Fast Auroral Snapshot Explorer (FAST) Packet Processing System

Jeff Shi, Tony Mao
RMS Technologies, Inc.
Code 520.9

James Chesney, Nicholas Speciale
Data Systems Technology Division
Code 521

Mission Operations and Data Systems Directorate
NASA, Goddard Space Flight Center
Greenbelt, MD 20771

ABSTRACT

This paper describes the design of a space telemetry level zero processing system for National Aeronautics and Space Administration's (NASA's) Fast Auroral Snapshot Explorer (FAST) science mission. The design is based on a prototype Very Large Scale Integration (VLSI) level zero processing system, and utilizes VLSI telemetry data processing functional components, VLSI system technologies, and Object-Oriented Programming. The system performs level zero processing functions based on Consultative Committee for Space Data Systems (CCSDS) data format [1], and features high data processing rates, highly automated operations, and Open Software Foundation (OSF)/Motif based Graphical User Interface (GUI).

1. INTRODUCTION

Fast Auroral Snapshot Explorer (FAST), scheduled for launch in August 1994, is the second mission of Goddard Space Flight Center (GSFC) Small Explorer (SMEX) program. Through a set of extremely high-resolution measurements unavailable in the past and at altitudes unattained by sounding rockets, FAST will provide the first opportunity to resolve the wealth of fine structures inside the auroral acceleration zone. The Principal Investigator (PI) at University of California, Berkeley, Science Operation Center (UCSOC) has defined mission's primary scientific objectives to investigate the plasma physics of the low-altitude auroral zone; the production of waves, double layers, and solitons by electrons and ions; nonlinear wave interactions; and the acceleration of electrons and ions by waves and electric fields.

The scientific phenomenon that FAST will study is very dynamic (temporally and spatially) and not entirely predictable. These characteristics create the following operational requirements:

a. To obtain high resolution-measurements, the FAST will collect and store science data at very high data rates. To ensure significant data is not lost, high volume data dumps (up to 337 Megabytes (Mbytes) at downlink rates up to 2.25 Megabits per second (Mbps)) will be required at ground stations around the world for up to 11 passes per day.

b. The types of science of interest depend on auroral conditions. During peak mission operations, project scientists will provide realtime changes to the planned command inputs to refine and improve the science experiments. To accurately predict auroral conditions and perform science planning, scientists require receipt of raw data in realtime and processed instrument data within two to four hours of spacecraft downlink.

Dumping large volume of high rate data over ground stations around the world and requiring near realtime science data processing, the FAST mission has posed a major challenge for NASA's ground data systems in its class [2].

To meet this challenge, the Data Systems Technology Division (DSTD) at GSFC is developing a new-generation Packet Processing System (PPS) for FAST science data processing. In order to deliver a high-performance system within 15 months, the development of this FAST PPS utilizes the functional component approach developed by the DSTD, and bases the system design upon state-of-the-art VLSI and software technologies. The system will support CCSDS telemetry format and perform frame synchronization, packet reassembly and sorting, error checking, merging of realtime and playback data, and overlap deletion at rates up to l0 Mbps. In addition, a highly automated operation environment is designed to minimize human intervention in operation scheduling, control and monitoring, and data distribution.

The FAST PPS consists of mainly two subsystems: the Processing Subsystem and Control Subsystem. The Processing Subsystem is responsible for capturing and processing FAST telemetry data. The Control Subsystem provides for operations control, scheduling, status monitoring, data storage, and distribution. The system interface is based on the NASA Communications (Nascom) network and GSFC operational networks.

This paper discusses the general architecture and functionality of the FAST PPS system, with emphasis on the design of Processing Subsystem. It describes major FAST science data processing requirements, and how they are implemented through a set of generic and custom NASA telemetry processing hardware and software functional components developed by the DSTD over the last seven years. Reference 3

gives detailed description of Control Subsystem design, including the operation environment based on Object-Oriented Programming and GUI.

2. FUNCTIONAL REQUIREMENT

The FAST PPS is responsible for all instrument (including both engineering and science) data processing. As shown in the system context diagram (Figure 1), the spacecraft downlink is supported by an array of NASA ground stations around the world, including Wallops, Virginia; Poker Flats, Alaska; Deep Space Network (DSN) station in Canberra, Australia; Ground Network (GN) station at Merritt Island, Florida and Santiago, Chile; and Europe Science Agency (ESA) station in Kiruna, Sweden. Captured telemetry data is transmitted as a serial data stream to the FAST PPS at GSFC in Greenbelt, Maryland through Nascom circuit in standard 4800-bit blocks at rates up to 1.5 Mbps. Data received at Kiruna station will be recorded on magnetic tapes and shipped to the PPS. Additionally, the system will receive weekly pass schedule from Wallops station via a Remote User Schedule Terminal (RUST).

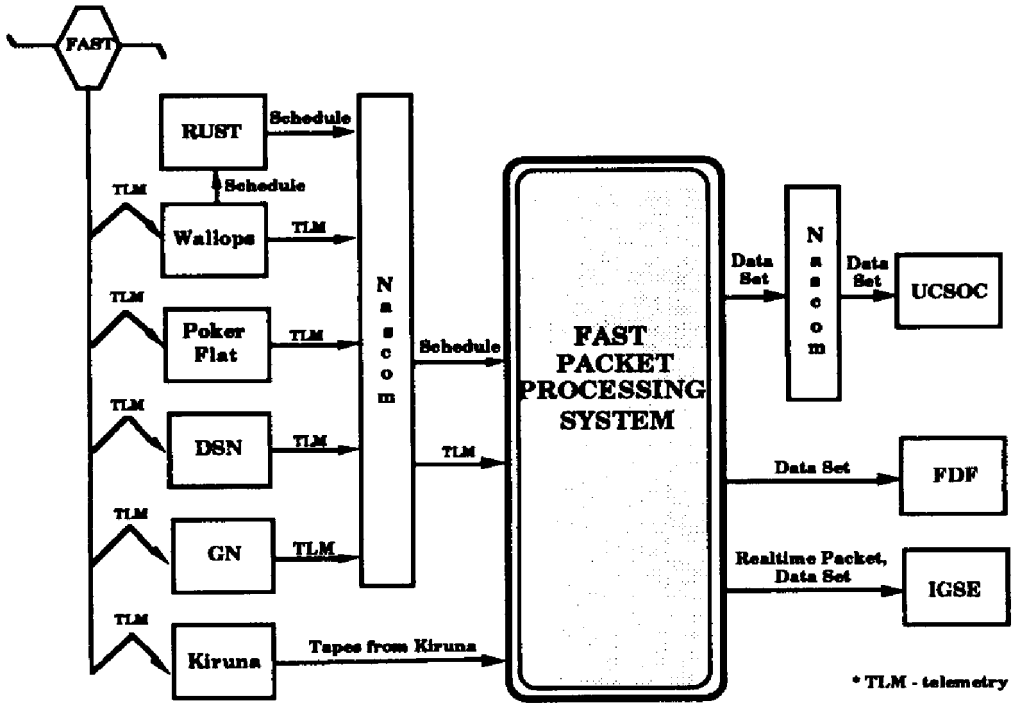

Figure 1. FAST Packet Processing System Context Diagram

The major processing functions required for the FAST PPS include:

- Support of CCSDS packet telemetry data format.
- Recording of all received Nascom blocks for 2-year raw data archive.
- Nascom deblocking and error checking.
- Frame synchronization and error checking.

- Packet extraction, reassembly, sorting and grouping.
- Forward-time-ordering.
- Data merging and overlap deletion.
- Quality checking and annotation.
- Realtime service.
- Production service.
- Data distribution to the PI within two to four hours of Loss of Signal (LOS).
- Support 8 Virtual Channels (VCs) and 128 Application Processes (AP's).
- Support tape playback at rates up to 10 Mbps.
- Support up to 11 passes and up to 5 Gigabytes (Gbytes) of data per day.
- Provide storage of processed data for 72 hours.

There are two services required by the FAST PI for instrument data processing. In the realtime service, packets from requested data sources will be routed to the Instrument Ground Support Equipment (IGSE) as soon as they are received at the PPS. In the production service, packets are grouped by AP ID into data sets. Packets in a data set are forward-time-ordered, with redundant data removed. The data sets are annotated with data quality information such as errors and gaps detected in the data set. The data sets are then distributed to the IGSE within two hours, and to the UCSOC within four hours, of Loss of Signal (LOS) at ground station. The PPS will also store processed data sets for 72 hours for possible retransmission to the user. Most data sets contain data received from a single spacecraft pass. However, the system is required to merge attitude data received within a 24-hour time period from up to 11 passes into data sets and to deliver them to the UCSOC and the Flight Dynamic Facility (FDF) of GSFC.

Highly automated operation is a primary design goal for the FAST PPS in order to reduce operation staff and thus operational cost. This automation will include all phases of system operations such as pass scheduling, session initialization and termination, data processing, data set generation and distribution, and status monitoring and reporting.

3. FAST PACKET PROCESSING SYSTEM BASELINE CONFIGURATION

Figure 2 shows the FAST Packet Processing System baseline configuration. It consists of two subsystems: the Processing Subsystem and Control Subsystem. The Processing Subsystem captures and processes FAST telemetry data, and generates realtime packets and data sets; while the Control Subsystem schedules and controls system operations, monitors status, stores and distributes data sets.

The Processing Subsystem comprises two identical processing strings named the VLSI Level Zero Processor (VLSI-LZP). One VLSI-LZP serves as a primary unit and

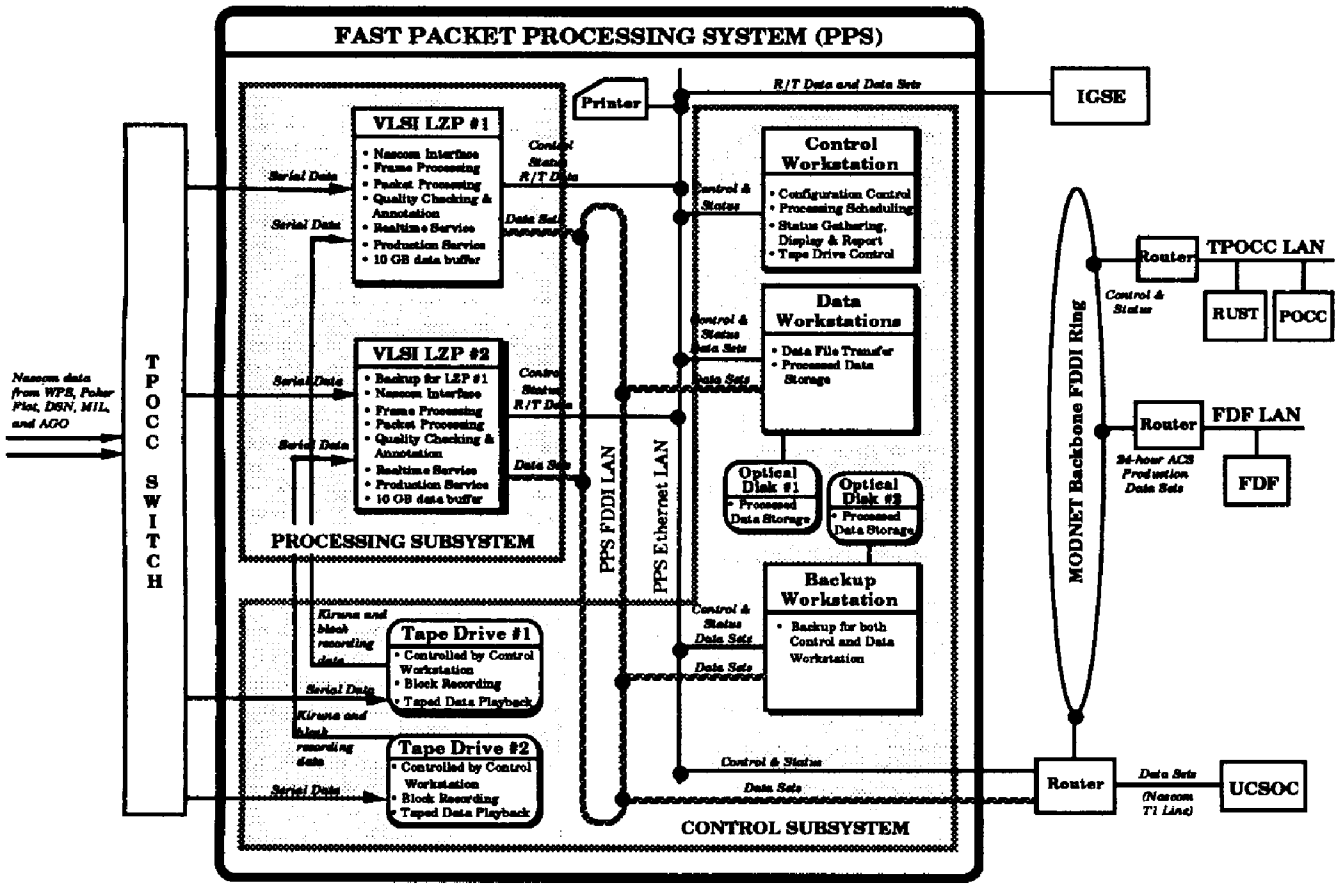

Figure 2. FAST Packet Processing System Baseline Configuration

the other as a backup unit. Each VLSI-LZP can perform level zero processing functions on telemetry data while the data is flowing through the system. The Control Subsystem is configured with three UNIX workstations and mass storage devices. The Control Workstation is responsible for pass scheduling, operations control, and status monitoring; the Data Workstation is responsible for data storage and data distribution to the user sites. The third workstation is configured as a backup for both the control and data workstations. Two high-speed tape drives are used for block recording and tape replay. Two optical disk drive systems serve as 72-hour storage for the data sets.

The VLSI-LZP's and workstations are connected through two Local Area Networks (LAN's): the PPS Ethernet LAN and the PPS Fiber Distributed Data Interface (FDDI) LAN. The Ethernet LAN is used for transferring control and status messages between the VLSI-LZP's and workstations. It is also used for delivering realtime packets and processed data sets to the IGSE. The FDDI LAN carries data sets from the VLSI-LZP's to the Data Workstation, and from the Data Workstation to user sites. A GSFC operational FDDI network, MODNET, will be used to connect the PPS to other ground system elements such as Payload Operation Control Center (POCC) and FDF.

Processed science data will be delivered to the UCSOC electronically through the Nascom network.

The focus of this paper is on the design of the Processing Subsystem. Since both VLSI-LZP's in the Processing Subsystem are completely identical and configured independently, the rest of this paper will discuss the design of VLSI-LZP.

4. VLSI-LZP SYSTEM ARCHITECTURE

The VLSI Level Zero Processor is built upon a conceptual VLSI Level Zero Processing System Prototype that has been developed by the DSTD [4]. The Prototype was completed in summer 1992, and has demonstrated the capability to level-zero process CCSDS telemetry data at sustained rates up to 20 Mbps.

The architecture of VLSI-LZP emphasizes the utilization of VLSI technologies and industry standards. Over the past 7 years, the DSTD has developed a set of VLSI Application-Specific Integrated Circuit (ASIC) chips that perform standard telemetry processing functions such as correlation, synchronization, and test data generation, etc. [5]. These chips are integrated into a set of custom-designed, highly reusable cards based on the industry standard Versa Module Eurocard bus (VMEbus). Each card performs one or more generic telemetry processing functions such as frame processing, packet processing, forward link processing, and data simulation. By the high-level integration of these common telemetry processing functions into VLSI chips and cards, the system achieves high performance, high reliability, and low maintenance.

To integrate these custom cards together with Commercial-Off-the-Shelf (COTS) VMEbus components into telemetry data processing systems, a modular software package has been developed that provides a generic software platform. With this platform, a system designer can select and configure into a system various VMEbus processing cards depending on the given system processing requirements. Thus, the system based on this architecture offers high configurability, reusability, and upgradability.

Automated operation is emphasized throughout the system design at all levels. The design of VLSI-LZP ensures that all operations can be controlled by a remote host such as the Control Workstation, and that all status required for monitoring operations be collected and reported to the remote host. Once initialized for a pass, the VLSI-LZP requires no remote intervention to process data. The system will sustain its operation even if the remote host breaks down during a pass.

The VLSI-LZP, shown in Figure 3, contains a VMEbus system, a 10 Gbytes disk array system (disk farm), and dual power supplies. A high-speed Very Large Data Store (VLDS) tape drive is also configured in the 19" rack for recording and playback telemetry data. Figure 4 illustrates the system block diagram of the VLSI-LZP, which comprises four subsystems: the Control and Communication Subsystem (CCS), Frame Processing Subsystem (FPS), Data Set Processing Subsystem (DSPS), and Mass Storage Subsystem (MSS).

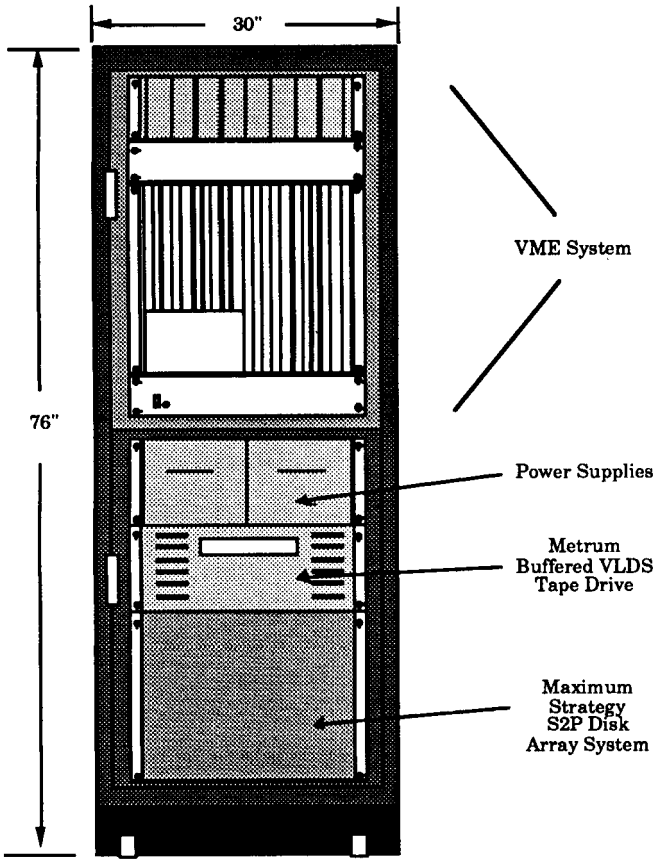

Figure 3. VLSI-LZP System Rack

The CCS provides system base functions, including command and control, network interface, and system data storage. The FPS receives serial telemetry data, performs standard Nascom and frame processing functions, and outputs synchronized frames to the DSPS. The DSPS extracts source packets out of the frames and delivers packets from specified sources to the IGSE in realtime. It sorts all packets by source, merges realtime and playback data into data sets, and removes redundant data from the data sets. The output of the DSPS is quality annotated data sets. The MSS serves as a large data buffer for data set processing and rate buffering. The detailed design of each subsystem is given in the following section.

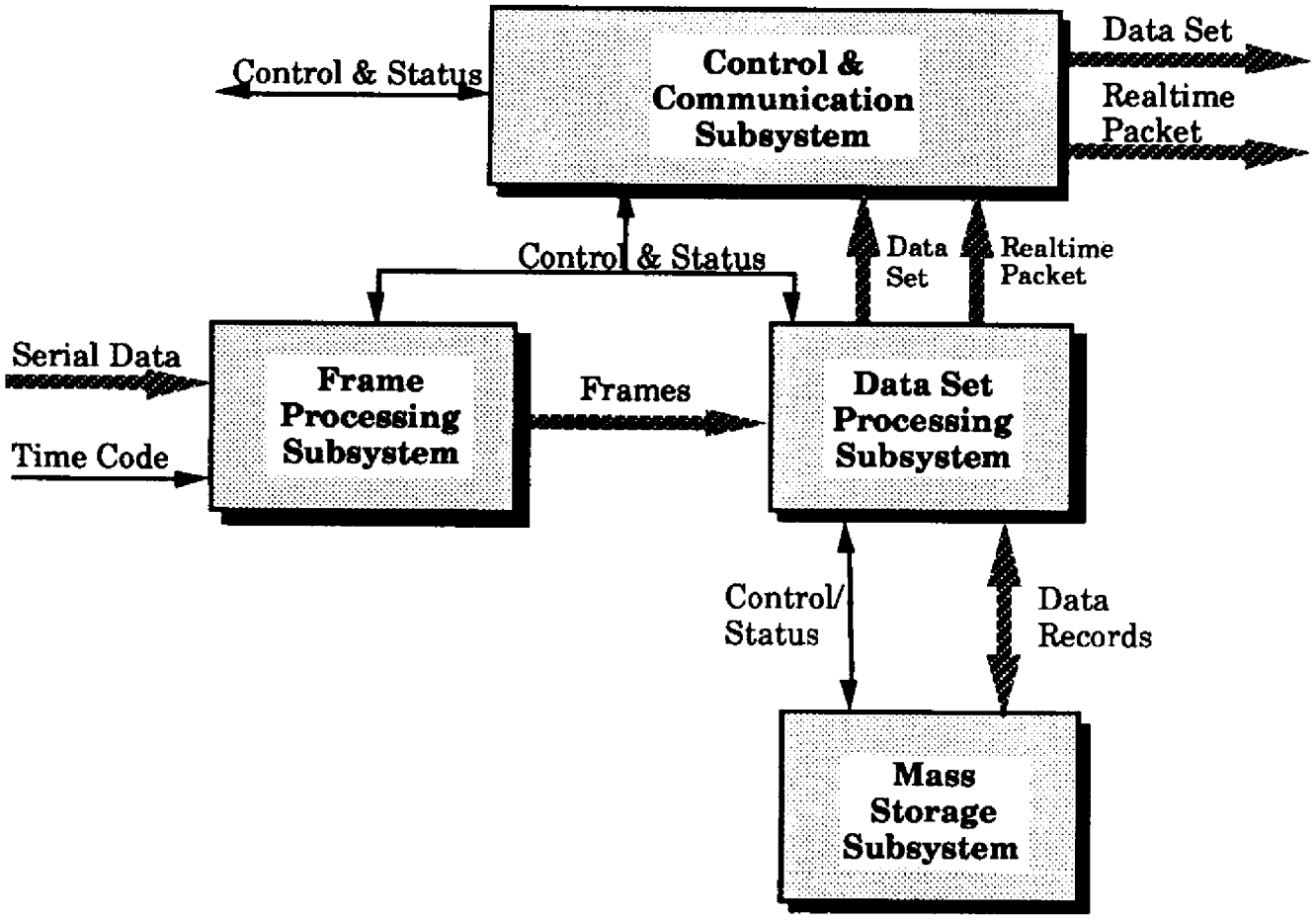

Figure 4. VLSI-LZP System Block Diagram

5. VLSI-LZP SUBSYSTEM DESIGN

The VLSI-LZP system functional block diagram is depicted in Figure 5, which shows a set of commercial and custom-designed VMEbus modules integrated in the VMEbus environment. These modules are grouped into the CCS, FPS, and DSPS subsystems, with the disk farm being in the MSS subsystem. The VMEbus is mainly used for transferring command and status information among the modules, while high-speed telemetry data is transferred from one module to the other through the VME Subsystem Bus (VSB) and custom telemetry pipeline implemented on J3 backplane.

5.1 The Control and Communication Subsystem

The CCS consists of a Master Controller Card, a Remote Interface Processor, an Ethernet Controller, a FDDI Interface Processor, a FDDI Interface Controller, a Disk Controller, two system disk drives, and 16-Mbyte battery-backed-up Static Random Access Memory (SRAM). All modules in the CCS are COTS products.

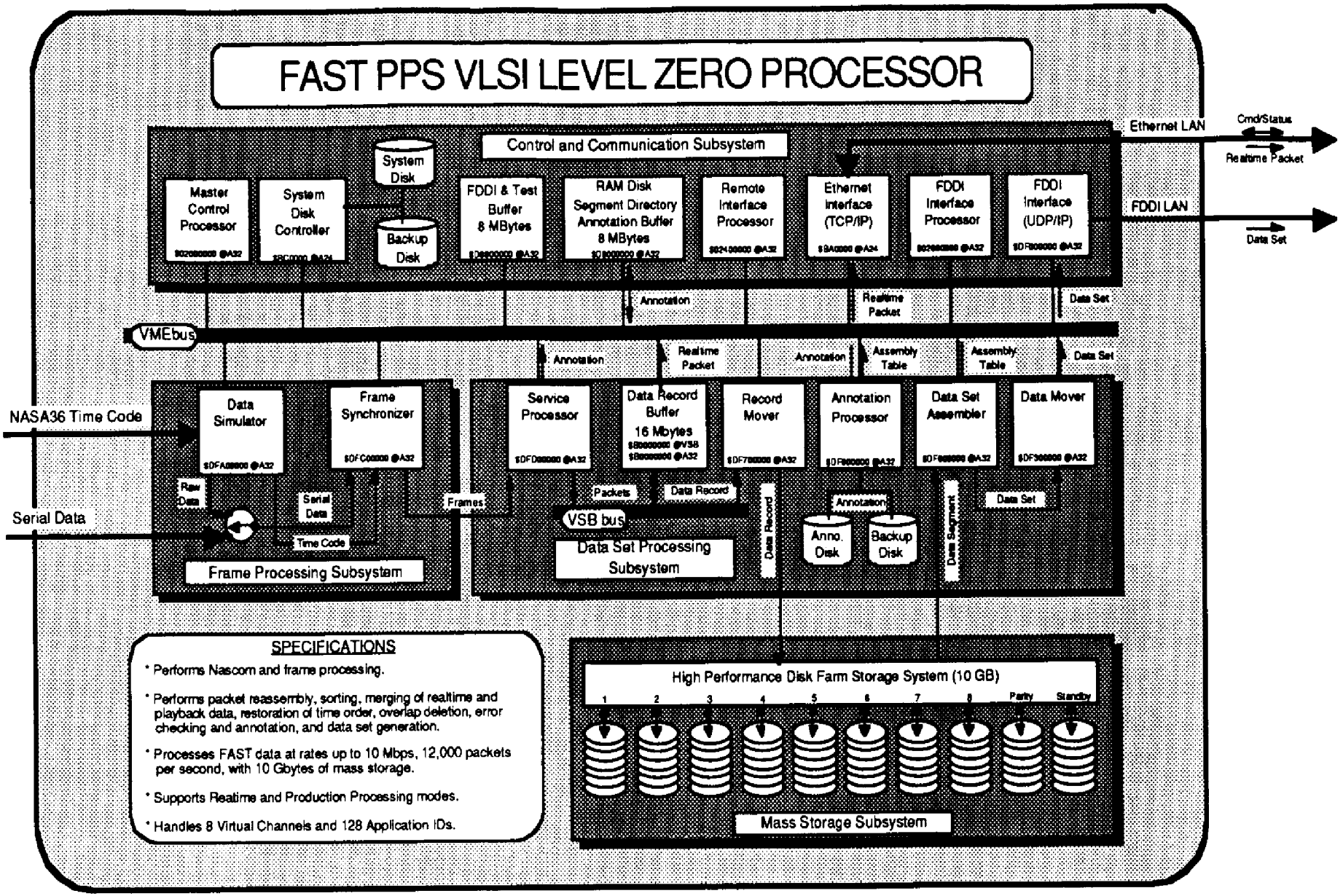

Figure 5. VLSI-LZP Functional Block Diagram

The Master Controller is based on a commercial VMEbus single board computer. It accepts commands and configuration parameters from the Control Workstation, interprets the commands, and sends appropriate subcommands to the other system modules. Based on the commands, it configures the system for processing sessions. The Master Controller also gathers housekeeping and processing status and reports them to the PPS Control Subsystem. If any processing statistics exceed user-specified thresholds, the Master Controller can send event messages to the Control Subsystem immediately to alarm the operator.

The CCS provides interface to two networks: the PPS Ethernet LAN and the PPS FDDI LAN. The Ethernet interface will be used for transferring command and status between the VLSI-LZP and Control Subsystem. It is also used for transferring realtime packets from the VLSI-LZP to the IGSE during the launch and instrument checkout mission phase. The Transmission Control Protocol/Internet Protocol (TCP/IP) is supported on the Ethernet LAN with sustained data rates up to 1.2 Mbps.

The FDDI LAN links the VLSI-LZP to the Data Workstation and is mainly used for transferring production data sets to the Data Workstation. The User Datagram Protocol (UDP) is supported for data transmission on the FDDI LAN and is enhanced with an acknowledgement scheme to prevent data loss. The global data storage need is satisfied with system disks and 16-Mbyte SRAM. The disk space is used for storing application programs and data set assembly files, while the memory space is used for maintaining a system database for high-speed access.

5.2 The Frame Processing Subsystem

The FPS consists of a Simulator and a Synchronizer designed and built by the DSTD [5]. Their functions are illustrated in a system data flow diagram (Figure 6), together with modules from the DSPS.

The Synchronizer performs Nascom deblocking and frame synchronization functions. It receives serial Nascom telemetry data and clock through a RS-422 interface, synchronizes the serial data to Nascom blocks, and checks for block CRC errors and sequence errors. It then extracts serial data in the blocks, synchronizes them to transfer frames according to a specified synchronization pattern and strategy. Frame CRC errors will be checked and the results will be reported in a quality trailer generated for and appended to each frame.

The Simulator can provide simulated FAST telemetry data for system self-test and diagnostic purpose. It also converts standard serial NASA36 time code to NASA PB1 time code for time stamping and for setup and calibration of the system clock.

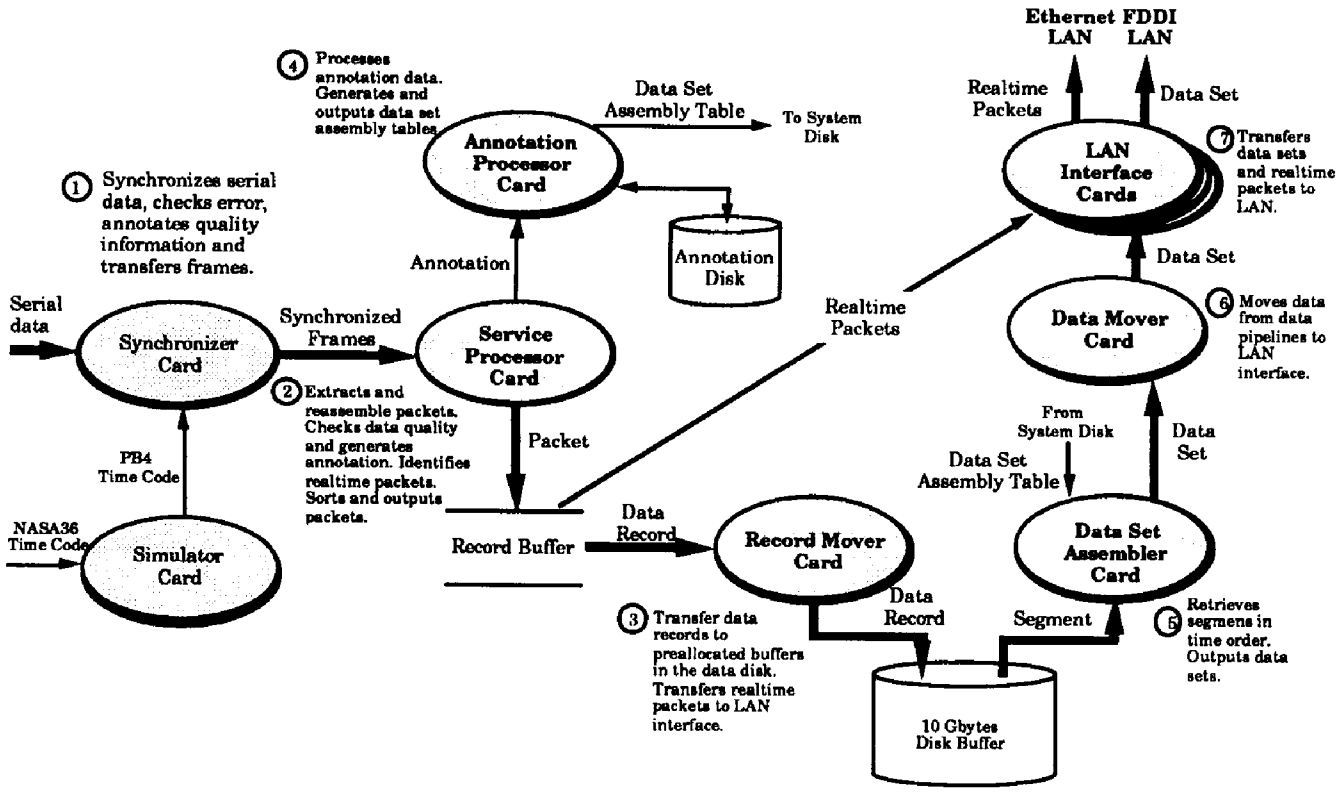

Figure 6. VLSI-LZP Data Flow Diagram

5.3 The Data Set Processing Subsystem

The DSPS consists of a Service Processor, a Record Mover, an Annotation Processor, a Data Set Assembler, a Data Mover, a dual bus (VME/VSB) DRAM module, and two SCSI disk drives. All processing modules are custom-designed and built by the DSTD and described in Reference 4. Their operations are illustrated in Figure 6.

The Service Processor receives transfer frames from the Synchronizer. It extracts packet data pieces from the frames, reassembles source packets, checks packet errors, and generates annotation for each packet. During a pass, packets from specified sources are output to the IGSE through the CCS as soon as they are received. The Service Processor also sorts packets by source and groups them into data records while outputting them to the record buffer on the VSB. Packet time code is extracted, and sent to the Annotation Processor together with packet quality information as annotation data for storage in the annotation disks. The Record Mover moves packets from the record buffer to the data disk whenever a record is full.

When the pass is over, the Annotation Processor examines the annotation data of each source to determine how to merge realtime and playback data into a production data set, how to forward-time-order the packets, and where the overlap boundaries and

redundant packets are. The result of this analysis will be stored in a data set assembly table file which will serve as an instruction set for assembling a data set.

Data sets are transferred to the Data Workstation via the CCS for data distribution. When outputting a data set, the Data Set Assembler reads in a data set assembly table, retrieves packets from the data disk in an order specified in the table, and sends them out to a data pipeline. The Data Mover then transfers the data set from the pipeline to the FDDI interface through the VMEbus. From the FDDI interface, the data set is routed to the Data Workstation for distribution to the PI. Included in the data set is also a quality annotation that specifies data set identification and accounts for packet errors, data gaps, and missing packets.

5.4 The Mass Storage Subsystem

In telemetry level zero processing, data merging and overlap deletion functions can only be accomplished after all data has been received. Therefore the VLSI-LZP needs to store at least up to 11 passes of data for the generation of 24-hour data sets. In addition, rate buffering is required between the telemetry input and data set output. Such data storage and rate buffering capability is provided by the Mass Storage System.

The MSS employs a Maximum Strategy SP2 disk array system (disk farm) with l0 Gbytes of disk space. Based on Redundant Array of Inexpensive Disks (RAID) technology, the disk farm is configured with eight data drives, one parity drive, and one stand-by drive. By stripping data across eight data drives, it offers data transfer rates up to 160 Mbps. There are two independent ports on the disk farm. One is connected to the Record Mover card for data input, and the other to the Data Set Assembler for data output. Through an internal command queuing scheme, the data input and output can be carried out concurrently.

As an operational system to support the FAST mission, the VLSI-LZP has very stringent requirements on data integrity and system reliability. These requirements are satisfied by the SP2 that combines a 48-bit Error Correction Code (ECC), parity disk drive, and stand-by disk drive to achieve true fault tolerant operations. Not only data dropouts can be corrected on the fly with the powerful 48-bit error correction polynomial which is stored with every sector written to disk, but also data availability and integrity is maintained even if a disk drive crashes. The data reconstruction and recovery process is transparent. The defective disk drive can be replaced on-line without interruption to normal operations.

Another major mass storage device employed is the Metrum buffered VLDS tape drive for Nascom block recording and data playback. The VLDS tape drive uses high-density helical scan recording technique to achieve data capacity of 10.4 Gbytes per standard T-120 high-energy cartridge, enough to hold three days of data during the peak of mission. With a large 16-Mbyte internal data buffer, the tape drive supports burst data rates from 0 to 160 Mbps and sustained data rates from 800 Kilobits per second (Kbps) to 32 Mbps. Data integrity is enhanced with powerful Reed-Solomon error correction code. The tape drive is controlled by the Control Workstation through a serial RS232 port.

6. SPECIAL PROCESSING CONSIDERATIONS

The data processing algorithm that FAST PPS employs was originally developed at the DSTD in 1989 [6], and implemented on the VLSI Level Zero Processing System Prototype. Taking advantage of new high-speed VLSI processing modules, it tedious data sorting and grouping functions done in realtime, thus drastically reducing necessary tdatabase management and production time. However, the initial algorithm is limited to spacecraft flying conventional tape recorders. Since the tape is always played back in reversed order, realtime and playback data are clearly separated by data direction and it can be assumed only one segment (a group of sequential packets of the same AP ID) will be generated from each source per session. Significant enhancement is needed to apply this algorithm to the FAST because the FAST mission utilizes a solid-state recorder. This section discusses new data scenarios and processing requirements encountered in the development of FAST PPS.

The FAST spacecraft uses a solid-state recorder of 128 Mbytes. As result, realtime data from instruments and playback data from the recorder are interleaved in a downlink data stream during a pass, as illustrated in an example in Figure 7. An instrument represented by APID 5 generates packets continuously. During the time the spacecraft is not in contact with a ground station, the packets will be stored in the recorder. During a pass, the recorded packets in VC 1 will be mingled with realtime packets in VC 0 together into a single physical bit stream down to the ground. This bit stream is relayed to the PPS and embedded packets are reassembled, sorted, and grouped into segments. However, the sorting can't be accomplished by AP ID alone, as done in the initial algorithm. A new definition for the data source has been defined as a combination of VC and AP ID's so that when sorting is done by source, realtime and playback data can be separated into different segments. In the example in Figure 7, source 1 is defined as VC 0/APID 5, and source 2 as VC l/APID 5.

Making things worse, data from a particular partition in the recorder may be dumped multiple times, and up to 180 packet groups of the same AP ID may be dumped out of

original time order, effectively creating hundreds of data segments from the same source during one pass.

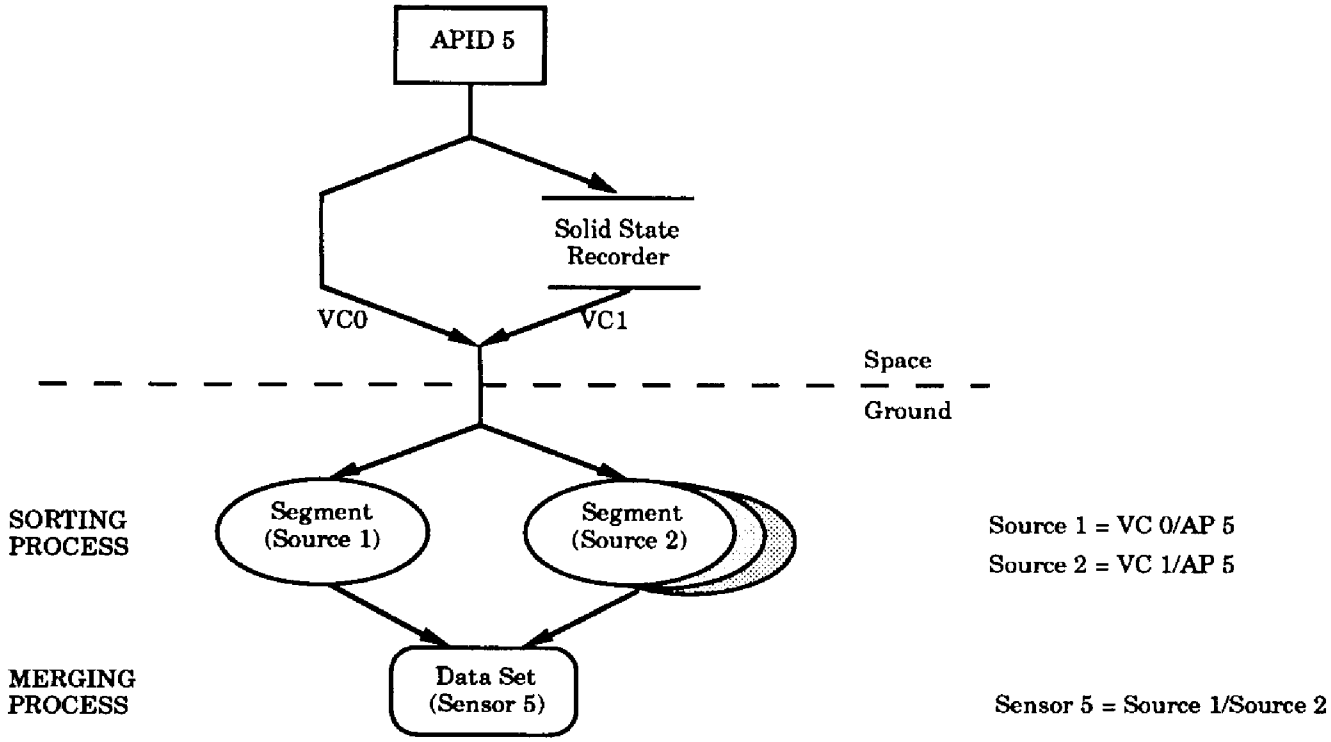

Figure 7. FAST Data Scenario

When generating a data set for a given AP ID, the PPS needs to merge all segments received from the AP ID. In the new algorithm, a sensor is defined as a combination of sources, typically realtime and playback data sources, for merging purpose. Again in the example, sensor 5 is defined as source 1/source 2 so that all segments of APID 5 will be merged into a data set for APID 5, as required by the FAST mission. In general, however, this algorithm gives future users greater flexibility to build their data sets. It allows a user either to merge partial data from an AP ID into a data set, or to merge data from multiple AP ID's into a data set.

7. SUMMARY

The design of the FAST Packet Processing System has been discussed with the implementation of its VLSI Level Zero Processor covered in details. Based on the VLSI technologies and advanced processing algorithm, the FAST PPS supports data processing rates up to 10 Mbps, handles up to 5 Gbytes of data per day, and provides realtime and near realtime science data processing and distribution to meet all FAST mission requirements.

The system features a highly automated operation environment to reduce operation staffing and operational cost. Data integrity and system availability are enhanced through powerful error correction scheme and equipment redundancy. Because of extensive use of the VLSI components and modular design, the system renders compact size, high reliability and high maintainability.

The FAST PPS is the first operational level zero processing system built upon the VLSI technologies and the functional component approach. The application of the functional component approach makes it possible to deploy the first system within 15 months. It demonstrates that high performance telemetry processing systems can be built in short time frame by utilizing a set of generic and custom-designed functional components both in hardware and software. It not only shortens development circle, reduces cost, but also provides expandability and upgradability in the future.

REFERENCES

1. "Packet Telemetry", CCSDS 102.0-B-2, Blue Book, Consultative Committee for Space Data Systems, January 1987.

2. "Engineering Study for Integrating the Level-Zero Processor into the SMEX TPOCC for the FAST Mission", Computer Science Corporation, NASA Contract NAS5-31500, December, 1992.

3. Costenbader, J., "TPCE - Reusable Software Components for Monitoring and Control of Telemetry Processing Systems", Proceedings of International Telemetering Conference, 1993.

4. Shi, J., Horner, W., Grebowsky, G., Chesney, J., "A Prototype VLSI Level Zero Processing System Utilizing the Functional Component Approach", Proceedings of International Telemetering Conference, 1991, pp. 519-531.

5. Chesney, J., Speciale N., Horner, W., Sabia, S., "High Performance VLSI Telemetry Data Systems", AIAA/NASA Second International Symposium on Space Information Systems, Pasadena, CA, September, 1990.

6. Shi, J., Horner, W., Grebowsky, G., Chesney, J., "Processing Algorithm for a VLSI Level Zero Processing System", Proceedings of International Telemetering Conference, 1989, pp. 925-936.

ACRONYMS AND ABBREVIATIONS

CCS	Control and Communication Subsystem
CCSDS	Consultative Committee for Space Data Systems
DSN	Deep Space Network
DSPS	Data Set Processing Subsystem
DSTD	Data Systems Technology Division
FAST	Fast Auroral Snapshot Explorer
FDDI	Fiber Distributed Data Interface
FDF	Flight Dynamic Facility
FPS	Frame Processing Subsystem
Gbytes	Gigabytes
GN	Ground Network
GSFC	Goddard Space Flight Center
IGSE	Instrument Ground Support Equipment
LAN	Local Area Network
LZP	Level Zero Processor
Mbps	Megabits per Second
Mbytes	Megabytes
MSS	Mass Storage Subsystem
Nascom	NASA Communications
NASA	National Aeronautics and Space Administration
PI	Principal Investigator
POCC	Payload Operation Control Center
RAID	Redundant Array of Inexpensive Disks
SCSI	Small Computer System Interface
SMEX	Small Explorer
SRAM	Static Random Access Memory
UCSOC	University of California, Berkeley, Science Operation Center
VLSI	Very Large Scale Integration
VME	Versabus Module Eurocard
VSB	VME Subsystem Bus