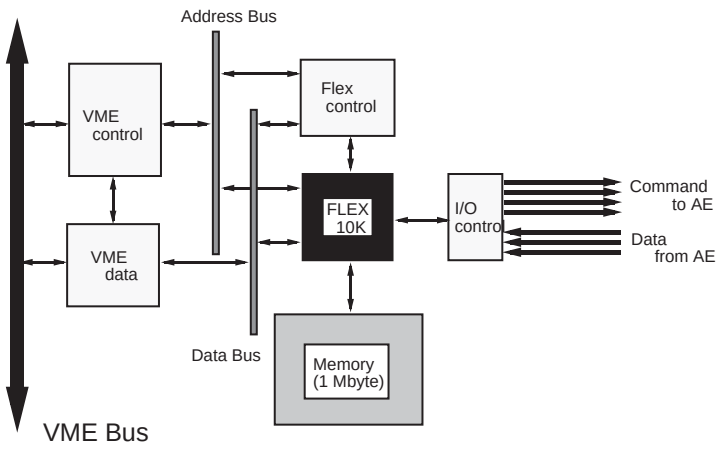

Figure 2. The design flow of a VME I/O board.

Figure 3. The picture of VME I/O board.

Each FLEX10K50 device contains 2880 Logic Elements (LE), 360 Logic Array Blocks (LABs) and 10 Embedded Array Blocks (EAB).[1] The configuration data of the newly designed circuit has tabular ASCII format and is loaded from a SUN sparcstation via a VME device driver. One can also use the serial JTAG[4] port on the FLEX device for the configuration. Throughout the development, we used the MAX+PLUS II development system, also provided by the Altera corporation. We designed all the circuits by Text Design Files[2] (TDF) written in AHDL.

3.2. Interface between the HXD-AE and VME I/O (GSE) Board

In order to achieve a high data acquisition rate, each AE board and the HXD-DE are connected via bi-directional serial lines as shown in Figure 4. The HXD-AE sends data to the HXD-DE by DMA transfer, while the commands for the operation of the HXD-AE and the HXD-S are sent from the HXD-DE by serial lines of inverse direction. We designed the interface between each AE board and the GSE board in the same way as for the HXD-AE and DE: four command lines and three data lines.

For this, we used seven I/O pins on the FLEX device for command output and data input. When one requests to send a command from a SUN workstation, this command data is first sent to the register on the associated FLEX device through the VME bus. The contents of the register will be checked immediately, and arranged into a serial format. It is finally transmitted to the AE board via the output port on the board. The data from AE are injected into the FLEX device via the input port and arranged into a parallel format to be sent to the memory device on the board. One can read data stored in the memory via the VME bus, by pointing to the address where the memory is mapped. The data are finally transfered to a DEC station connected via a network, for further analysis and recording.

3.3. Configuration of the Circuit in FPGA

3.3.1. Outline

To realize the functionality presented above, we divided the configuration of the FLEX device into five blocks as shown in Figure 5. The *VME Address Controller* is the interface to the VME address bus. It also supplies control lines to the memory device. The *VME Data Controller* is connected to the VME data bus and works as a bus tranceiver. A SUN workstation sends the requests to the FLEX device for the operation of the HXD-AE and the HXD-S. *AE Command Controller* reacts to the requests by generating two signals, *Data* and *Enable* (Figure 4). As for the ACU commands, an additional signal (*Act*) is generated for verification of a very important operation about the power supply (hardware command). For example, the power-on of each AE board and the high voltage control for the PMTs are requested with the *Act* signal. The AE data are received by the *AE Data Controller* and sent to the *Memory Controller* to be stored in the memory device. In order to set the address in memory, the data size is also verified by the *AE Data Controller*.

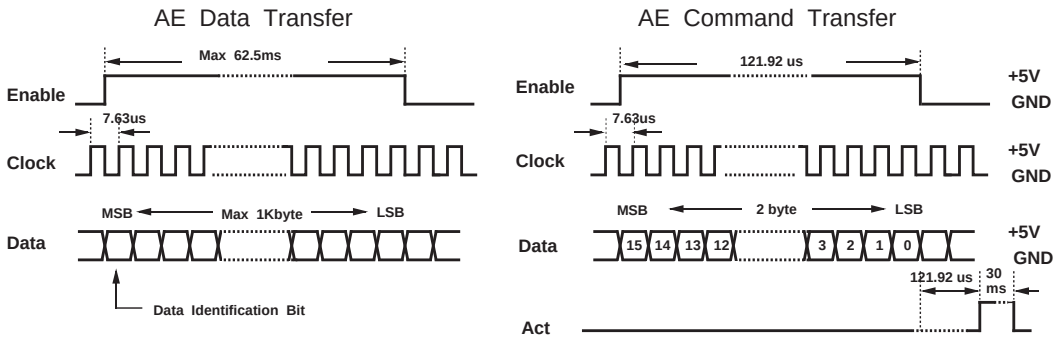

Figure 4. The serial interface between the HXD-AE and the HXD-DE.

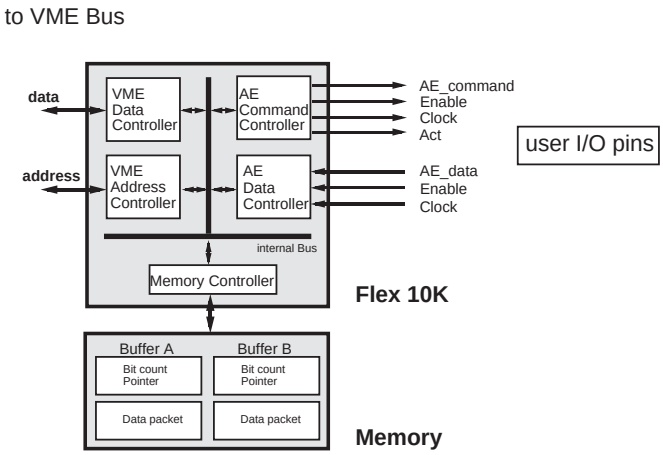

Figure 5. The block diagram of the FLEX10K device and the memory device.

3.3.2. *AE Command Controller*

Figure 6 illustrates the sequence of tasks for the command output by means of a simple state diagram. In this notation, the present state depends only on its previous input and previous state, and the present output depends only on the present state. The clock runs at 20 MHz in the FLEX device, and the transition to the next state will be synchronized with the leading edge of the clock pulse.

The initial state is *Idle*. It is waiting for requests for sending commands to the HXD-AE. The strobe signal on the VME bus, which is activated during the request, is examined on every clock pulse. Once a command is requested, it will be detected within 50 nsec and generate a trigger (*go*) for starting the sequence. The next state *Synch* is a state for obtaining synchronization with the *Clock* (Figure 4). It sets the clock counter (*bit_cnt*) to 15, corresponding to the command length of 2 byte. During the iteration of *Clk High* and *Clk Low*, the *Enable* signal is activated and the command data is read out until the clock counter decreases to zero (*Done*). The command profile is checked at *Judge*. If it turns out to be a hardware command to the ACU board, the *Act* signal will be generated in the succeeding states *Att wait* and *Att High*, although they are skipped when WPU/TPU commands. This cycle ends at *Att done* and the state goes back to *Idle*, waiting for the next request.

3.3.3. *AE Data Controller*

Figure 7 illustrates the sequence of tasks for receiving data from each AE board. This sequencer is separated into two parts according to functionality. The first part handles the raw AE data. The initial state is *Idle* and it is waiting for an input of AE data to start the action. It senses the low-to-high transition on the *Enable* signal as a trigger for changing the state. The most significant bit (MSB) of AE data is the 'Data Identification Bit'. It is low

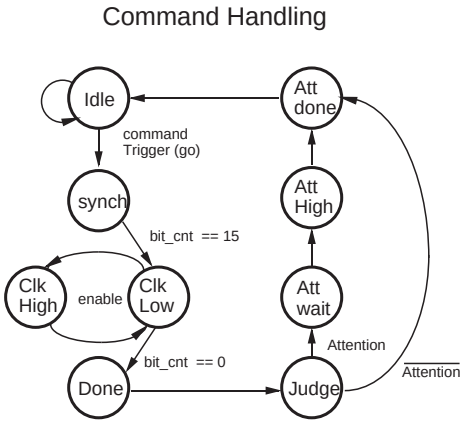

Figure 6. State diagram for the *AE Command Controller*.

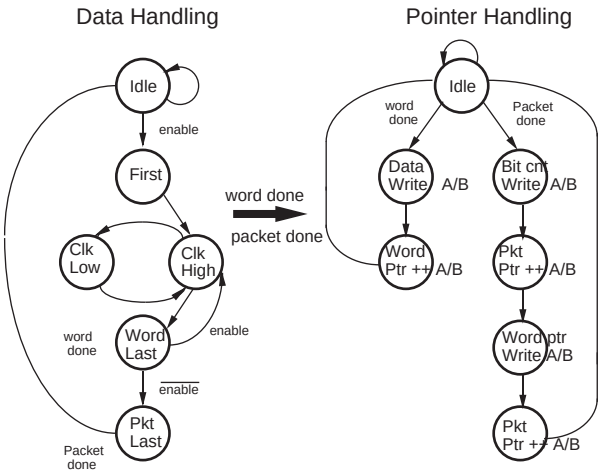

Figure 7. State diagram for the *AE Data Controller*.

for *observational data* and high for *monitor data*, and recognized at *First* in the sequence. After this identification, the iteration of *Clk High* and *Clk Low* continues until the *Enable* line is released to the low level. The sequence passes through the state *Word Last* every one word (4 byte) and *Pkt Last* at the end of receiving the data packet. The signals *word done* and *packet done* will be generated at *Word Last* and *Pkt Last* respectively, and both of them are injected into the second part of the sequencer.

Another part handles the pointer of AE data, by checking the data size for an increment of the pointer. It also allows permission to memory access, after receiving a *word done* signal (*Data Write*). The input data are thus transfered to the *Memory Controller* in one-word packets and stored in the memory. After receiving the total packet (*packet done*), data size and packet pointer are also recorded in the memory.

3.3.4. Memory Structure and *Memory Controller*

A double buffer configuration can be one of the more effective and easiest to design fast data acquisition systems. A 1 Mbyte memory device on each GSE board is divided into two identical buffers A and B by designating the pointer (Figure 8). One can read the data accumulated in one side of the buffer, without intervening the write cycle proceeding on the other side. By changing the readout buffer cyclically, a ring buffer of infinite memory size is virtually constructed.

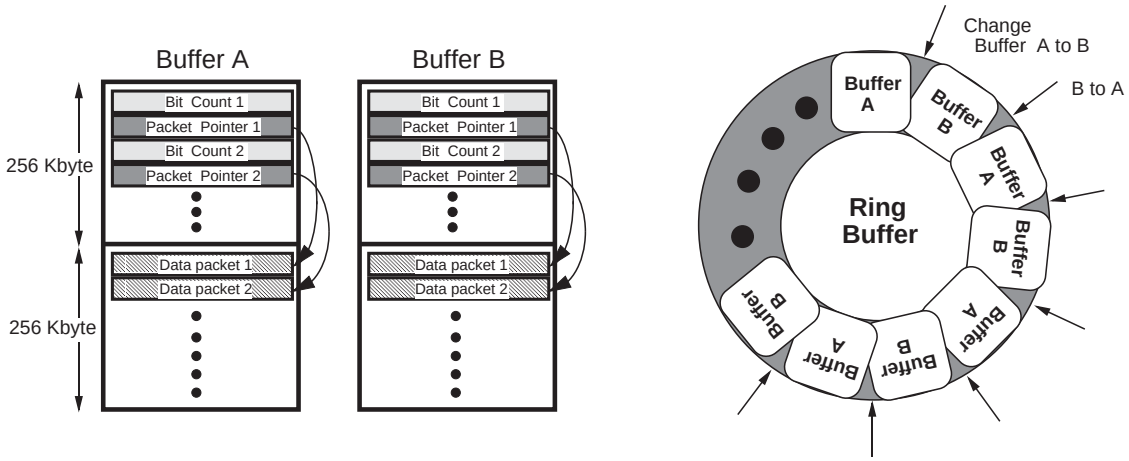

Figure 8. The memory configuration.

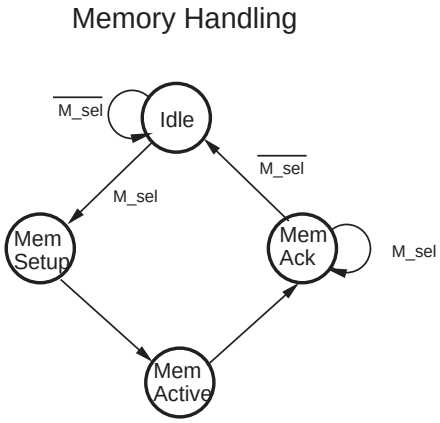

Figure 9. State diagram for the *Memory Controller*.

Each 512 kbyte buffer consists of a pair of 256 kbyte blocks: one is used to store the data size and packet pointer for each event, and the other is shared as a cache of raw AE data (Figure 8). In our design, a pointer to the nth data packet is recorded at the $(2n+1)$th word from the top of the memory, and the data size will be stored at the $2n$th word. By refering to the pointer and a size of the data for the nth packet, one can easily access the raw data stored in another cache of the memory block. In the erroneous case that the data are not read out before a memory overflow, the increment of the pointer will be stopped automatically and an overwrite of the data can be avoided.

Figure 9 illustrates the state diagram for handling the memory (*Memory Controller*). The initial state of the sequence is *Idle* and waiting for an access to the memory. As soon as it receives a request, the trigger signal (*M_sel*) is activated and the sequence moves to the setup mode (*Mem Setup*). The memory becomes active (*Mem Active*) during the access from the *AE Data Controller*. After the data transfer has been completed, *M_sel* is released to the low level to be verified (*Mem Ack*). Then the state finally goes back to *Idle*.

4. PERFORMANCE TEST

The performance of the *AE Command Controller* can be tested by direct measurement of the output signals. Also the functionalities of the *AE Data Controller* and the *Memory Controller* are tested by making a simple system consisting of two VME I/O boards (A, B). We connected the output port on board-A to the input port on board-B. The signals are output from the FLEX device on board-A and sent to board-B as input to the *AE Data*

Controller. The data stored in the momory device on board-B will be checked from a SUN sparcstation. Owing to the reconfigurability of the FPGA device, errors can be corrected and different algorithmic approaches explored, with no further hardware expense. No complicated simulations for performance tests are necessary. For our design, we used 38 % of gate logic in the FLEX10K device and the propagation delay was estimated to be 60 nsec at maximum.

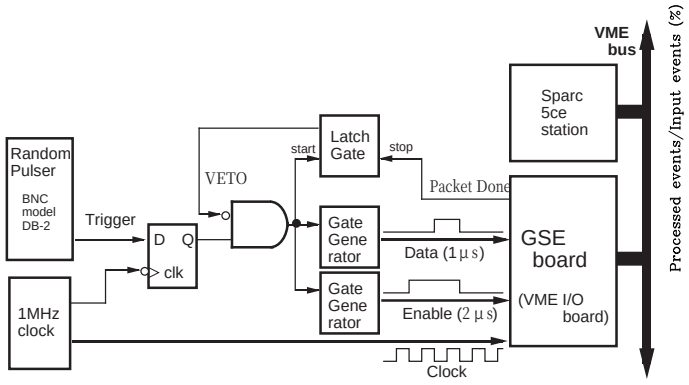

Figure 10. Set-up for the performance test of the VME I/O board.

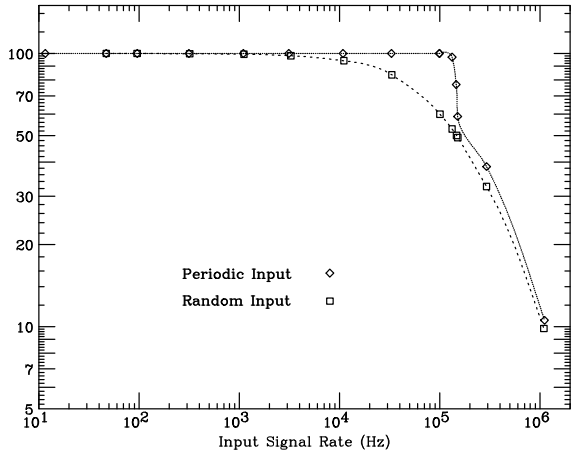

Figure 11. The result of the performance test using the newly developed VME I/O board.

Since the data acquisition rate is limited to 1 kHz for each AE board, this can place an apparent limit on the performance of the GSE. However, our configuration of the FPGA device coupled to the memory on each board, allows data acquisition faster than 1 kHz to be realized. We tested the potential of the VME I/O board using the set-up shown in Figure 10. Three serial ports were used for data input. We increased the serial *Clock* frequency to 1 MHz and shortened the *Enable* signal to 2 bit (2 μsec). Triggers for the input *Data* and *Enable* signals were generated by the Random Pulse Generator (BNC model DB-2), which can generate periodic or random pulses arbitrarily from 10 Hz to 1 MHz. The latch gate (Lecroy 222) was used to inhibit the triggers during the data-processing time. A stop signal for the latch gate is generated in the FLEX device, at the trailing edge of the *packet done* signal. After the VETO signal is released, the system is cleared to accept the next trigger.

As shown in Figure 11, both random and periodic events are collected without any artificial losses or misreading of the data below 140 kHz. This limit is consistent with a VETO signal of $\sim$7 μsec width for an event: 2 μsec for the *Enable* signal, 1 μsec before the memory access and 4 μsec for writing the data in the memory device. This result implies our approach can be advantageous for more applied purposes, such as a particle monitor in high-energy physics research and multi-channel read-out systems for silicon strip detectors. Very fast data acquisition and more than 300 I/O pins on each FLEX device gives many possibilities to the end user.

5. VERIFICATION TEST OF THE HXD FLIGHT MODEL

Based on the Ground Support Equipment (GSE) described in the previous sections, the verification test of the Flight Model of the HXD is now in progress. As a first result of these ground-based experiments, we present the performance of a single phoswich counter under the high count rate environment expected in orbit. The physical origin of the background is complex, some from diffuse cosmic γ-rays and others may have atmospheric origin. Activated nuclei due to charged particles penetrating through the large inorganic scintillators also produce background. One can estimate that this background, mostly due to the large volume of BGO scintillators, could amount to a maximum of 10 kHz for a single phoswich counter. However, the photons from celestial sources are expected to be less than 10 Hz, even for the brightest source like the Crab Nebula. Therefore, the intrusion of the piled-up events due to successive events within short time intervals causes a significant effect on the observed energy spectrum. In the following sections, we present a technique for eliminating piled-up events and selecting photon events that cleanly hit on the GSO detection part (hereafter 'pure GSO' event).

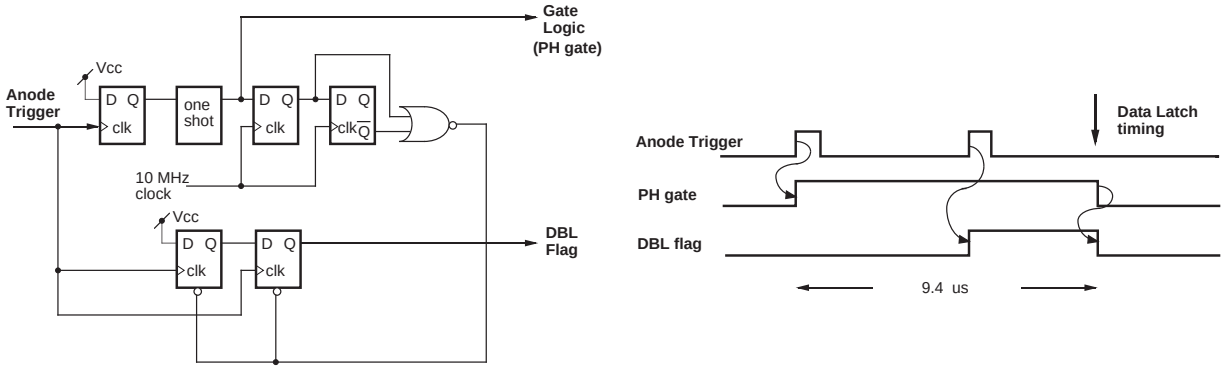

Figure 12. Circuit for detecting piled-up events and a timing chart for event-status latch.

5.1. Signal Processing of the Well-Type Phoswich Counter

The analog signal taken from the last dynode of each PMT is first fed to a charge sensitive amplifier, and passed through a CR/RC filter to a Pulse Shape Discriminator (PSD[5,8,16]) . Among several methods of pulse-shape discrimination, we have selected the double-integrating method for the HXD.[3,5] The PSD selects the clean-hit events on the GSO by comparing the output pulse heights from the two shaping amplifiers with different integration times.[5,16] The integration time is $\tau_F = 150$ ns for fast shaping, and $\tau_S = 1$ μs for slow shaping. The output of the last amplifier is fed to the peak-hold circuit and sample-held at 6 μs from the fast trigger. The peak-hold circuit is released at 9.4 μs, sufficiently long after the conversion of analog data. After the reset sequence of a few μs, digitized data will be written into the FIFO carried on each WPU board. The anode signal of the PMT is also used for a fast pre-trigger that generates the peak-hold gate for the PSD.

5.2. Technique for Eliminating Piled-Up Events

The GSO signals are digitized and written into the FIFO within 15 μs from the fast anode trigger. Assuming the worst case when the background rate amounts to 10 kHz for a single phoswich counter, more than 10 % of the total signal could be piled up. In such an environment, pure GSO events can be smeared from the piled-up events and thus may cause the deterioration of the spectral resolution.

We have devised an effective method that can eliminate piled-up events (Figure 12). We use two D-Flip Flops for detecting a trigger generated by the second pulse. The trigger status (DBL flag) is latched at 9.4 μs, at the trailing edge of peak-hold gate. The DBL flag is then included in the *observational data* and one can recognize the trigger status for each event at the ground station. The detailed scheme for the data processing in the WPU is described in a separate paper in this volume.[14]

5.3. Simulation of the Piled-Up Events

To verify the performance of the circuit for eliminating piled-up events, we first simulate the situation where the detector is irradiated by γ-rays from ^{137}Cs (662 keV) at very high rate. For isolated events that are not accompanied by succeeding triggers within the peak-hold duration (hereafter 'single-trigger events'), pure GSO signals are expected to be separated clearly from BGO signals and Compton-scattered events because of the difference in scintillation time scale.[15] Thus in a 2-dimensional plot of fast-shaper output versus slow-shaper output, three significant regions can be appeared (Figure 14): 662 keV peak for GSO, Bottom BGO (bottom part of the phoswich counter) and Well BGO (collimator part of the phoswich counter).[8] Since the light yields from the Well BGO are smaller than that of the Bottom BGO by a factor of 2, they make separated peaks on the BGO line. Compton-scattered events form bridges between the two regions corresponding to 662 keV peak for the GSO and 662 keV peak for the BGO (Bottom/Well).

We simulated the pile-up effects on the 2-dimensional plot. Here, we define δt as the time interval between the first and second signal inputs. We scan δt from 0 μs to 7 μs, after which the signals are less affected by the pile-up effect (almost equal to a single-trigger event). For example, the simulation for BGO piled-up pulses is shown in

Figure 13 for the case of δt = 2 μs. As one can see, the pulse is released to the baseline rapidly for the fast shaping, and thus makes small change on the peak-held pulse in this case. However, for the slow shaping, the integration time is so large (1 μs) that almost twice as much output of the signals can appear. This separates piled-up events considerably from the usual BGO line. The trajectory of double-trigger events for various δt forms a clear loop in the left region of the single-trigger BGO line (Figure 14). Interestingly, in the mixed case of a GSO signal injected after the BGO signal, hysteresis appears in the histogram (h, f in Figure 14).

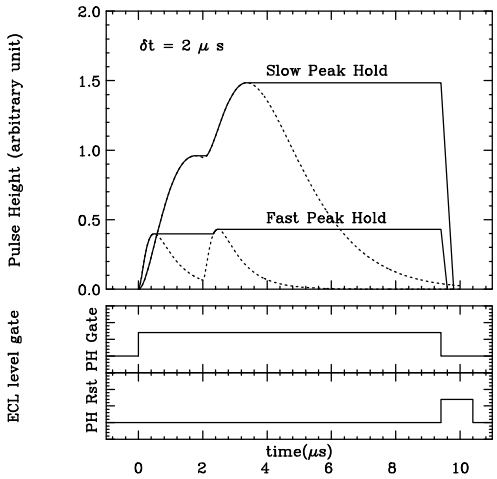

Figure 13. Simulation of the BGO piled-up pulses for separation time δt = 2 μs.

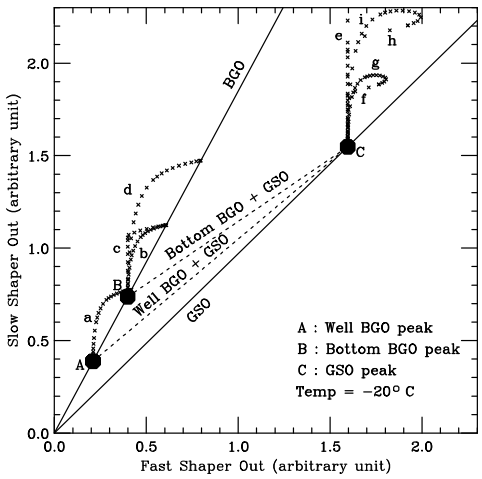

Figure 14. Simulation of 2-dimensional plot of fast shaper output (150ns) vs slow shaper output (1μs). a: Well+Well BGO, b: Well+Bottom BGO, c: Bottom+Well BGO, d: Bottom+Bottom BGO, e: GSO+GSO, f: GSO+Well BGO, g: Well BGO+GSO, h: GSO+Bottom BGO, i: Bottom BGO+GSO.

5.4. Performance Test of the HXD Flight Model

The overview of the ground test system for the HXD Flight Model was presented in Figure 1. It consists of 16 units of well-type phoswich counters, 20 units of BGO shield counters, nine AE boards associated with nine GSE boards

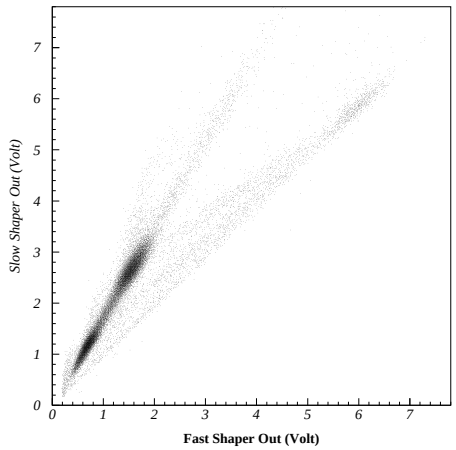

Figure 15. Response of a single phoswich counter to exposure of 662keV γ-rays. Input signal rate is about 6 kHz. The horizontal and verical axes show the peak-held pulse heights of the fast and slow shapers, respectively.

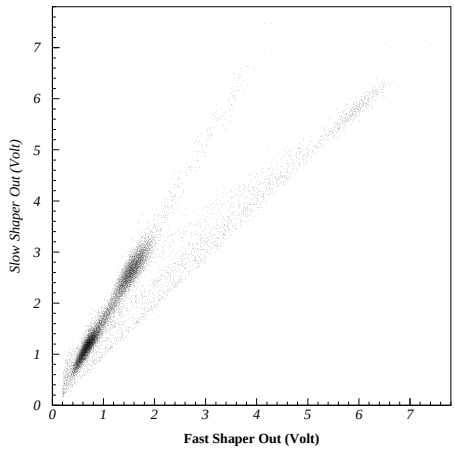

Figure 16. The same figure as Figure 15, but eliminates the piled-up events by the technique described in §5.2.

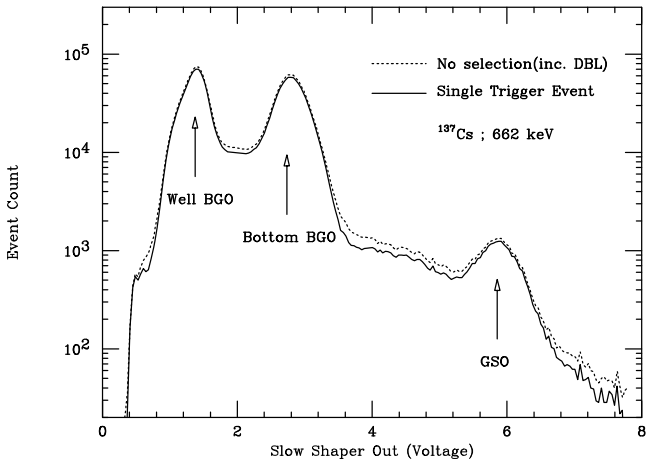

Figure 17. Energy spectrum obtained by projecting Figure 15 and 16 to the vertical axis.

and workstations for user operation. Here we concentrate on the ground test using a single phoswich conter and one WPU/ACU board. In particular, we focus on the perfomance of the circuit to eliminate piled-up events. The verification test of the total detector system, including calibration of each sensor, will be discussed elsewhere.

To increase the clean hits on GSO, we irradiated with γ-rays from ^{137}Cs, from the side of the phoswich counter. Thus the relative ratio of GSO hit events to BGO events are slightly higher than in the actual situation. The counting rate was about 6 kHz in total. The temperature of the sensor was kept at $-20°$C, which is expected in orbit. As one can see in Figure 15 and 16, our approach eliminates piled-up events effectively. Also the results of the simulation reproduce the actual 2-dimensional plots exactly. In Figure 17, we compare the spectrum of single-trigger events to a mixture of piled-up events. The energy spectrum derived from the single-trigger events seems to have a clearer peak than that containing piled-up events.

6. CONCLUSION

We have developed a fast and versatile GSE based on nine VME I/O boards for the ground test of the Astro-E Hard X-ray Detector. This board carries a reconfigurable FPGA with 50,000 gates, together with a 1 Mbyte SRAM device tightly coupled to each FPGA device. Owing to the reconfigurablility of the FPGA device and the modular structure, development and extension of the system was extremely simplified. In our performance test, 140 kHz input signals are successfully acquired with no data losses or misreading. Such FPGA-based reconfigurable systems can be a viable platform for future applications. As a first result of ground test of the HXD, we have presented the performance of a single phoswich conunter under the high count rate enviroment expected in orbit. We have shown that our system can eliminate the piled-up signals effectively, even if the background exceeds 6 kHz in total. Verification of the Flight Model of the HXD is now in progress, directing all efforts to the launch in 2000.

Acknowledgments

We would like to thank Mr. K. Mori of Clear Pulse K.K. and Messrs. K. Taguchi, M. Horii, M. Hamaya, S. Shinoda, and R. Shoumura of Meisei Electronics for design and production of analog electronics. We also thank the members of the Astro-E HXD team, Mr. M. Tanaka, Dr. H. Ezawa, Mr. K. Matsuzaki, N. Iyomoto, E. Idesawa, T. Mizuno, A. Kubota, M. Kokubun, K. Nakazawa, N. Isobe, Y. Matsumoto, H. Ozawa and S. Kubo. This research was partially supported by Fellowships of the Japan Society for Promotion of Science for Japanese Young Scientists, Grant-in-Aid for Scientific Research No. 05242101, and Grant-in-Aid for COE Research No. 07CE2002 by the Ministry of Education, Culture and Science, Japan.

REFERENCES

1. Altera Corporation, "Altera 1998 Data Book", 1998
2. Altera Corporation, "MAX+PLUS II Programmable Logic Development System – AHDL –", 1998
3. F. D. Brooks "A Scintillation Counter with Neutron And Gamma-Ray Discriminators", Nucl. Instr. Meth. 4, 151, 1959
4. P. J. Covert "Design tips and experiences in using re-configurable FLEX logic", SPIE 2914, 124-132, 1997
5. H. Ezawa et al. "Development of the Pulse-Shape Discrimination LSI of *Astro-E* Hard X-ray Detector", IEEE Trans. Nucl. Sci. NS-43, 1521, 1996
6. B. K. Fawcett, "Field Programmable Gate Arrays and Reconfigurable Computing", SPIE 2607, 155-166, 1995
7. S. Hauck, "The Roles of FPGA's in Reprogrammable Systems", Proceedings of the IEEE 86, No.4, April , 1998
8. T. Kamae et al., "Astro-E hard X-ray detector", SPIE 2806, 314-328, 1996
9. H. Kaneda et al., "Development of the hard X-ray detector for the ASTRO-E mission", SPIE 2518, 85-95, 1995
10. H. Ozawa et al., "Integration of the readout electronics for the ASTRO-E Hard X-ray Detector", SPIE 3115, 235-243, 1997
11. S. J. Smith, "Programmable Hardware for Reconfigurable Computing Systems", SPIE 2914, 133-140, 1997
12. M.Sugizaki et al., "Development of the large area silicon PIN diode with 2 mm-thick depletion layer for Hard X-ray Detector (HXD) on-board ASTRO-E", SPIE 3115, 244-254, 1997
13. T. Takahashi et al., "Development of the Hard X-ray Detector for the ASTRO-E Mission", Astron. Astrophys. Suppl. 120, 645-648, 1996
14. T.Takahashi et al, "Electronic System for the Astro-E Hard X-ray Detector", 1998, in this volume.
15. N. Tsuchida et al., "Temperature dependence of Gamma-ray excited scintillation time profile and light yield of GSO, YSO, YAP and BGO", NIM 385, 290-298, 1997
16. K. Tsukada et al. "Peak Hold Monolithic Integrated Circuit with Built-in Shaping Amplifier for Hard X-ray Detector", IEEE Trans. Nucl. Sci NS-40, 724, 1993
17. D. E. Thomas and P. Moorby, "The Verilog Hardware Description Language", Kluwer Academic Publishers, 1991
18. Institute of Electrical and Electronic Engineers, Inc., "VHDL Language Reference Manual", IEEE Standard 1076-1987, 1988