IRIG-106 PCM IMPLEMENTATION UTILIZING CONSULTATIVE COMMITTEE FOR SPACE DATA SYSTEMS (CCSDS)

by
Casey Tubbs
SCI Technology, Inc.
8600 South Memorial Pkwy
Huntsville, Alabama 35802
(205) 882-4267

ABSTRACT

Asynchronous data sources such as those associated with Space Based Radar create a unique problem for Time Division Multiplexed (TDM) Pulse Code Modulation (PCM) frame formats. The problem consists of data arrival based on external occurrences such as target tracking, and not due to sampling polls from internal sequencers. Reserved time slots for asynchronous data must be provided within the synchronous TDM telemetry stream. This increases the required bandwidth to transfer collected data to ground sites proportional to the worst case arrival rate of asynchronous data and the maximum latency allowed for the application.

Asynchronous data is readily handled by the Consultative Committee for Space Data Systems (CCSDS) recommended formats without the need to increase the bandwidth disproportionately. The recommendation maintains the ability to provide synchronous telemetry data collection and transmission provided by the TDM PCM frame formats.

This paper provides an implementation of CCSDS recommendations and addresses the methodology of merging asynchronous and synchronous data sources without the prerequisite increase in bandwidth associated with purely synchronous TDM approaches. Additional implementation details are provided for the implementation of a Telemetry Operation Procedure (TOP) to downlink error free telemetry frames. The TOP is not currently supported within the CCSDS recommendation. The implementation is provided through the Micro Packaged Data Acquisition and Control Systems developed by SCI Technology in Huntsville, Alabama.

INTRODUCTION

This paper discusses asynchronous data sources and the method whereby they can be multiplexed into a synchronous telemetry stream without an adverse increase in bandwidth. The discussion proceeds from a typical flight system application where one telemetry processor generates a telemetry stream which is multiplexed by a second telemetry processor into a final output telemetry stream. The output telemetry stream is transmitted to the ground where it is decommutated and telemetry points are time correlated. Figure 1 illustrates the system configuration. Contrasting approaches to the implementation of PCM formats and CCSDS formats are provided to demonstrate the benefits of packet telemetry for asynchronous applications with asynchronous data sources.

BACKGROUND

TDM PCM formats for telemetry transmission requires data be sampled and inserted into known and fixed time slots within the downlink format. The perceived gain in TDM PCM formatting and transmission is the ground system ability to determine the sample time of each and every telemetry point based on its arrival time at ground decommutators. The resolution and accuracy of time correlation of telemetry points multiplexed via TDM is dependent on several factors. These factors are illustrated in Figure 1.

Within the TDM PCM system of Figure 1 a sequencer sample table in TP2 identifies when data should be read from the input interfaces. The frequency of reads for a particular interface whether it be synchronous or asynchronous is fixed due to the cyclic nature of the sequencer. TP1 provides an asynchronous input to TP2. The frequency of reads of the TP1 interface within the sequencer sample table is selected to meet the maximum input data rate of the asynchronous data source balanced by the average rate of arrival over a major frame time period and the maximum allowable latency of asynchronous data words. To simplify the discussion, we will assume the asynchronous data arrives in packet format (minor frames), and the output PCM multiplexing occurs in submultiple frames.

If TP2 generates submultiple frames at the maximum arrival rate of asynchronous packets, then bandwidth requirements are maximized and arrival time uncertainty of asynchronous data is minimized. If submultiple frame generation is balanced across a major frame to meet the average asynchronous packet arrival rate, then bandwidth utilization is enhanced, transfer latency of asynchronous packets is increased and arrival time resolution is decreased. Arrival Time resolution can be offset through time stamps at the asynchronous interface. Latency of asynchronous data on the other hand cannot be offset by synchronous PCM multiplexing implementations without wasting bandwidth. The CCSDS implementation on the other hand can decrease the required bandwidth, increase the arrival time resolution, while minimizing the latency of asynchronous data arrival at the ground system.

It is also interesting to note that for TP2, while the asynchronous packets are multiplexed into a submultiple frame by the TP2 synchronous sequencer, no other telemetry data is collected. Ground processing can determine quite accurately when a packet arrived, but within a telemetry processor, no data is taken at the exact same time. CCSDS affords concurrent data sampling as well.

THE CCSDS SOLUTION

CCSDS is a packet telemetry format built around four layers of telemetry processing. Packet layer sequencers feed a segment layer sequencer which in turn feeds a channel and transport layer sequencer (5). This approach is similar to the PCM telemetry implementation discussed previously (TP1 is a distributed telemetry processor generating packets to TP2, which in turn provides transport layer multiplexing).

The CCSDS Implementation

With CCSDS the implementation can be synchronous or asynchronous. Under the synchronous implementation operating on asynchronous data sources the CCSDS suffers the same ineffective utilization of bandwidth as PCM as it requires a packet for a group of data sources be generated on fixed intervals. The fixed interval packets are multiplexed into segments at fixed interval, and the fixed interval segments are multiplexed into fixed interval channels for transportation to the ground. With asynchronous implementation, the bandwidth utilization is determined from the maximum latency requirements for all data types and the average arrival rate of asynchronous data. The latency requirement traded against the bandwidth utilization requirement determines the amount of buffering required on the spacecraft between multiplexing layers of CCSDS.

To offset the latent arrival of data at the ground processing facility, and to facilitate time space correlation of data points, everything in CCSDS is block time stamped with a 32 bit time code (6) providing a resolution of about 60 nsec for blocks of data at the data collection point. The ground is always able to correlate the data in time space regardless of the latency of arrival created by spacecraft buffering since all data words within a block are synchronously collected from the time stamp forward in time.

THE CCSDS ARCHITECTURE

The CCSDS architecture shown in Figure 2 provides a distributed telemetry sequencer architecture. This approach does not deviate from distributed telemetry processing under the PCM case. It does deviate from the PCM case in the method for interconnecting the distributed telemetry sequencers for data flow management. The interconnect controls implemented is the method whereby bandwidth utilization is maximized.

The architecture consists of smart I/O interface cards providing packets of data from asynchronous and synchronous data sources. The packet data from synchronous data sources are time stamped by the CCSDS sequencer card. The packets from asynchronous data sources are time stamped by the asynchronous data interface cards. Correlation of clocks between CCSDS sequencer and asynchronous data interface cards is provided through uplink command, onboard software and preflight simulations.

The Remote Interface Unit (RIU) of Figure 1 is implemented with identical CCSDS architecture up to the segment layer, however the chassis and number of cards are reduced. Within TP2, the interface to the RIU is through a synchronous serial interface card providing transmission rates up to 4 MHz.

The CCSDS Sequencer card provides dual sequencers which can be programmed to perform multiple multiplexing functions. CCSDS is just one format. Arbitration for bus access for the dual sequencers is performed by an I/O ASIC. When both sequencers are configured as synchronous sequencers, the card supports output of real time and stored mode telemetry in packet or TDM-PCM formats on two data links without software intervention. To support encryption under this mode, the card must be placed back to back with the KG Interface card, and be cross-strapped to the KG Interface card to avoid latency in transmission associated with bus access arbitration (4 bus cycles or 2 microseconds).

when the card is configured as shown in Figure 2, one sequencer is configured as a segment sequencer, and the other is configured as a channel sequencer. The segment sequencer operates on fixed sample and format tables identically to a TDM-PCM sequencer. However, to avoid single byte reads on the backplane, the collection of analogs, and serial digital data words is distributed to the interface cards. The analog to digital and serial I/O interface cards are downloaded with sampling tables. The segment sequencer starts an interface card sampling sequence through a sample poll command. A single sample poll command consists of a 32 bit time tag, and a sample sequence identifier. The 32 bit time tag is used as a time tag by the interface card to mark the start of a sample sequence. The buffer generated by the interface card as a result of a sample sequence is returned to the segment sequencer as a positive acknowledgement of the receipt of the next poll command. The time stamp is embedded in the returned buffer. Segments returned are stored in segment format with segment header information filled.

Discrete data points are retrieved by the segment sequencer through a block memory read. The time stamp is appended by the I/O ASIC on receipt of the first 32 bit wide discrete pattern at the backplane interface. Likewise the time tag is appended to every response received by the I/O ASIC and is available for insertion in the formatted segment buffers. Most spacecraft applications require less than 2048 discrete data points, therefore discrete segments consist of multiple fixed length discrete packets delimited by a 32 bit time stamp.

The channel sequencer builds transport frames through polls to the asynchronous data interface card and the segment sequencer at two times the maximum input packet rate. Polls to the asynchronous interface card are at a lower priority than the segment sequencer sampling polls, and are arbitrated by the I/O ASIC. When at least one full packet of data is available at the asynchronous interface, the channel sequencer, through the I/O ASIC retrieves as many of the packets of data possible without interference with the segment sequencer sampling. It requires 64 bus cycles (32 microseconds) to poll and retrieve a single 256 byte asynchronous data packet.

The channel sequencer concurrently generates up to 8 virtual channel CCSDS formatted transport frames. Multiplexing of segment sequencer output and asynchronous segments is performed according to segment maps downloaded to its control memory. Secondary headers within the transport frame (5) are supported, however the secondary header data field must be written to the channel sequencer by an external process such as a CPU or 1553 interface. When a virtual channel transport frame is complete, or a time out value has been reached for any segment of data within the virtual channel, the channel sequencer raises transport frame available flags for the KG interface card. Transport frames consisting of 4-256 byte segments are generated at a maximum rate of 1953 per second. This rate far exceeds the capability of current RF space/ground links, therefore the channel sequencer provides extended memory storage for generated transport frames. The extended memory is managed as a FIFO.

The channel sequencer implements a Frame Operation Procedure (FOP) through the FIFO buffer management and by assigning and tracking frame sequence numbers (virtual and physical). Through uplink command and on-board software, transport frames can be re-queued for transmission as long as they reside in the channel sequencer FIFO. The transport frames, minus the Command Link Control Word are retrieved by the KG Interface Card for real (9024 bits). When CCSDS transport frames require storage prior to transmission, the transport frames are retrieved from the FIFO by a mass storage interface controller.

Optional configurations allow for separate channel/transport and sequencer cards which are interfaced through the backplane. Under this configuration the segment sequencer card provides dual sequencers and the channel sequencer provides separate channel and a transport sequencers. This configuration provides an additional synchronous sequencer to generate transport layer secondary header data fields as a part of the CCSDS card set.

A KG Interface Card performs the encryption and final transport layer processing required to transport the channel data to ground CCSDS decommutators consistent with the Command Operation Procedure (COP) of CCSDS (7). The KG interface card provides the ability to retrieve transport frames from the channel sequencer FIFO or if directed can retrieve transport frames from mass storage.

The result of the layered multiplexing and the FIFO memory prior to the KG interface is the bandwidth is dynamically allocated to transport asynchronous frames or high priority synchronous frames. Bursts of asynchronous data are transmitted as fast as the KG interface data rate allows. When asynchronous data is not present, the bandwidth is utilized to transfer previously generated asynchronous and synchronous virtual channel data until the FIFO is empty. When the FIFO is empty, fill transport packets are generated until a new transport frame enters the FIFO.

TELEMETRY OPERATION PROCEDURE (TOP)

The buffered CCSDS implementation affords a flexible method for retransmission of transport frames which are received in error at the ground CCSDS decommutator. Retransmission requires additional bandwidth for error retries and additional buffering between the channel sequencer and KG interface. The added bandwidth under the synchronous implementation is wasted unless there is an error. Under the asynchronous application the bandwidth is dynamically allocated for retransmission.

The condition for error reporting is provided through the CCSDS Frame Access Reporting Mechanism (FARM) (7) which must be implemented on the ground at the receiving end of the transmission and reported to the FOP (7) within the transport layer (the transmitting end). The FOP within the channel sequencer provides the required storage for retransmission.

The first and second levels, FARM-0 and FARM-1 each lock out reception of transport frames when a frame is received in error. The FOP for these applications will go-back n-frames transfer a TYPE-B frame to unlock the FARM at the receiving end and initiate transmission of the transport frames over again. FOP-0 and FOP-1 are not the recommended approach for real time CCSDS transmission since everything backs up during retransmission and the buffering required can become extensive. They can be implemented for stored mode playback as the data is already in storage and can be retrieved without extra buffering.

The third level of error recovery, FARM-2 continues to receive transmission from the FOP even after error. For each frame received with error an error report is issued to the ground based command processor. The command processor generates retransmit commands through the uplink to the FOP. The FOP in the channel sequencer retransmits frames which are reported as received in error. The frames are stored for a programmed FIFO of n-deep frames for error retransmission. If the frame does not reside within the n-deep FIFO, a fill transport buffer is issued with the frame sequence number inserted to clear the FARM error for that frame. The FOP-2 is the preferred solution for real-time transmission of CCSDS frames.

CONCLUSION

While Time Division Multiplexed-Pulse Code Modulation (TDM-PCM) provides a highly capable method for correlation of telemetry sampling in time space for completely synchronous application, it has short falls for applications involving asynchronous data sources. CCSDS formats provide equally capable methods for time correlation under the synchronous and asynchronous applications without the shortfalls of TDM-PCM.

TDM-PCM requires extra bandwidth in the downlink to handle asynchronous data sources. The extra bandwidth is directly proportional to the worst case burst arrival rate of asynchronous data. CCSDS requires extra bandwidth in the downlink for asynchronous data sources. The extra bandwidth is less than that for the TDM-PCM system and is directly proportional to the average data rate of asynchronous data and the maximum latency allowed for arrival of asynchronous and synchronous data.

TDM-PCM is limited in time space correlation to many factors dealing with the transmission of the downlink telemetry, and the ground system relay of the transmitted data. CCSDS is limited in time space correlation to an equivalent of a 166.66 Mbps (60 ns resolution) link directly connected to a decommutator. Time reference accuracy with the CCSDS implementation is dependent on the ability of the ground to set the μDACS internal time stamp clocks to ground truth.

TDM-PCM provides no mechanism for retransmission of frames received in error. CCSDS provides retransmission capability however the Telemetry Operation Procedure (TOP) has not been defined and the uplink Command Frame format does not provide the equivalent Command Link Control Word involved with the Command operation Procedure (COP). The error frame retransmission must be commanded by the ground Command Processor.

TDM-PCM decommutated data is at the telemetry point, ID, Time tag level. Ground processing at the interface to the decommutator is limited to the input processing capability of a CPU to accept these data points. CCSDS is at the packet level thereby simplifying ground processing equipment since the CPU operates on buffered data rather than paired data points.

ACKNOWLEDGEMENTS

I would like to thank the SCI Technology, Inc., Instrumentation Department for their patient review of this paper. The μDACS Team within that group designed μDACS as an evolutionary replacement for the existing DACS. The DACS was designed for synchronous TDM PCM applications. Asynchronous multiplexing into a synchronous PCM link is a revolutionary requirement. The fact the μDACS can perform the function is a testimony to the Team's modular design.

REFERENCES

1. Hideaki Takagi, "LIMITED SERVICE SYSTEMS", Analysis of Polling Systems, The MIT Press, Cambridge, Massachusetts, 1986, 1-7, 118-142.

2. Telemetry Group, Range Commanders Council, "PULSE CODE MODULATION STANDARDS", IRIG 106-86 TELEMETRY STANDARDS, Secretariat, Range Commanders Council, U.S. Army White Sands Missile Range, New Mexico 88002, May 86.

3. Consultative Committee for Space Data Systems, "Source Packet Format",, CCSDS 102.0-B-2 BLUE BOOK, Packet Format, CCSDS Secretariat, Communications and Data Systems Division (Code-TS), National Aeronautics and Space Administration, Washington, DC 20546, USA, January 1987, 3-1 through 3-6.

4. Consultative Committee for Space Data Systems, "PACKET SEGMENTATION", CCSDS 102.0-B-2 BLUE BOOK, Packet Format, CCSDS Secretariat, Communications and Data Systems Division (Code-TS), National Aeronautics and Space Administration, Washington, DC 20546, USA, January 1987, 4-1 through 4-6.

5. Consultative Committee for Space Data Systems, "TRANSFER FRAME FORMAT", CCSDS 102.0-B-2 BLUE BOOK, Packet Format, CCSDS Secretariat, Communications and Data Systems Division (Code-TS), National Aeronautics and Space Administration, Washington, DC 20546, USA, January 1987, 5-1 through 5-15.

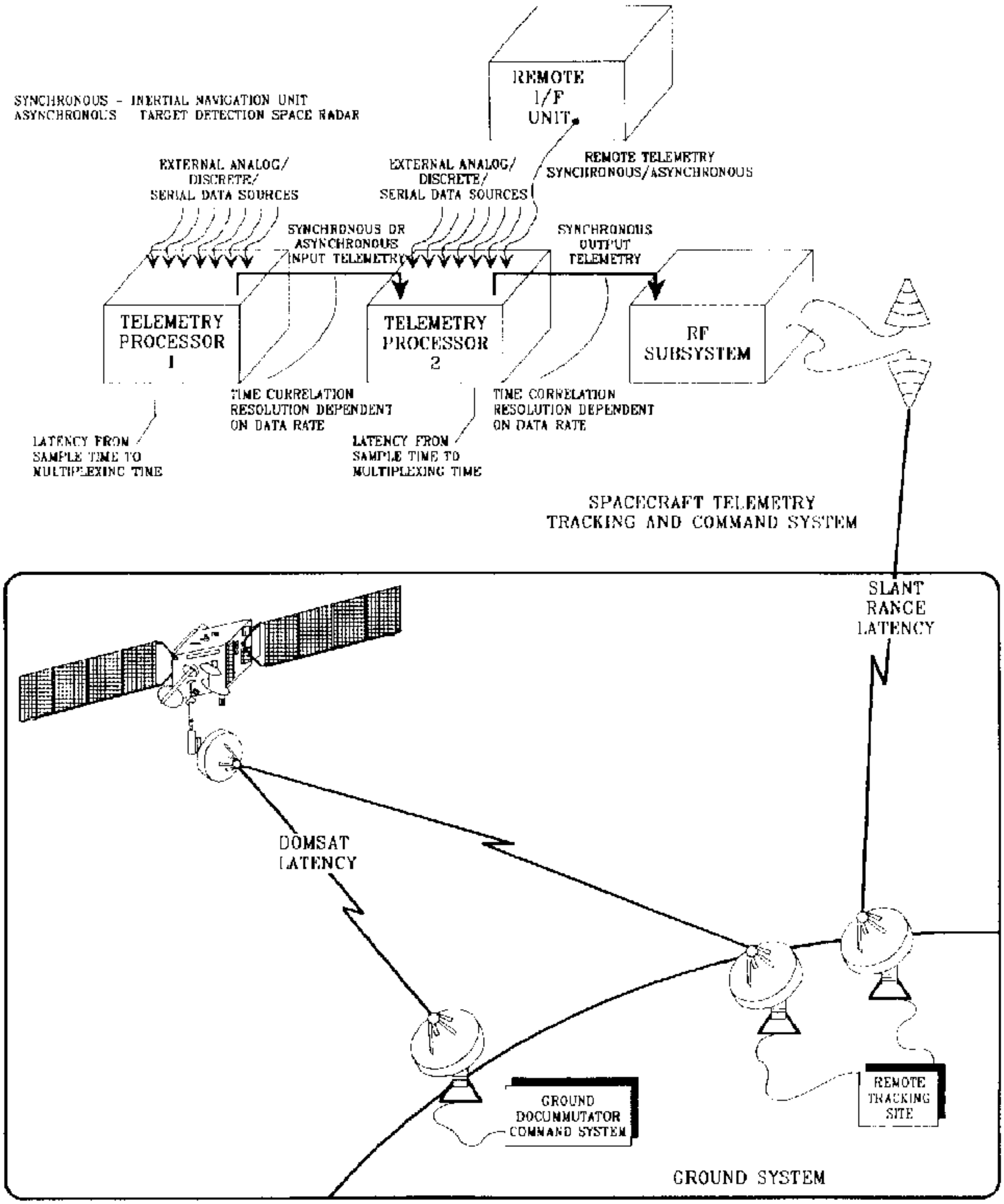
6. Consultative Committee for Space Data Systems, “TIME CODE FORMATS”, CCSDS 301.0-B-1 BLUE BOOK, Time Code Formats, CCSDS Secretariat, Communications and Data Systems Division (Code-TS), National Aeronautics and Space Administration, Washington, DC 20546, USA, January 1987, 2-1 through 2-6.

7. Consultative Committee for Space Data Systems, All, CCSDS 201.0-B-1 BLUE BOOK, Telecommand, Part 2, CCSDS Secretariat, Communications and Data Systems Division (Code-TS), National Aeronautics and Space Administration, Washington, DC 20546, USA, January 1987, 1-1 through 4-43.

ACRONYMS

CCSDS	Consultative Committee for Space Data Systems
COP	Command Operation Procedure
FARM	Frame Access and Reporting Mechanism
FIFO	First In First Out
FOP	Frame Operations Procedure
PCM	Pulse Code Modulation
TDM	Time Division Multiplexed
TOP	Telemetry Operation Procedure
TP1	Telemetry Processor 1
TP2	Telemetry Processor 2
μDACS	Micro Packaged Data Acquisition and Control System

FIGURE 1 - SPACE GROUND TELEMETRY SYSTEM



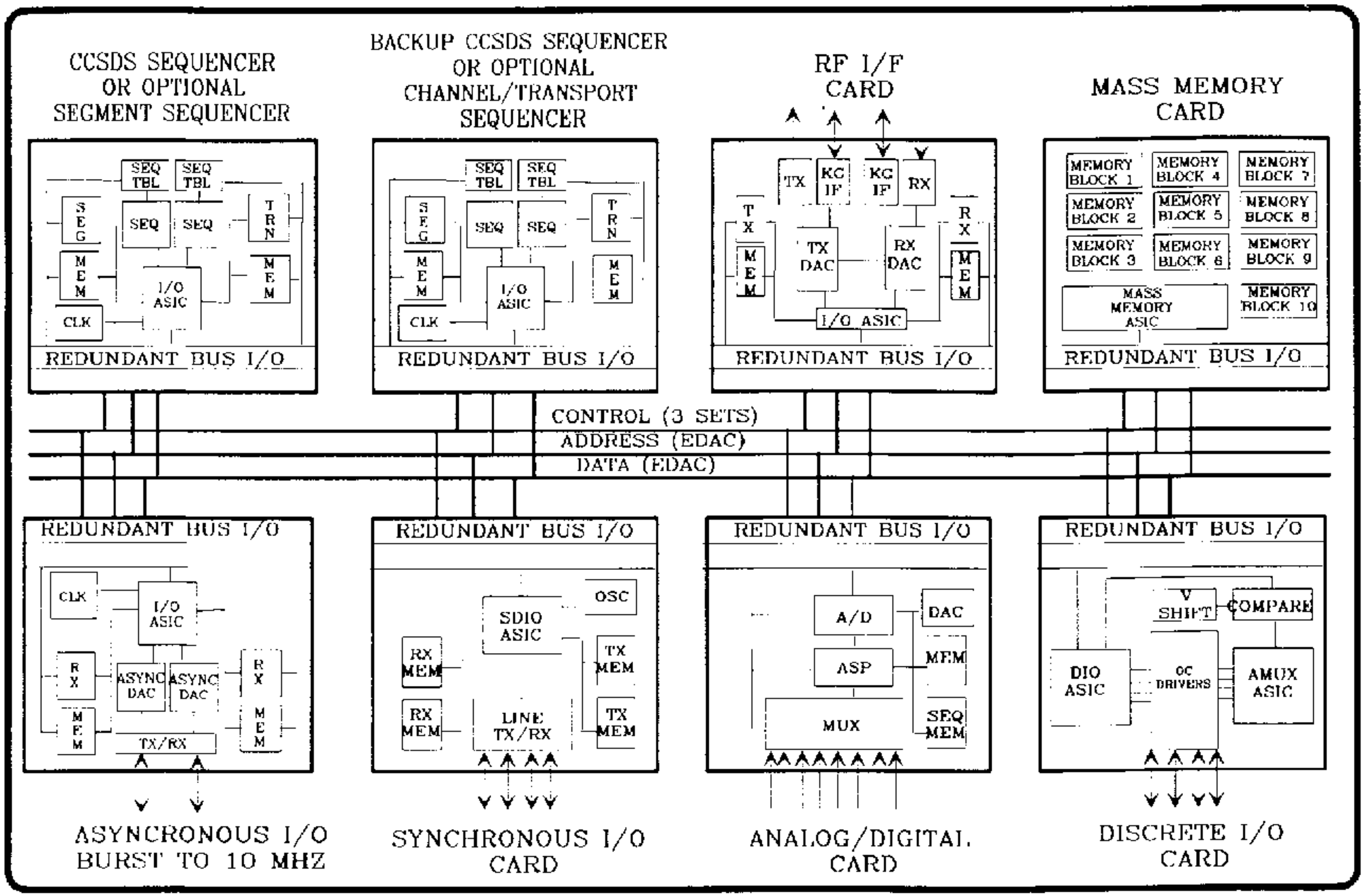


FIGURE 2 - CCSDS ARCHITECTURE