

# **ASYNCHRONOUS TRANSFER MODE (ATM) CONVERSION DEVICE (ACD)**

**Carol Harris, Michele Mascari, Kevin Rice,  
Jeff Smith, John Steedman**

## **ABSTRACT**

The Asynchronous Transfer Mode (ATM) Conversion Device (ACD) System is based on state-of-the-art ATM technology. The system interfaces between high-rate ECL/RS-422 raw data bitstreams and Synchronous Optical Network (SONET) OC-3 fiber. The SONET OC-3 interface uses ATM Adaptation Layer Type Five (AAL5) format.

The system exceeds its 50 Mbps raw data, single stream requirement and provides single stream raw data throughput at rates up to 75 Mbps. With ATM and SONET packaging overhead, this translates into 90 Mbps on the OC-3 fiber.

In addition to high-rate throughput, the system provides multiplexing and demultiplexing of multiple stream throughput based on the ATM cell header Virtual Path and Virtual Channel Identifier (VPI/VCI) values. The system is designed with the flexibility to provide between three and six throughput channels. All of which are multiplexed/demultiplexed to and from the same OC-3 interface. Multiple stream cumulative raw data throughput rates of up to 80 Mbps, or 96 Mbps on the fiber, have successfully run.

## **KEY WORDS**

Telemetry Processing, ATM, SONET OC-3C, AAL5 frames

## **INTRODUCTION**

The Asynchronous Transfer Mode (ATM) Conversion Device (ACD) System is an evaluation system for the NASA Communications (Nascom) division of Goddard Space Flight Center (GSFC). The system is being examined as an option to replace high-cost satellite links that transfer telemetry data between remote sites.

To provide more flexibility and lower communication costs, the ACD System was developed as a terrestrial solution for high-rate data transmission between remote sites. It provides a bi-directional interface between a serial data source and an ATM OC-3C

interface, which provides a line rate of 155.52 Mbps. ATM bandwidths can be purchased on an as needed basis and also provide high-rate single stream capabilities.

The ACD System is a Versa Module Eurocard bus (VMEbus)-based system that integrates commercial hardware with a custom designed board and custom application software. The system includes multiple processors and the software runs in the VxWorks operating environment. The system includes a Small Computer System Interface (SCSI) disk that houses operational code and local set up files so that it can run stand alone, but the system also includes a network interface that allows configuration files to be transferred and the SCSI disk to be mounted into an Ethernet environment.

To provide debug and control that is not dependent on a network, the ACD System can be setup and monitored by a custom local operator interface. However, the system also includes a Simple Network Management Protocol (SNMP) MIB II, which allows the user to integrate system control and monitoring into a standard SNMP Operator Interface that controls other network equipment. The system's SNMP interface allows it to be remotely operated via an Ethernet connection.

The design of the ACD System chassis and power supplies were customized to support the prospective needs of NASCOM or similar applications. The system chassis is a standard 21" rack mountable chassis, but it houses two separate 10 slot VMEbus backplanes. The split-backplane feature essentially allows two ACD Systems to be housed within the same chassis, which reduces equipment space requirements for additional bandwidth. Also, since the system's most likely use is critical data transmissions, each ACD System runs off redundant, hot-swappable power supplies.

## **THEORY OF OPERATION**

### **Hardware Architecture**

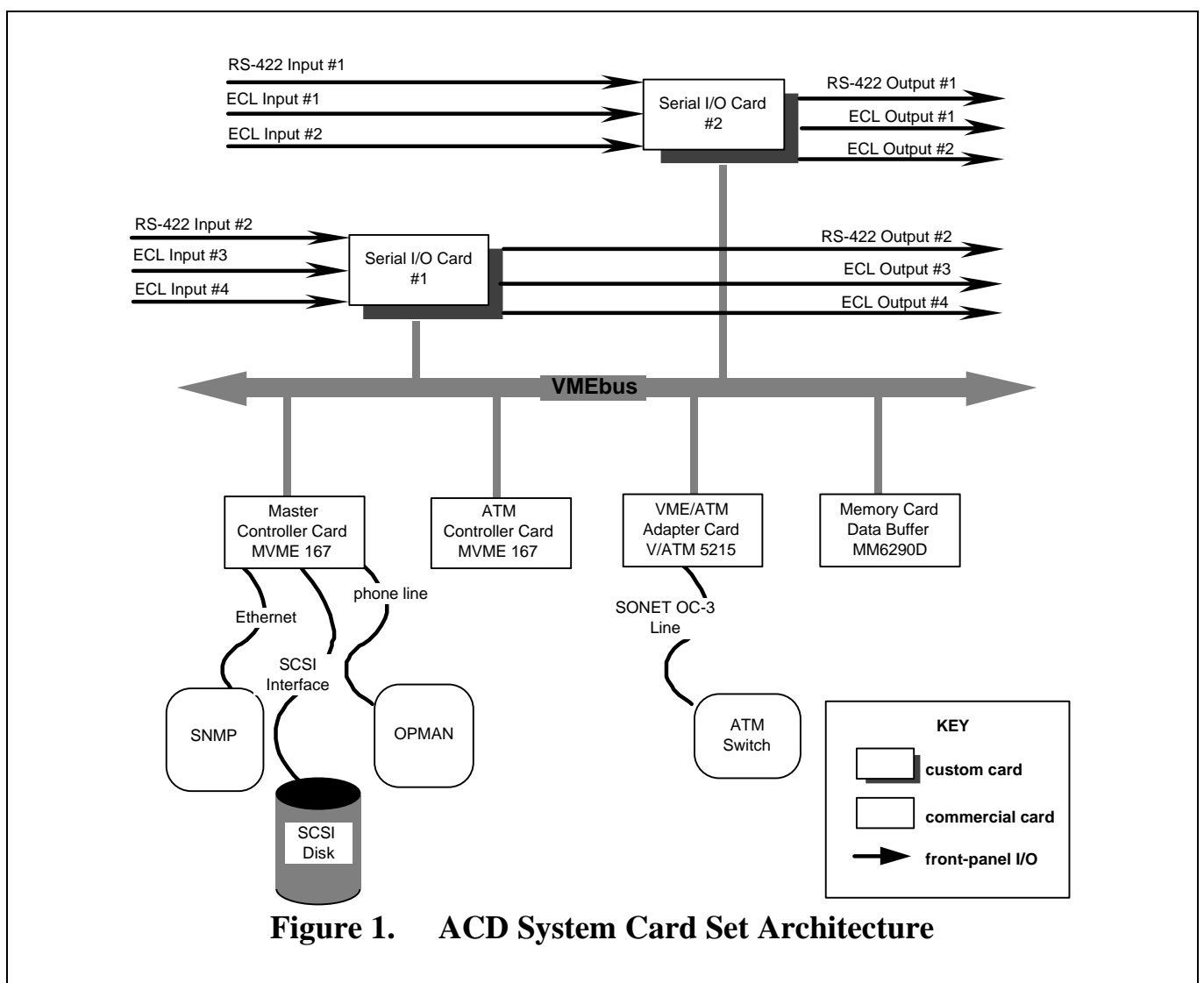
The ACD System card set is comprised of the following components:

- (1) (one) MVME 167 32B/32M MCC
- (2) (one) MVME 167 32B/32M ATM Controller Card
- (3) (one) MM6290D 32M Memory Card
- (4) (one) Interphase V/ATM 5215 Adapter Card
- (5) (one or two) Custom-designed Serial Input/Output (SIO) Card(s)
- (6) (one) Seagate SCSI Disk Module

All components interface via the VMEbus. Figure 1 illustrates the ACD System card set architecture. Figure 2 illustrates two ACD System card sets installed in the same chassis and each card set supported by redundant, hot-swappable power supplies.

### System Functionality

Each ACD System card set is either transmitting ATM cells to the OC-3C interface or it is receiving ATM cells from the OC-3 interface. Therefore, for simplicity, the ACD Card Set mode of operation is identified as either Transmit Mode (transmitting ATM cells) or Receive Mode (receiving ATM cells). System Input/Output (I/O) is via a rear I/O panel. Cables route I/O to and from the correct component front panels.



In Transmit Mode, raw data is input via a SIO Card ECL or RS-422 interface. It is then transferred via DMA across the VMEbus to memory. Once in memory, the V/ATM 5215 Adapter Card DMA's data across the VMEbus to its OC-3C output interface. A custom

memory buffer scheme called “zippy queues” is used to maintain order during memory data transfers and maintain the address of the next data due for transfer and the next memory location available for data storage. The V/ATM Adapter Card is setup and controlled by the ATM Controller Card. The V/ATM board packages data into AAL5 frames, ATM cells and OC-3C format. To maximize data transfer speeds, the ACD system utilizes VME64 for all data transfers.

Receive Mode is the reverse of Transmit Mode. OC-3C ATM cells that contain AAL5 formatted data is input from the fiber line to the V/ATM Adapter Card. The adapter card strips off all packaging, and then transfers data across the VMEbus using DMA to the correct zippy queue location in memory. Once in memory, the SIO Card DMA's data across the VMEbus from memory to one of its output channels.

If the ACD System were a single channel system, processing would remain very simple and throughput rates could be increased. However, each ACD card set can include one or two SIO Cards, and each SIO Card has three I/O channels: one RS-422 channel and two ECL channels. The multiple channel multiplexing and demultiplexing adds several complications to Receive Mode processing. However, Transmit Mode remains fairly simple. In Transmit Mode, one zippy queue is mapped to each of the SIO Card channels. This means that data from each SIO Card channel is sent to a separate zippy queue. The ATM subsystem software is setup so that data from each zippy queue is packaged into ATM Cells with VPI/VCI header fields set to specific values. The ATM cell VPI/VCI header value can be used to identify data from a specific input channel.



**Figure 2. Two ACD Card Sets with Redundant, Hot-swappable Power**

The main complication for multiple stream processing in Receive Mode is a result of the V/ATM Adapter Card design. That card requires a memory address be allocated for data from the next input ATM cell before that cell is even received. Therefore, it is not possible to read the VPI/VCI value of a cell header, and then route the cell data to a zippy queue designated to that specific VPI/VCI combination. To overcome this limitation, a novel, but complicated, software routine is used. The software sends all received data to the same huge memory buffer area. The VPI/VCI value can be read once the data is already buffered. So at that point the ATM subsystem software reads the header value of the data it just transferred, and then sends the address location of the data already transferred to a zippy queue designated for that VPI/VCI combination only. The SIO Card reads the address from the zippy queue, and then transfers data from that address out the correct channel. Once the data is output from the address, the SIO Card software must indicate that the address is now available for new data. A series of zippy queues are also used to maintain the available memory locations.

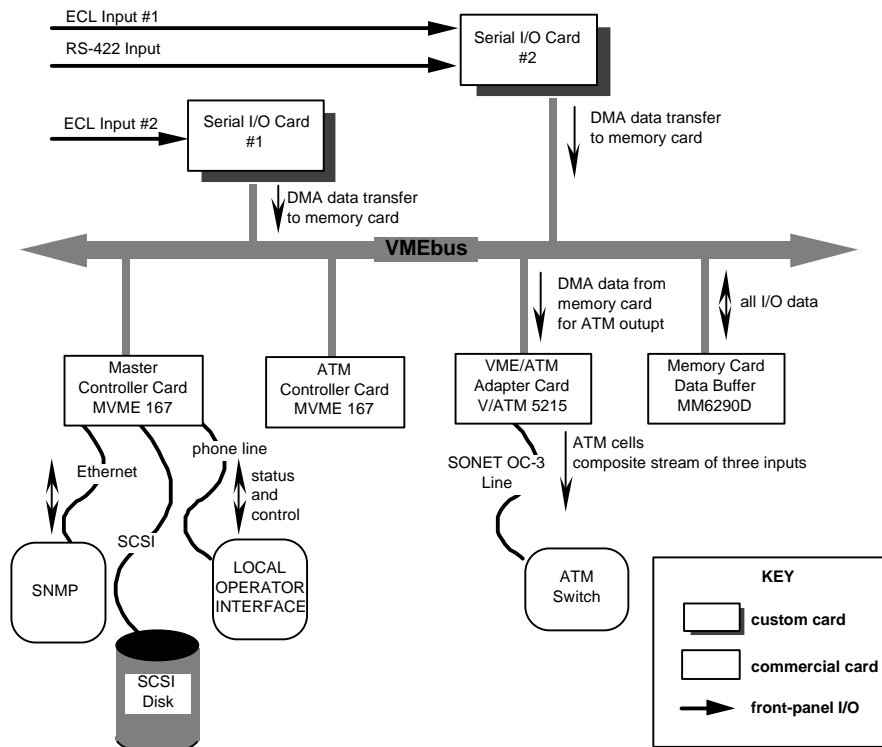
Operator setup, which can be done from either SNMP or the local interface, defines the VPI/VCI value assigned to each channel. It also defines the raw data rate to be throughput on each channel. Throughput rate setup is critical in Receive Mode, which requires an Numerically Controlled Oscillator (NCO) be set to clock data out.

### **Data Flow**

In the ACD System, each component independently performs the functions for which it was designed, but simultaneously reports status and sets indicators to show the state of processing. Each component receives its directions (setup) and reports status to the one central point, which is the Master Controller Card. The local operator interface resides on the MCC, and the MCC is the card with which a remote operator interface communicates.

By setting indicators, which may be a semaphore or an entry in an array, the components can directly communicate with each other without requiring the MCC as an interface. This type of communication is most typically done when components are passing data.

Figures 3 and 4 illustrates the receive and transmit data flows through the ACD card set.

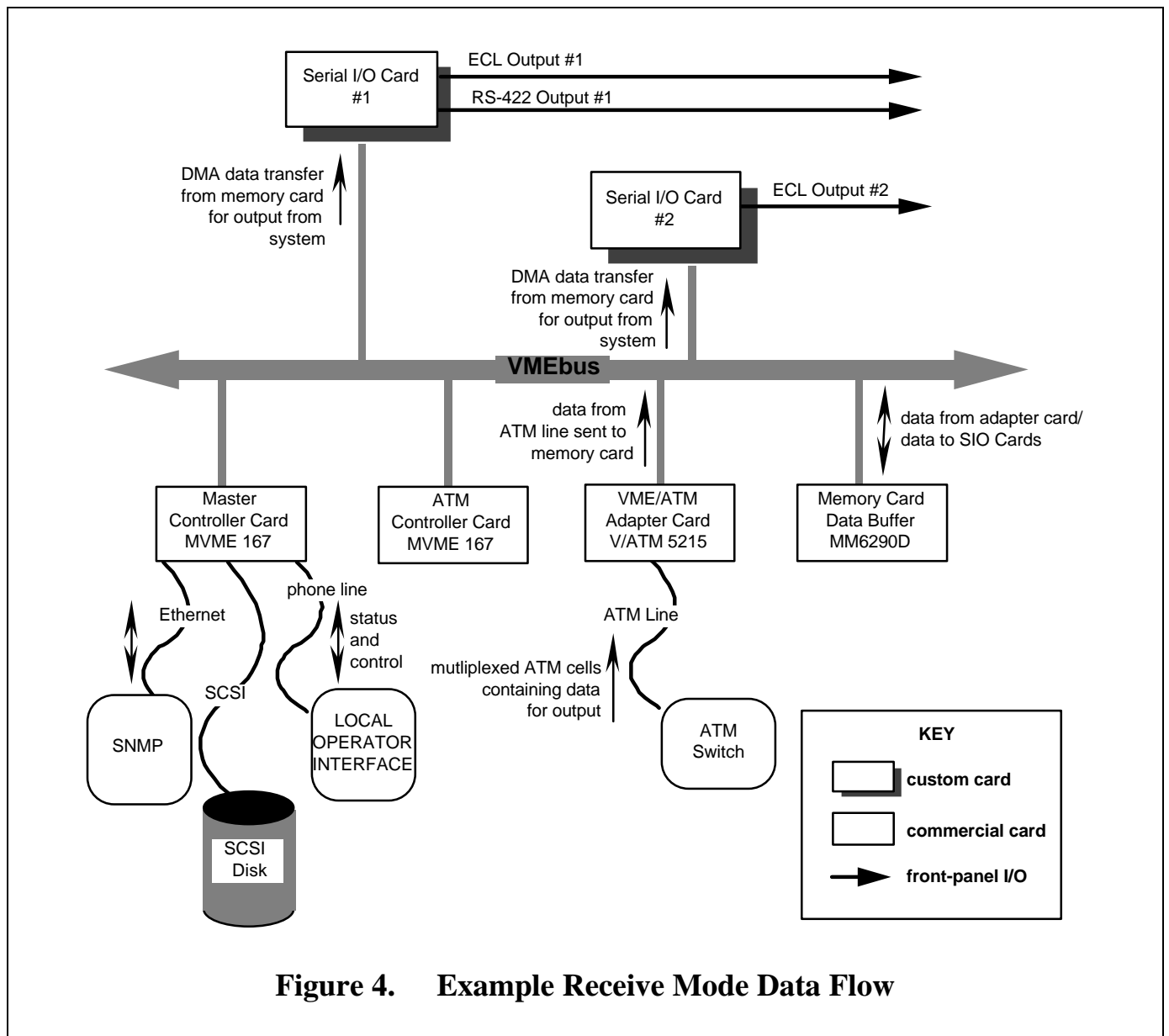


**Figure 3. Example Transmit Mode Data Flow**

## Software Environment

Two software environments, VxWorks Commercial-Off-the-Shelf (COTS) and Modular Environment for Data Systems (MEDS), (internally developed) support the variety of commercial and custom hardware installed in the ACD System. VxWorks provides a real-time operation environment; MEDS controls and monitors the system and the movement of data throughout the system. To control and monitor the system, SNMP software interfaces with MEDS.

VxWorks is a high-performance, real-time operating system and a powerful development environment for real-time applications. VxWorks includes a fast and flexible run-time system, powerful testing and debugging facilities, and a UNIX cross-development package (at the core of which lies VxWorks' extensive UNIX-compatible networking facilities). The VxWorks system can operate in a standalone manner, or can be networked with other systems running VxWorks or UNIX.



**Figure 4. Example Receive Mode Data Flow**

MEDS is a real-time environment for card telemetry processing systems that provides a mechanism to control the system and report status. MEDS manages the mix of cards that are part of the ACD Card Set. MEDS works in conjunction with VxWorks to provide the system software platform upon which application-specific code is developed.

The overall MEDS software design is modularized into packages that supply general purpose system functions, such as operator interface support, status gathering support, command handling support, interprocessor communications, and network communications. Each package implements a set of functions that serve as a resource to application software, while hiding the details of their implementation. Consistent interfaces have been defined for each package, so code within a package can be changed without affecting the application code if the functions and interfaces remain the same.

A MEDS-based system is built by adding custom code to the general purpose MEDS code, which spares the application developer the burden of creating an infrastructure for each new system. It also adds consistency to system design, implementation, and maintenance.

## **CONCLUSION**

The ACD System meets and exceeds its original requirement to provide single stream raw data throughput rates of 50 Mbps. It can provide single stream throughput at rates up to 75 Mbps. The single stream rate limitation is the receive side NCO. Multiple stream processing total throughput capabilities decrease with the number of data streams being processed. Two streams can run at a cumulative rate of 80 Mbps, and rate capabilities decrease from there. The major limiting factor in system processing is the ATM interface board chosen.

The DSTD works with the GSFC Office of Commercial Programs to commercialize the technologies that it prototypes. The goal of the commercialization effort is to allow NASA to procure low cost/high performance ground acquisition systems for future space and earth science missions. The approach that DSTD uses is to license and commercialize technology prototypes, including chips, boards, and software, through the Office of Commercial Programs.

For technology transfer information about the ACD System or any custom component included in the system please contact:

## **ACKNOWLEDGMENTS**

The authors would like to acknowledge Steve Koubek, Kevin Mulcahy, and Aiping Wang for the roles they played in the design and implementation of this system.