

APPENDIX A

DIGITAL DATA ACQUISITION AND ON-BOARD RECORDING: SUMMARY OF CONCEPT AND RATIONALE

This appendix presents the conceptual framework and rationale for the RCC Digital Data Acquisition and On-Board Recording System. The background information provided here will be found helpful in understanding the RCC Digital Data Acquisition and On-Board Recording Standard. Through the process of normal evolution, it is expected that expansion, deletion or modification to this report may occur. Questions relative to the contents or status of this report should be addressed to the RCC Secretariat.

TABLE OF CONTENTS

<u>Sections</u>	<u>Page</u>
1. APPENDIX A: PURPOSE, SCOPE AND ORGANIZATION	A-4
1.1 PURPOSE	A-4
1.2 SCOPE	A-4
1.3 ORGANIZATION.....	A-5
2. OVERVIEW OF DIGITAL DATA ACQUISITION AND ON-BOARD RECORDING SYSTEM.....	A-6
2.1 INTRODUCTION.....	A-6
2.2 DIGITAL DATA ACQUISITION AND ON-BOARD RECORDING SYSTEM CONCEPT	A-7
2.2.1 Packetization Layer	A-7
2.2.2 Segmentation Layer [No Longer Used in CCSDS].....	A-8
2.2.3 Transfer Frame Layer	A-8
2.2.4 Channel Coding Layer [Not Specified By This Standard Except As Contained In Appendix B]	A-8
2.2.5 Relationship Between Telemetry and Telecommand Systems [Not Used By This Standard].....	A-9
3. DIGITAL DATA ACQUISITION AND ON-BOARD RECORDING SYSTEM DESCRIPTION AND RATIONALE.....	A-10
3.1 PACKET RECORDING.....	A-10
3.1.1 Introduction	A-10
3.1.2 Source Packet	A-11
3.1.3 Flow Control Mechanisms	A-14
3.1.4 Transfer Frame	A-15
3.2 TELEMETRY CHANNEL CODING [Not Covered by This Standard Except As Contained In Appendix B].....	A-19
 <u>Annexes</u>	
A-1 GLOSSARY OF TERMINOLOGY	A-20
A-2 "APPLICATION NOTES" FOR DIGITAL DATA ACQUISITION AND ON-BOARD RECORDING STANDARD.....	A-21
A-3 RECORDING TRANSFER FRAME ERROR DETECTION ENCODING/DECODING GUIDELINE.....	A-31

Figures

2-1	Layered digital data acquisition and on-board recording service model.	A-8
3-1	Digital data acquisition and on-board recording system data flow.	A-11
3-2	Source packet (version 1) format.	A-12
3-3	Transfer frame format.	A-16
A3-1	Encoder.	A-33
A3-2	Decoder.	A-33

APPENDIX A

1. PURPOSE, SCOPE AND ORGANIZATION

This report contains the concept and supporting rationale for the Digital Data Acquisition and On-Board Recording System developed by the Telemetry Group (TG) of the Range Commanders Council (RCC). It has been prepared to provide an introduction and overview of the system concept upon which the detailed RCC Digital Data Acquisition and On-Board Recording Standard is based. The Digital Data Acquisition and On-Board Recording Standard is based on the Consultative Committee for Space Data Systems (CCSDS) Packet Telemetry Recommendation (reference [1]). (The concept and supporting rationale for the CCSDS Packet Telemetry Recommendation is contained in reference [4]).

Currently there are a large number of unique data structures that have been developed by vendors and government for specific on-board data recording applications. These unique data structures require unique decoding software programs. Writing unique decoding software, checking the software for accuracy, and decoding the data tapes is extremely time consuming and costly.

A need was identified to develop a digital data acquisition and on-board recording standard that will support the multiplexing of multiple data streams and maintain the accuracy of data correlation with time. Specifically, the Digital Data Acquisition and On-Board Recording Standard should be compatible with the multiplexing of both synchronous and asynchronous digital inputs such as Pulse Code Modulation (PCM), MIL-STD-1553 asynchronous data bus, digital voice, time, discrete, video, and RS-232/422 communication data. In addition, the Digital Data Acquisition and On-Board Recording Standard should be aligned with current developments in layered communications architecture.

The digital data acquisition standard should allow the use of a common set of playback/data reduction software, take advantage of emerging random access recording media, improve efficiency over traditional PCM for asynchronous data, and take advantage of the rapid improvement in commercial communication technology.

1.2 SCOPE

The concepts, protocols and data formats developed for the CCSDS Telemetry System and adopted for the RCC Digital Data Acquisition and On-Board Recording System described herein are designed for flight and ground data systems supporting conventional, contemporary, instrumented vehicles. Data formats are designed with efficiency as a primary consideration, i.e., format overhead is minimized. (A report on the analysis of candidate data structures as a possible Digital Data Acquisition and On-Board Recording Standard is contained in reference [5]).

The CCSDS Packet Telemetry Recommendation is primarily intended to support multi-point to multi-point data transfer over space based transmission links. In contrast, the RCC Digital Data Acquisition and On-Board Recording Standard is intended to support point-to-point data acquisition and recording and subsequent play back on a ground-based data acquisition system. While there are significant similarities, the differences require some minor deviations from the absolute adoption of the CCSDS Packet Telemetry Recommendation. It is therefore the intent of the RCC Digital Data Acquisition and On-Board Recording Standard to duplicate the CCSDS Packet Telemetry Recommendation and deviate only when unique on-board recording/ground playback or point-to-point considerations warrant.

1.3 ORGANIZATION

An overview of the RCC Digital Data Acquisition and On-Board Recording System is presented in section 2. This discussion introduces the notion of architectural layering to achieve transparent and reliable delivery of scientific and engineering sensor data (generated aboard remote vehicles) to an on-board recording device and subsequently played back through ground data systems to a user on the ground. Section 3 presents a more detailed description of the Digital Data Acquisition and On-Board Recording System and rationale for the specific RCC Digital Data Acquisition and On-Board Recording Standard.

Annex A-1 provides a glossary to familiarize the reader with the terminology used throughout the RCC Digital Data Acquisition and On-Board Recording System. Annex A-2 contains application notes that describe how a project may implement complete or partial compatibility with the RCC Digital Data Acquisition and On-Board Recording Standard. Annex A-3 is a guideline for transfer frame error detection coding.

2. OVERVIEW OF THE RCC DIGITAL DATA ACQUISITION AND ON-BOARD RECORDING SYSTEM

2.1 INTRODUCTION

The purpose of a Digital Data Acquisition and On-Board Recording System is to reliably and transparently convey measurement information from a data-generating source to a recording device and then to playback the data to user(s) located on the ground. Typically, data generators are scientific sensors, science housekeeping sensors, engineering sensors and other subsystems on-board a vehicle.

The advent of capable microprocessor based hardware will result in data systems with demands for greater throughput and a requirement for corresponding increases in vehicle autonomy and mission complexity. These facts, along with the current technical and fiscal environments, create a need for greater Digital Data Acquisition and On-Board Recording capability and efficiency with reduced costs.

The lack of effective standardization among various missions and ranges forces "multi-mission/multi-range" data acquisition systems to require significant set-up costs/time to handle the variety of unique on-board data recording formats. Higher level data handling services oriented more toward computer-to-computer transfers and typical of modern day commercial and military communication networks must be custom designed and implemented on a mission by mission basis. The intent of the Digital Data Acquisition and On-Board Recording System is not only to ease the transition toward greater automation within organizations, but also to promote standardization among the organizations, thereby resulting in greater cross-support opportunities and services.

Digital Data Acquisition and On-Board Recording: packet telemetry is a concept which facilitates the on-board recording of remote vehicle-acquired data from source to on-board recorder and then playback to the user via a ground data acquisition system in a standardized and highly automated manner. Packet telemetry provides a mechanism for implementing common data structures and protocols that can enhance the development and operation of range missions. Packet telemetry for Digital Data Acquisition and On-Board Recording addresses the process of the end-to-end transport of DOD mission data sets from source application processes located on a vehicle to an on-board recording device, and then played back from the recording device to user application processes located on the ground.

The Digital Data Acquisition and On-Board Recording Standard is primarily concerned with describing the on-board recording formats which are generated by remote vehicles in order to execute their roles in the above processes.

Packet telemetry services provide the user with reliable and transparent recording and playback of remote vehicle digital data.

2.2 DIGITAL DATA ACQUISITION AND ON-BOARD RECORDING SYSTEM CONCEPT

The system design technique known as layering was found to be a very useful tool for transforming the Digital Data Acquisition and On-Board Recording System concept into sets of operational and formatting procedures. The layering approach is patterned after the International Organization for Standardization's Open Systems Interconnection Reference Model for networking (reference [6]), which is a seven layered architecture that groups functions logically and provides conventions for passing information from layer to layer. Layering decomposes a complex procedure into sets of peer functions residing in common architectural strata with standardized interfaces between them.

Within each layer, the functions exchange data according to established standard rules or "protocols." Each layer draws upon a well-defined set of services provided by the layer below, and provides a similarly well defined set of services to the layer above. As long as these service interfaces are preserved, the internal operations within a layer are unconstrained and transparent to other layers. Therefore, an entire layer within a system may be removed and replaced as dictated by user or technological requirements without destroying the integrity of the rest of the system. Further, as long as the appropriate interface protocol is satisfied, a customer (user) can interact with the system/service at any of the component layers. Layering is therefore a powerful tool for designing structured systems that change due to the evolution of requirements or technology.

A companion standardization technique that is conceptually simple, yet very robust, is the encapsulation of data within an envelope or "header." The header contains the identifying information needed by the layer to provide its service while maintaining the integrity of the envelope contents.

Figure 2-1 illustrates the RCC Digital Data Acquisition and On-Board Recording System in terms of a layered service model. It should be noted that the RCC Digital Data Acquisition and On-Board Recording Standard only addresses two of the layers (packetization and transfer) of this model. ("Application Notes" presented in Annex B detail implementation guidelines for compatibility with the layered services at the interface between application data and the packetization layer and between transfer frames and the coding layer).

Figure 2-1 is modified from the CCSDS Packet Telemetry Recommendation by showing a recording device as the "recipient" of the physical layer waveform. It also depicts applicability to only the packetization and transfer layers.

2.2.1 Packetization Layer.

Within the Digital Data Acquisition and On-Board Recording Standard, packet telemetry vehicle-generated application data are formatted into end-to-end transportable data units called source packets. These data are encapsulated within a primary header that contains identification, sequence control and packet length information. A source packet is the basic data unit recorded

on-board the vehicle and generally contains a meaningful quantity of related measurements from a particular source.

2.2.2. Segmentation Layer.

[NO LONGER USED - PER ISSUE 4, CCSDS PACKET TELEMETRY RECOMMENDATION]

2.2.3 Transfer Frame Layer. The transfer frame is used to reliably multiplex source packets onto the on-board recording device. As the heart of the RCC Digital Data Acquisition and On-Board Recording System, the transfer frame protocols offer a range of delivery service options. An example of such a service option is the multiplexing of transfer frames into virtual channels (VCs).

The transfer frame begins with an attached frame synchronization marker and is followed by a primary header. The primary header contains frame identification, channel frame count information, and frame data field status information.

The transfer frame data field is followed by a frame error control field. The frame error control field provides the capability for detecting errors that may have been introduced into the frame during the data handling process. The delivery of transfer frames requires the services provided by the lower layers (e.g., coding/decoding) to accomplish its role.

2.2.4 Channel Coding Layer.

The channel coding layer is not specified in the RCC Digital Data Acquisition and On-Board Recording Standard although a CCSDS Channel Coding Recommendation, reference [2], is included in the CCSDS Packet Telemetry Recommendation. Data errors and/or loss on recording are significantly less than experienced in over-the-air transmission of telemetry signals. Therefore, the specific coding implementation is left to the discretion of the implement depending on the characteristics of the physical recording medium. It is recommended that channel coding for purposes of error detection and/or correction be integrated into the recording device and be transparently removed during playback. Appendix B contains those portions of the CCSDS Channel Coding Recommendation that are applicable to the Digital Data Acquisition and On-Board Recording Standard. (Future versions of the RCC Digital Data Acquisition and On-Board Recording Standard may specify a coding layer standard).

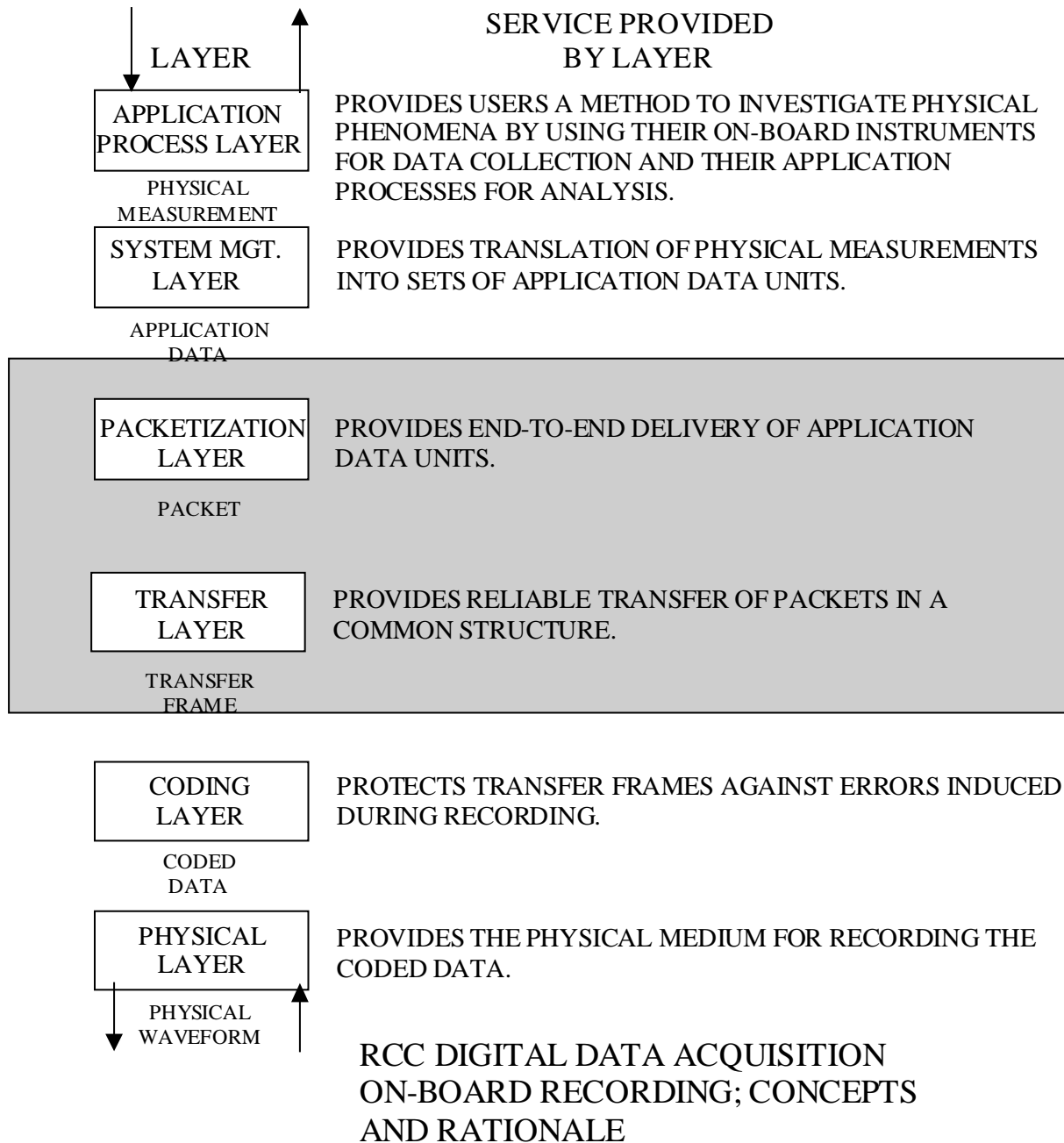


Figure 2-1: Layered digital data acquisition and on-board recording service model.

2.2.5 Relationship Between Telemetry and Telecommand Systems.

The RCC Digital Data Acquisition and On-Board Recording Standard is for one-way communication from the vehicle to the on-board recording device and from the playback device to the ground data acquisition system. There are no provisions for telecommands (uplink).

3. DIGITAL DATA ACQUISITION AND ON-BOARD RECORDING SYSTEM DESCRIPTION AND RATIONALE

This section describes the services and protocols characterizing the Digital Data Acquisition and On-Board Recording System and presents the rationale for detailed structure of the data units. The section discusses the two main protocol and format areas: 1) source packet and 2) transfer frame.

3.1 PACKET RECORDING

3.1.1 Introduction. Packet telemetry represents an evolutionary step from the traditional Time-Division Multiplex (TDM) method of acquiring, recording, and playing back scientific applications and engineering data from instrumented vehicle sources to ground data systems sinks. It also relies on a layered architectural model to isolate independent interfaces. The packet telemetry process has the conceptual attributes of

- (1) Facilitating the acquisition, transmission, and recording of instrument data at a rate appropriate for the phenomenon being observed.
- (2) Defining a logical interface and protocol between an instrument and its associated on-board recorder/playback and ground support equipment which remains constant throughout the life cycle of the instrument (bench test, integration, flight, and possible re-use).
- (3) Simplifying overall system design by allowing microprocessor-based symmetric design of the instrument control and data paths ("transfer frames in," transfer frames out") compatible with commercially available components and interconnection protocol standards.
- (4) Facilitating interoperability of instrumented vehicle systems whose data acquisition and on-board recording systems interfaces conform to CCSDS/IRIG guidelines.
- (5) Enabling the delivery of high-quality data products in a mode that is faster and less expensive than would be possible with conventional methods.

Figure 3-1 is a functional diagram of the Digital Data Acquisition and On-Board Recording System data flow from the creation of a data set by an application process operating within a vehicle "source" (instrument or subsystem), through to the delivery of the same data to a user "sink" (application process) on the ground. Since many of the elements of this flow are presently mission-unique, a primary objective of packet telemetry is to define stable, mission-independent interface standards for the communications path within the flow.

Figure 3-1 is modified from the CCSDS Packet Telemetry Recommendation by showing the transfer frames recorded onto a physical medium and then played back from the physical medium, instead of the over-the-air transmission of the coded transfer frames.

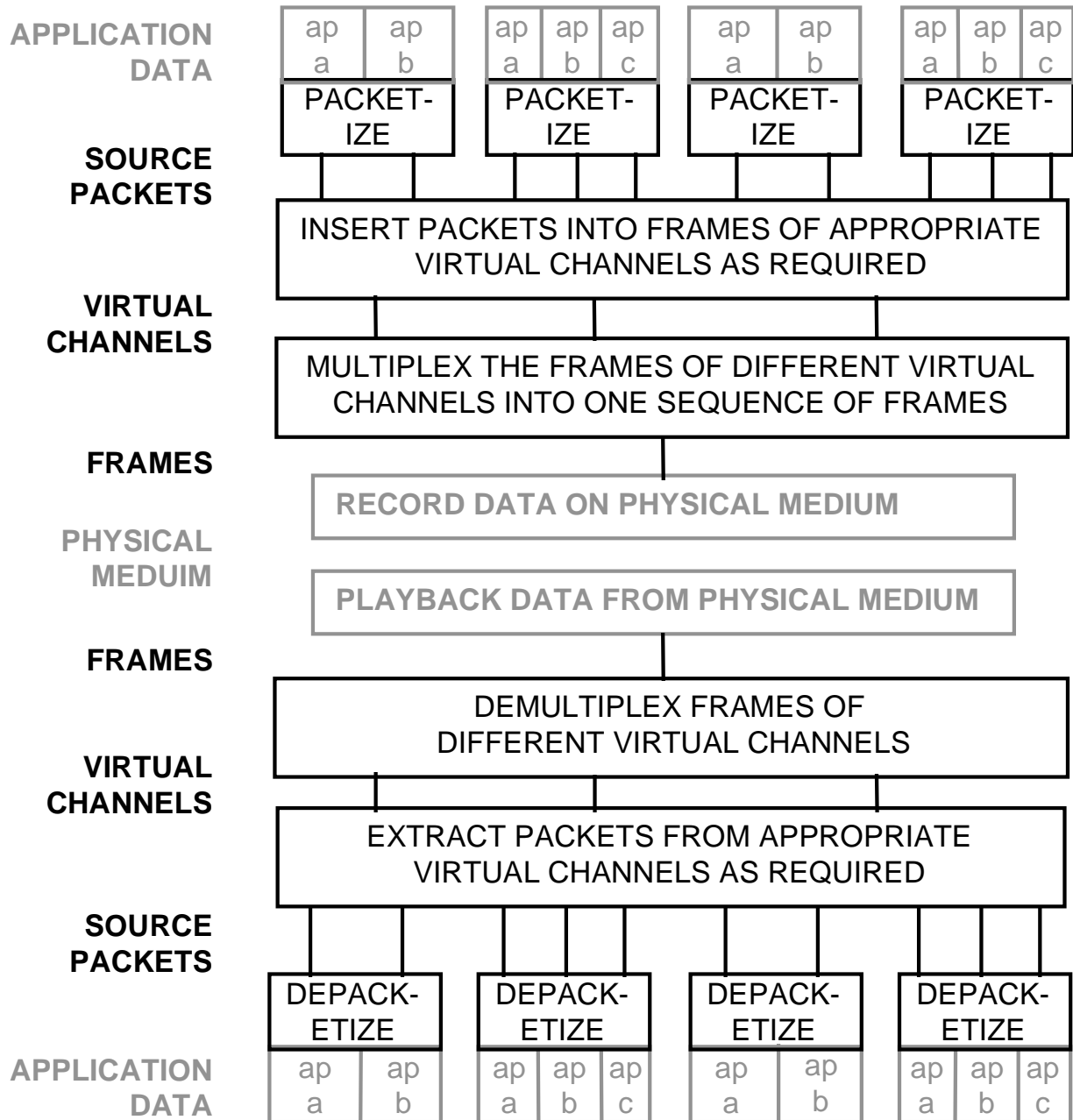


Figure 3-1. Digital data acquisition and on-board recording system data flow.

3.1.2 SOURCE PACKET. A source packet is a data unit encapsulating a block of observational data that may include ancillary data and may be directly interpreted by the receiving end application process. The source packet format (version 1), with the addition of a secondary header and packet error control field, is reproduced in Figure 3-2 below for the convenience of the reader.

The source packet data structure of Figure 3-2 is identical to the source packet data structure of the CCSDS Packet Telemetry Recommendation (reference [1]).

User application data are encapsulated within a packet by prefacing them with a standard label or "primary header," which is used by the data transport system to route the data through the system and to allow the user to reconstruct the original data set. The primary header consists of three main fields: packet identification, packet sequence control and packet length.

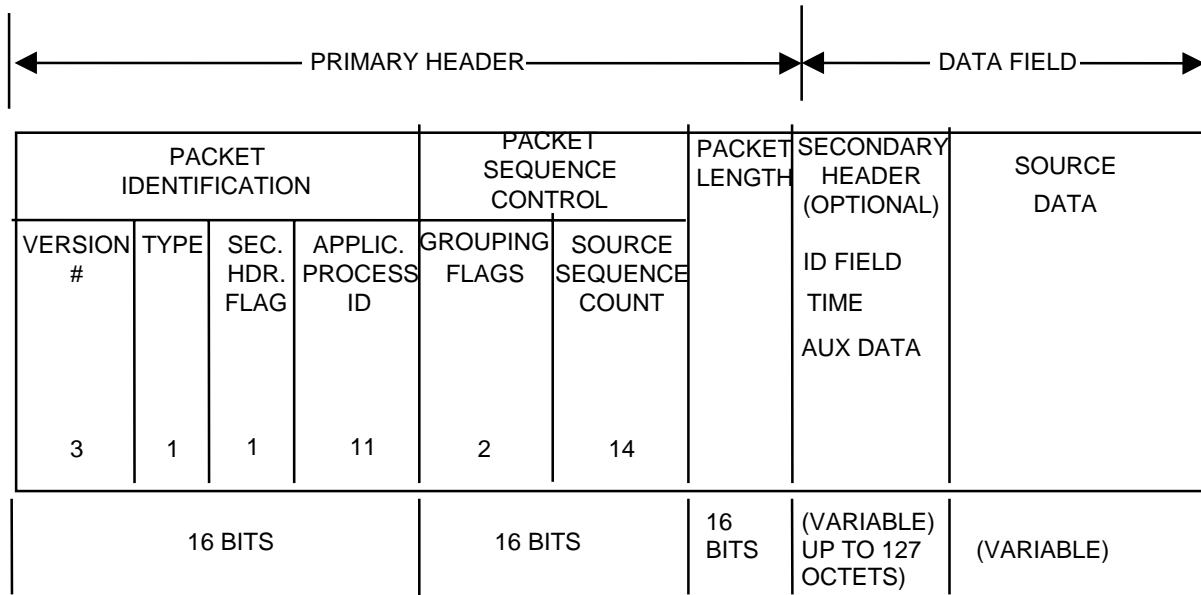


Figure 3-2: Source packet (version 1) format.

3.1.2.1. Packet Identification.

Version Number. The version number is the first of four sub-fields of packet identification. This sub-field explicitly indicates the version of the formatted packet, and its length of three bits allows eight different versions to be identified. While only two versions are currently defined, this arrangement allows a reasonable growth capability to support future needs. However, in the interest of constraining the proliferation of standards, additional versions will be discouraged unless it can be demonstrated that the current versions are truly inadequate.

Type. The second sub-field is a one-bit identifier to signal that this packet is a "Telemetry" packet and not a "telecommand" packet. It is always set to "zero" for telemetry packets.¹

¹In the first issue of reference [1] (May 1984) this field was described as a "reserved spare" and was, by convention, set to zero for telemetry. In issue 2 (January 1987), the value of the field has not changed, but its function has been established.

As there is no “telecommand” packet identified in the RCC Digital Data Acquisition and On-Board Recording Standard, the type is always set to “zero.”

Secondary Header Flag. The third sub-field is a one-bit secondary header flag. The RCC recognizes that users may need a means of encapsulating ancillary data (such as time, internal data field format, vehicle position/attitude, etc.) which may be necessary for the interpretation of the information contained within the packet. Therefore, this flag, when set to one, indicates that a secondary header follows the primary header.

Application Process ID. The last sub-field in the packet identification field is used to uniquely identify the originating source packet application process. Eleven bits are allocated to the Application Process ID, permitting identification of up to 2048 separate application processes per vehicle, sufficient for any envisioned mission. For positive identification, one can consider this sub-field an extension of the vehicle ID, which is in the transfer frame primary header (see Figure 3-4). The meaning of the application process ID shall be static for the complete mission phase.

3.1.2.2 Packet Sequence Control.

Grouping Flags. The first sub-field of the packet sequence control field is called "grouping flags," and provides for a logical representation of four types of grouping status. Grouping is no longer used per issue 4 of the CCSDS Packet Telemetry Recommendation; however, the 2 bits for the grouping flags remain and are always set to “11” to signify no grouping.

Source Sequence Count. This second sub-field calls for each packet to be numbered in a sequential manner, thus providing a method of checking the order of source application data at the receiving end of the system. It is normally used for accounting purposes to measure the quantity, continuity, and completeness of the data received from the source. The field provides a straight sequential count to modulo 16,384. Longer-term unambiguous ordering (beyond 16,384 packets) may be accomplished by associating the measurement time code contained within the packet with the source sequence count.

3.1.2.3 Packet Length. The last major field of the primary header delimits the boundaries of the packet. It is a count of the number of octets in the packet beginning with the first octet after the 48-bit primary header and ending with the last octet of the packet. The 16-bit field allows packet lengths up to 65,536 octets (not counting the 48-bit primary header). This packet limit was a compromise between the majority of users who produce medium-size packets and the few users who may produce exceptionally long packets. Placing a reasonable limit on packet size helps avoid the flow control problems associated with very long packets, and eliminates the overhead penalty of a larger length field for the great majority of packet producers.

3.1.2.4 Data Field. The remainder of the packet may consist of any data desired, although some suggestions are provided in the application notes, annex B. The total length of all subsequent data should be an even number of octets (a multiple of 16 bits) for efficiency in computer

processing. In addition, Figure 3-2 indicates two possible sub-fields: secondary header and source application data.

Secondary Header. A secondary header may be desirable for providing ancillary data generated by the application process (time, vehicle position/attitude). A secondary header ID field is the first field in the secondary header. The ID field contains two sub-fields. A time code present sub-field provides a bit to indicate the presence of a time code format. A secondary header length sub-field provides the length of the secondary header. If time is included in the secondary header, it must consist of a time code preamble field (P- field), a time specification field (T field), and adhere to either a day or calendar binary segmented or unsegmented binary Time Code Format specified in Appendix C of this standard. The ASCII time code formats are not allowed due to the absence of a preamble identifier for ASCII codes.

Source Data. Following the secondary header, the source data sub-field contains source application data generated by the application process identified in the primary header. For efficiency in computer processing, this sub-field should be a multiple of 16 bits.

Packet Error Control. NOT USED

3.1.3 Flow Control Mechanisms. Telecommunications systems are usually constrained by the capacity or the bandwidth of the telecommunications channel/on-board recording device. Flow control becomes crucial when multiple application processes must share the same telecommunications channel if significantly high data rate data is to be recorded. The Digital Data Acquisition and On-Board Recording System must ensure that all sources have proper access to this common resource frequently enough to ensure timely recording as well as to control the need to buffer data while other sources are being serviced. Long source packets may present flow control problems if they monopolize the data channel for unacceptable periods of time while forcing other sources to implement unreasonably large buffering of their data. Several alternative solutions to the problem of flow control are presented below, and are summarized in Annex A-3 of this report.

3.1.3.1 Virtual Channelization. One solution to the flow control problem is to assign each source (which generates long packets) its own virtual channel. This is accomplished by inserting these packets into specially identified transfer frames. These dedicated frames form a virtual channel and may be interleaved with other frames containing data from other applications. Detailed discussion of virtual channelization occurs in section 3.1.4.

3.1.3.2 Source-Internal Segmentation: Source Packet (Version 1). [No longer used per issue 4 of CCSDS Packet Telemetry Recommendation]

3.1.3.3 Spacecraft Segmentation: Source Packet (Version 1). [No longer used per issue 4 of CCSDS Packet Telemetry Recommendation]

3.1.3.4 Spacecraft Segmentation: Telemetry Segment (Version 2). [No longer used per issue 4 of CCSDS Packet Telemetry Recommendation]

3.1.4 Transfer Frame. The source packet data structures described in the previous sections are embedded within a data transfer structure, the transfer frame, which provides reliable, error-controlled transfer to and onto the recording media. The transfer frame has a fixed length for a given mission or vehicle. The attributes of the transfer frame and its supporting rationale are presented in the discussion of the transfer frame format. Figure 3-3 illustrates the transfer frame format.

The transfer frame data structure of Figure 3-3 is identical to the transfer frame data structure of the CCSDS Packet Telemetry Recommendation (reference [1]).

3.1.4.1 Synchronization Marker. Attached to the beginning of the transfer frame primary header is a 32-bit frame synchronization marker used by the ground station to acquire synchronization with the frame boundaries during playback. The particular bit pattern is found in Appendix B of this standard. All transfer frames in a single physical data channel in a given mission must be of constant length.

3.1.4.2 Frame Identification. The first major field of the transfer frame primary header is the frame identification field.

Version Number. Only one version of the transfer frame has been defined, although this 2-bit field allows growth to four. The "version" refers to the frame structuring principles that are described in this section.

Vehicle ID. The vehicle identification field provides for positive identification of the vehicle that generated the transfer frame. The 10 bits assigned to vehicle identification allows up to 1024 separate positive IDs.

The CCSDS Packet Telemetry Recommendation defines this field as "spacecraft ID." The CCSDS Packet Telemetry Recommendation also requires the CCSDS Secretariat to assign spacecraft identifiers. This standard changes the name to "vehicle ID" and leaves the authority for assigning vehicle IDs to the individual projects/ranges.

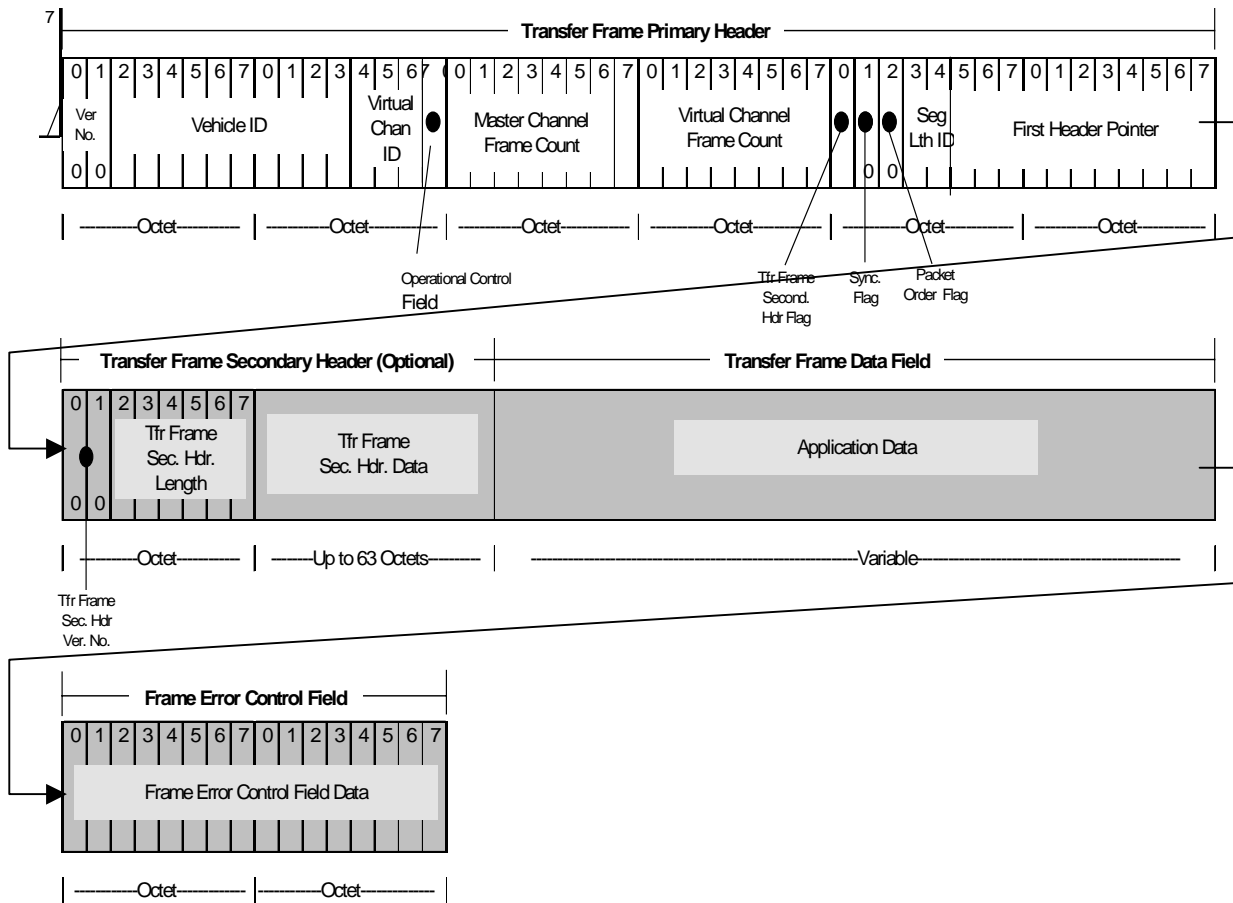


Figure 3-3. Transfer frame format.

Virtual Channel ID. This three-bit sub-field allows up to eight virtual channels to be run concurrently on a particular physical data channel. Frames from different virtual channels are multiplexed together on the recording medium, and, with this identifier in each frame, can be easily split apart during playback on the ground. Virtual channels can be used for a variety of purposes such as flow control to prevent long packets from "hogging" the channel; selecting out different types of data for stream splitting (e.g., when low-rate engineering data must be split out from multiplexed high-rate science data) or when different levels of data quality are to be accommodated for different types of data (in which case error protection may be applied to certain virtual channels but not others). Eight virtual channels are considered sufficient to provide adequate flexibility for envisioned future vehicles.

Operational Control Field Flag. The last bit of the frame identification field, when set to "one," signals the presence of the 32-bit operational control field that is contained within the frame trailer. The RCC Digital Data Acquisition and On-Board Recording Standard does not specify the use of the operational control field, and the operational control field flag is always set to "zero."

For the CCSDS Packet Telemetry Recommendation, the information in this field is defined to provide a standardized spacecraft reporting mechanism for spacecraft telecommanding (uplink) which is not used in this standard.

3.1.4.3 Master Channel and Virtual Channel Frame Count. The next two fields provide a running count of the number of frames transmitted. These counters provide a degree of data accountability (for short duration data loss), the ambiguity level being defined by the field lengths.

Master Channel Frame Count. This 8-bit field provides sequential count (modulo 256) of the number of frames transmitted by a single physical data channel. The counter is long enough to provide a reasonable probability of detecting a discontinuity, in a sequence of frames, when the physical channel is briefly interrupted. If such a discontinuity does occur, the virtual channel accounting process can provide a greater probability of detecting the number of missing frames.

Virtual Channel Frame Count. The following 8-bit field provides accountability for each of the eight independent virtual channels. This field is used with the virtual channel ID sub-field to provide accountability via a sequential count (modulo 256). The rationale for the counter ambiguity level is the same as for the master channel frame counter. If only one virtual channel is incorporated for a given mission, both the virtual channel frame counter and the master channel frame counter must increment once per generated transfer frame (i.e., the two fields should not be concatenated into a master frame counter).

3.1.4.4 Frame Data Field Status. The frame data field status provides control information that allows the recorder playback end to extract and reconstitute packets and/or segments.

Secondary Header Flag. The first sub-field indicates the presence or absence of the optional secondary header. If its presence is so indicated, the secondary header must appear in every frame transmitted through a physical data channel, and its length must also be fixed. The rationale for this requirement is provided later in the discussion (section 3.1.4.5) about the secondary header.

Synchronization Flag. This flag indicates whether or not the packet or segment data units are inserted into the transfer frame data field on octet boundaries. If they are, then they are said to be "synchronously inserted" (packet octet boundaries align with frame octet boundaries) and the extraction technique (pointing to specific octet) is valid. If the flag indicates "asynchronous" data insertion (i.e., unstructured (non-packetized) data contents or packets inserted without regard to octet boundaries), the transfer frame layer at the recorder playback end will not be able to reconstitute the original data sets without additional knowledge.

Packet Order Flag. This flag indicates whether the sequence count order of the contained packet is increasing (forward) or decreasing (reverse). This has important implications when tape recorded data are played back opposite to their recorded direction. When this is the case, the vehicle electronics re-justifies the bit direction of each packet so each packet individually flows in the forward direction and its header can be read to allow proper packet extraction from the

transfer frame. Even though the playback packets appear individually to flow the same as the rest of the data, the sequence of packets will be running backwards in time as indicated by the decreasing sequence counter. A discussion of various options for handling tape recorded data is contained in annex B.

Segment Length ID. [No longer used per, CCSDS Packet Telemetry Recommendation, issue 4, November 1995.]

First Header Pointer. The first header pointer sub-field points directly to the location of the starting octet of the first packet header structure within the frame data field. It counts from the end of the primary header (secondary header if present) and effectively delimits the beginning of the first packet. The packet length field, in turn, delimits the beginning of the next packet, and so on. Since the pointer counts octets, this feature works only when the headers are aligned with octet boundaries, i.e., when the packet data are synchronously inserted (data field synchronization flag set to zero). The eleven bits allocated to the pointer allow for a count to 2048 octets, which exceeds the count required to point to an octet at the end of the data field. Two special pointer values are reserved. These denote:

- (1) No packet header is contained in this frame, but there is valid data, or
- (2) No valid data is contained in this frame (idle channel).

3.1.4.5 Frame Secondary Header (Optional). An optional secondary header is provided for users who desire a means for inserting time or for deterministically inserting real-time data (e.g., Time-Division-Multiplexed data). When the secondary header presence is indicated by the secondary header flag, its length must be of a fixed value and must appear in every frame transmitted through a physical channel. Given the requirement for fixed transfer frame length, a fixed secondary header length simplifies data processing and packet extraction at the recording playback end.

Secondary Header ID. The first part of the secondary header has two sub-fields. The first is the secondary header version number, a 2-bit field allowing four versions (or structuring rules). Only version 1 (represented by 00) is currently defined. This provides for a reasonable future growth capability.

The second sub-field, secondary header length, indicates what length has been selected for the secondary header. This 6-bit sub-field provides a binary count of the total number of octets contained within the entire transfer frame secondary header (including the ID field itself, which is one octet in length). This limits the total secondary header length to 64 octets (512 bits) which is considered adequate for currently understood applications.

Secondary Header Data. This sub-field contains up to 504 bits of user specified data. Specifically, if time is to be included in the secondary header data field, it must conform to the Time Code Formats of appendix C.

3.1.4.6 Transfer Frame Data Field. The transfer frame data field contains an integral number of octets of data (source packets) to be recorded by the on-board recording device. The maximum length of this field depends on which optional fields are implemented.

3.1.4.7 Transfer Frame Trailer. A transfer frame trailer provides for frame error detection.

Operational Control Field. NOT USED IN THIS STANDARD.

The CCSDS Packet Telemetry Recommendation uses the Operational Control Field to uplink Telecommand information.

Frame Error Control Field. This field occupies the two trailing octets of the transfer frame. Its presence is mandatory and must appear in all transfer frames. It provides the capability for detecting errors that have been introduced into the frame during the data handling processes.

A Cyclic Redundancy Code (CRC) has been selected for this purpose because of its effectiveness and simplicity, and is defined and specified in section 5.5.1 of this standard. Parity is generated over the entire transfer frame (less the final 16 bits), and the 16 bits of parity checks are then appended to complete the frame. It should be noted that in the 1984 issue of the CCSDS Packet Telemetry Recommendation, the frame was defined to include the "attached sync marker." In the 1987 issue, the frame definition was changed to exclude the marker, but it was still considered to be "attached." To maintain compatibility with already-built systems, it was necessary to allow for two options over which the CRC is applied, that is; it may include the sync marker or it may exclude it. Since the marker pattern is always known, the preferred choice is to omit the marker when encoding. If the frame error control field is not utilized, it is recommended to fill the field with all "ones" or "zeroes."

3.2 TELEMETRY CHANNEL CODING

The channel coding layer is not specified in this standard.

The CCSDS Packet Telemetry Recommendation includes a Channel Coding Recommendation as an integral part of the recommendation. Because of the over-the-air transmission issues, the Channel Coding Recommendation is appropriate. However, for this standard, data errors and/or loss on recording are significantly less than experienced in over-the-air transmission of telemetry signals. Therefore, the specific coding implementation is left to the discretion of the implementor depending on the characteristics of the physical recording medium. (Future versions of this standard may specify a coding layer standard).

ANNEX A-1

GLOSSARY OF TERMINOLOGY

fill bit(s) - Additional bit(s) appended to enable a "data entity" to exactly fit an integer number of octets.

octet - An 8-bit word consisting of eight contiguous bits.

packet - An efficient application-oriented protocol data unit that facilitates the transfer of source data to users located in space or on Earth.

physical channel - The single bit stream that is recorded onto the physical media. It includes all multiplexed transfer frames as well as any implementor unique coding algorithms.

protocol - A set of procedures and their enabling format conventions that define the orderly exchange of information between entities within a given layer of the TM System.

reliable - Meets the quality, quantity, continuity, and completeness criteria that are specified by the TM System.

Digital Data Acquisition and On-Board Recording System - The end-to-end system of layered data handling services that exist to enable a vehicle to send measurement information, in an error-controlled environment, to on-board recorders and then played back through data processing equipment on Earth.

transfer frame - A communication-oriented protocol data unit that facilitates the transfer of application-oriented protocol data units through the space-to-ground link.

user - A human or machine-intelligent process which directs and analyzes the progress of a mission.

virtual channel - A given sequence of transfer frames, which are assigned a common identification code (in the transfer frame header), enabling all transfer frames that are members of that sequence to be uniquely identified. It allows a technique for multiple source application processes to share the finite capacity of the physical channel (i.e., through multiplexing).

ANNEX A-2

APPLICATION NOTES

Purpose:

The Digital Data Acquisition and On-Board Recording Standard provides extensive flexibility in formatting a wide variety of data types for the purpose of recording them. While the basic formats are standardized, the implementor still must make choices when using this standard to format data for recording.

The use of this standard alone does not imply interoperability between the processes that format the data for recording and those that subsequently read the recording and interpret the data.

These application notes have been developed to ensure that a process can be implemented to extract application data from the Digital Data Acquisition and On-Board Recording Format and to interpret it as it was intended to be interpreted. The specific purpose of the application notes, therefore, is to guide the implementors in making commonly accepted choices when applying Digital Data Acquisition and On-Board Recording Formats to record the most frequently recorded types of data.

Status:

These application notes are subject to periodic revision by the RCC Telemetry Group and are to be considered a work in progress.

The goal of these reviews is to eventually achieve a single application layer standard for each data type instead of the dual standards that now exist for some data types.

Annex Organization:

The application notes are organized into sections each of which describes guidelines related to a particular type of application data stream presented to the entity which implements this standard. The first section gives general guidelines, while the remaining sections contain guidelines of particular relevance to specific types of data to be recorded or particular layer processes.

1. GENERAL GUIDELINES

The following guidelines may be of interest to implementors of the standard irrespective of the type of data being recorded.

1.1. SELECTION OF TRANSFER FRAME LENGTH

The standard specifies that the length of a transfer frame is fixed for a mission. When the recording capacity during a mission is of concern, the transfer frame length should be made as large as possible to reduce the amount of recording space used for transfer frame primary header and transfer frame secondary header octets. In contrast, to meet operational requirements, the maximum transfer frame length may be limited by multiplexing needs, data criticality, or buffer size.

The maximum length of a transfer frame is limited by appendix B to 8920 bits.

1.2 TRANSFER FRAME BUFFERING CONSIDERATIONS

For channels where data may not arrive for long periods of time, and when there is no data to be recorded, a transfer frame filled with idle data may be forwarded to mark time on the virtual channel. This will allow the playback mechanism to find at least one transfer frame from each virtual channel within a short sequence on the recording medium. Section 1.3 of this annex provides guidelines for forwarding partially filled or empty transfer frames.

1.3 FORWARDING OF PARTIALLY FILLED TRANSFER FRAMES

An implementor may choose to have transfer frames forwarded before they are completely filled. For example, there may be a need to forward transfer frames at a constant periodic rate, whether or not there is sufficient data to fill them, to maintain synchrony with the recording equipment, to facilitate playback as noted in section 1.2 of this annex, or to pass user defined information contained in the transfer frame secondary header. Also, the implementor may want to forward the transfer frame at a time increment rollover even if the frame is not completely filled.

When there are not sufficient source packets to fill a transfer frame and a transfer frame must be forwarded on a particular virtual channel, there are two methods available. The first method is applicable if there is already a transfer frame under construction that is partially filled with one or more source packets. For this case, the transfer frame data field may be filled with one or more idle packets so as to completely fill the transfer frame and thereby cause it to be forwarded for recording. If more than 7 octets remain in the partially filled transfer frame, an idle packet should be inserted. If 7 or fewer octets remain in the partially filled transfer frame, fill the remaining bits in the transfer frame with idle data. A second method is applicable if no transfer frame is being filled with source packets when a transfer frame must be forwarded. In this case, it is left as an implementation option whether to forward a complete transfer frame containing only idle data or containing one or more idle packets that completely fill the transfer frame data field.

1.4 OCTET ALIGNMENT

All data placed into the source data field of a source packet should end on an octet boundary. If there are not sufficient application data bits in the source data field of a source packet to make up an integral number of octets, one to seven fill bits should be appended to the end of the application data to make the source data field an integral number of octets in length (see following sections for details on octet alignment options relative to specific application data types).

1.5 TREATMENT OF TIME VALUES

The time values used in this standard are associated with several different events. Care must be taken in how these time values are used. There are two major associations of time values: time associated with the application data, and time associated with the packaging of the application data using the formats (source packets and transfer frames) prescribed by this standard.

1.5.1 Time Associated with the Application Data. When there is a critical need for associating time with an application data stream, the time values should be contained in the application data stream itself and will be considered part of the application data stream. Time information which is embedded into the application data stream is considered independent of any time information which may be included in the source packet secondary header or the transfer frame secondary header (see section 1.5.2 of this annex). Time information that is embedded into the application data is not covered by this standard.

1.5.2 Time Associated with the Source Packet and Transfer Frames. This standard provides the optional capability to carry source packet time in the packet secondary header time code field (see paragraph 3.2.1.1 of this standard) and/or transfer frame time in the transfer frame secondary header data field (see paragraph 5.2.2 of this standard). When source packet time and/or transfer frame time is included

- a.) The source packet time and the transfer frame time are considered to be independent of one another.
- b.) The CCSDS Time Code Recommendation, Level 1, segmented or unsegmented binary time code formats, shall be used to represent time information (i.e., the time values shall be represented completely and unambiguously). (See reference [3] and appendix C.) ASCII time code formats are not allowed, because preamble identification codes do not exist.
- c.) The time code formats shall consist of a Preamble field (P-field) and a Time field (T-field).
- d.) The length of the time code field shall be defined by the implementor and resolution shall be indicated in the P-field. (See reference [3] and appendix C.)
- e.) The time value shall represent the arrival time of the first bit of data in the source packet data field and/or transfer frame data field.

1.6 SOURCE PACKET TIME

Where there is a requirement to time-tag application data with a high degree of accuracy, the implementor should include application data time as one input to the PCM frame in accordance with the format prescribed in Chapter 4 of reference [7]. The reason for requiring application data time to be embedded into the PCM data stream is that there may be a relatively large difference between the time of origin of the application data and the time at which the application data is finally placed into a source packet. The source packet time represents the time at which the source packet was created.

2.2 PACKETIZATION LAYER CONSIDERATIONS

The source packet is allowed to be of variable length depending on the needs of the application process(es). However, with synchronous data, it is considered prudent to use fixed length source packets. The choice of the length of the source packet may be selected freely upon consideration of latency, buffer size requirements, efficiency, and characteristics of the physical media of the recording.

There are two basic approaches for placing PCM application data into source packets - frame aligned, and frame not aligned. These approaches are described in sections 2.2.1 and 2.2.2 of this annex, respectively.

2.2.1 Frame Aligned with Start of Source Data Field. A straightforward approach for placing PCM application data into source packets is to place an integral number of minor frames into a single source packet. Under this approach, the beginning of a minor frame of data is aligned with the beginning of the source data field. There are two acceptable methods for using this approach. One is to place a single complete minor frame into a single source packet. The second method is to place more than one complete minor frame of application data into a source packet.

In either case, the implementor may choose to provide for octet alignment on a PCM word basis for all PCM words in the PCM minor frame. If octet alignment is on a PCM word basis, each individual PCM word must either naturally occupy an integral number of octets, or fill bits will be included at the end of each PCM word (LSBs per Figure 1.1 of this standard). If octet aligned on a PCM minor frame basis, only the last PCM minor frame in the packet must end on an integral number of octets. This may require 1-7 fill bits included at the end of the last PCM minor frame (LSBs per Figure 1.1 of this standard)

a.) Single Minor Frame Per Source Packet - By convention, use of this method for packaging a single minor frame in a source packet implies that the first bit of a minor frame will be found at the first bit position of the source data field. Insertion of the PCM frame shall either be contiguous bits of the PCM stream or octet alignment on a PCM word basis (see section 2.2.1 for PCM word octet alignment). An example of PCM application data implementation with one minor frame per source packet is shown in Figure 2-1.

PCM Octet Aligned 510 16-bit Words with No Synch Marker
 Day of Year Calendar Variation CCS to 10⁻² s

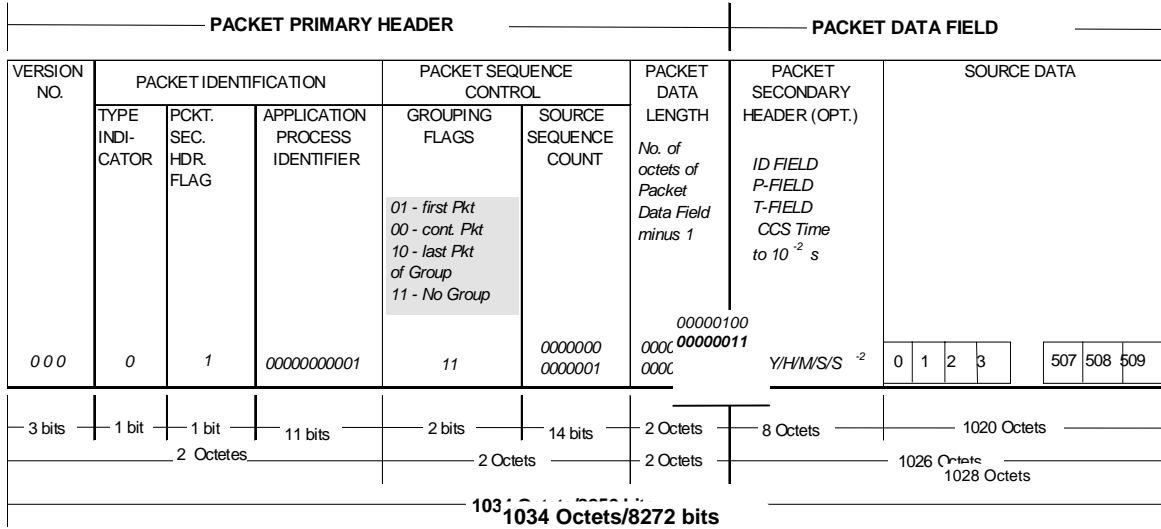


Figure 2-1. PCM, octet aligned with one frame per packet.

b.) Multiple Frames Per Source Packet - By convention under this method, when placing more than one PCM minor frame into a source packet, the first bit of the first PCM minor frame synch marker is placed into the first bit position of the source data field. The first bit of the second PCM minor frame synch marker immediately follows the last bit of the first PCM minor frame, and so on for the remaining PCM minor frames placed into that source packet. An example of a PCM application data implementation with two minor frames per source packet is shown in Figure 2-2.

2.2.2 Frame Not Aligned with Start of Source Data Field. An implementor may choose not to align the PCM minor frame with the start of the source data field of a source packet. Two methods exist. The first method involves stuffing bits of the PCM bit stream, including synch marker, into contiguous bit positions in the entire source data field. The source data field shall be an integral number of octets in length. The source packet length for the PCM bit stream shall be fixed for the duration of the mission. The second method entails aligning each PCM word with the source data field octets (see section 2.2.1 for a description of octet alignment on a PCM word basis). The PCM frames are not aligned with the source data field boundaries.

3. MIL-STD-1553 A/B APPLICATION DATA

MIL-STD-1553 defines the format for messages consisting of command, status, error, and data words. When MIL-STD-1553 messages are recorded using this standard, the 24-bit word structure defined in Chapter 8 of reference [7] for telemetry output shall apply with the exceptions noted in the following sections. In this Annex, MIL-STD-1553 words represented in the Chapter 8 24-bit structure are referred to as “IRIG 106-formatted 1553 words” and MIL-STD-1553 messages represented in Chapter 8 format shall be called “IRIG 106-formatted 1553

messages.” The primary departure of this standard from the Chapter 8 conventions is that there is no need to create a composite output PCM frame structure as defined in section 8.6 of reference [7] for transporting the IRIG 106-formatted 1553 messages because under this standard the source packet effectively replaces the PCM frame. (Note: IRIG 106-formatted 1553 messages in a composite output PCM frame structure per Chapter 8 of reference [7] are treated as PCM application data which is described in section 2 of this annex).

To release the first issue of this standard, only the MIL-STD-1553 A/B application data format contained in IRIG 106 (reference [7]) was considered. Future issues of this standard will consider other more efficient techniques for packetization of MIL-STD-1553 A/B application data.

3.1 APPLICATION LAYER CONSIDERATIONS

MIL-STD-1553 messages shall be formatted and presented to the entity performing packetization layer functions, i.e., creating source packets, as a sequence of IRIG 106-formatted 1553 words as defined in section 8.4 of reference [7]. In addition, where time values need to be inserted into the IRIG-formatted 1553 message, the time words will include embedded high-order, low-order, and microsecond time words formatted according to section 8.5 of reference [7].

3.2 PACKETIZATION LAYER CONSIDERATIONS

For the purposes of placing data into source packets, the key difference between the packetization of PCM data described in Section 2 of this annex and the packetization of IRIG 106-formatted 1553 messages is that the latter are of variable length. When recording IRIG 106-formatted 1553 messages, the maximum source packet length may be freely selected upon consideration of latency, buffer size requirements, efficiency, and characteristics of the physical media of the recording. Because a source packet can vary in length, the implementor may choose to forward a source packet even if it has not been filled to its maximum length.

3.2.1 Insertion of Time Values into IRIG 106-Formatted 1553 Messages. The implementor shall insert time values into IRIG 106-formatted 1553 messages. When placing time values into an IRIG 106-formatted 1553 message, the technique described in section 8.5 of reference [7] applies. Use of this technique requires the placement of all three time words, i.e., high-order time, low-order time, and microsecond time, into each IRIG 106-formatted 1553 message contained in the source packet. This method is illustrated in Figure 3-1 which shows only the contents of the source data field of a source packet.

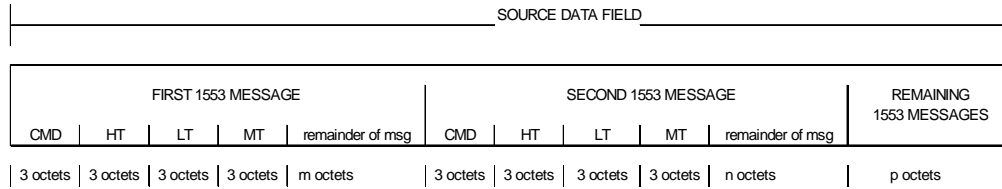


Figure 3-1 Source data field containing IRIG 106-formatted 1553 messages with complete time values in each message.

3.2.2 Source Data Field Structure. Figure 3-2 describes the general format for source packets containing IRIG 106-formatted 1553 messages. By convention, the contents of the source data field shall contain an integral number of IRIG 106-formatted 1553 words, i.e., an IRIG 106-formatted word cannot be split between two source packets. In the example of Figure 3-2, the source data field is 900 octets in length which means that it can hold up to 300 words of IRIG 106-formatted 1553 messages where each word is 3 octets (24 bits) in length.

General Format of a Source Packet of IRIG 106-Formatted 1553 Messages
 Illustrated with a Source Data Field of 900 Octets
 Day of Year Calendar Variation of CCS to 10^{-2} s

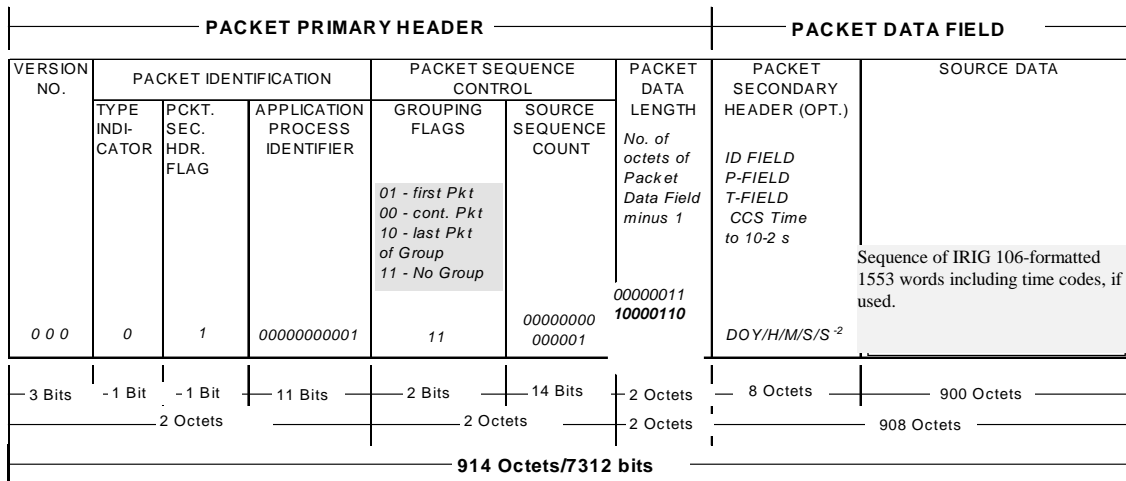


Figure 3-2. Example of the general format for placing IRIG 106-formatted 1553 messages into source packets.

The implementor does not need to align messages to source data field boundaries. The stream of messages is actually viewed more simply as a stream of IRIG 106 formatted words. The implementor may choose to pack as many whole words into the source data field as is convenient. Thus, a single IRIG 106 formatted message may span two source packets. This

method of recording IRIG 106 formatted 1553 messages is illustrated in Figure 3-3 which shows the tail of one message, a complete second message, and the beginning of a third message. The bus ID's shall remain constant for unique application IDs.

Two IRIG 106-Formatted 1553 Messages of 31 24-bit Words and 25 24-bit Words
Stream of 66 IRIG 106-formatted 1553 Words
 Where Messages are not Octet Aligned on Source Data Field Boundaries

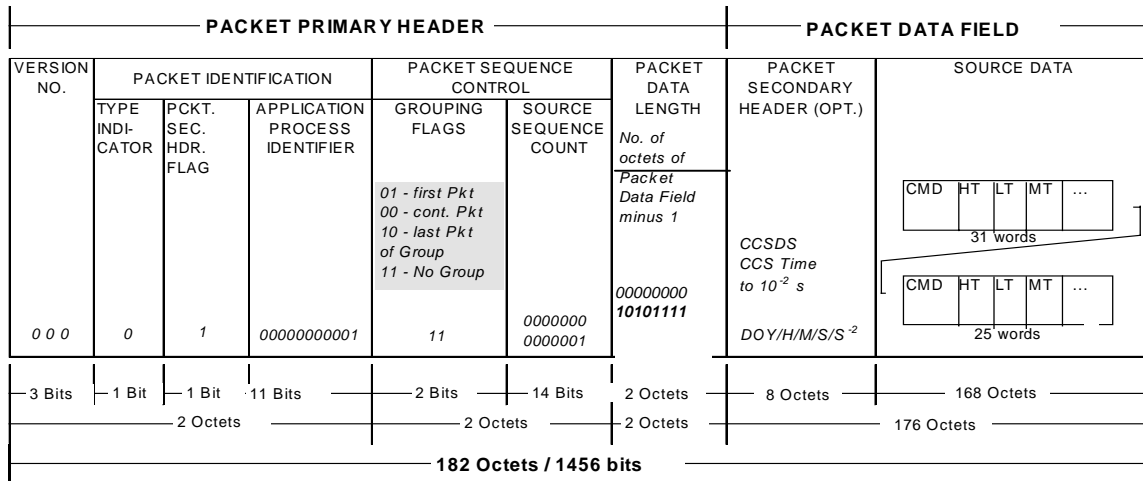


Figure 3-3. IRIG 106-formatted 1553 messages not aligned with source packet illustrated with 240 IRIG 106-formatted 1553 words.

4. PACKETIZATION OF OTHER APPLICATION DATA TYPES

This standard can be used for any other bit stream data, parallel data, asynchronous data, communications data, etc., that can be identified using an application ID and can be placed into source packets as contiguous bits as the data arrives without special formatting at the packetization layer. It is left to the implementor to determine how this data is to be placed into source packets and how to recreate the higher order structure of the original bit stream.

To release the first issue of this standard, no other application data types were specifically considered. Future issues of this standard may provide application notes for specific techniques for packetization of other application data types.

5. DIGITAL VOICE APPLICATION TELEMETRY DATA

Digital voice will normally be an input to a PCM multiplexed data stream. It therefore does not require any special application guidelines other than those described for frame-oriented multiplexed application data.

6. RCC DIGITAL DATA ACQUISITION AND ON-BOARD RECORDING STANDARD DATA

In the future, application data systems may be designed to conform to this standard. In that case, the application process will provide integral source packets of data to the Digital Data Acquisition and On-Board Recording System. These data will be directly inserted into transfer frames for on-board recording.

7. CODING LAYER INTERFACE CONSIDERATIONS

This standard does not encompass the coding layer and therefore does not prescribe how to identify the start of transfer frames. It also does not specify how channel coding is to be applied except as described in appendix B.

7.1 USE OF AN ATTACHED SYNCH MARKER

The use of an Attached Synch Marker (ASM) to indicate the start of a transfer frame is required. It shall conform to the specification as described in appendix B, which provides the appropriate excerpted sections of reference [2].

7.2 CHANNEL CODING

Although not specified in this standard, channel coding for purposes of error detection and/or correction must be integrated into the recording device and be transparently removed during playback. (Future versions of this standard may specify a coding layer standard.)

8. HEADER RECORDS

A header record may be written to a recording device to describe the contents of the recording. A future addition to the application notes will specify the use of the Telemetry Attributes Transfer Standard (TMATS) described in Chapter 9 of reference [7] to convey header information. There is no explicit definition of the contents of header records at this time.

ANNEX A-3

TRANSFER FRAME ERROR DETECTION ENCODING/DECODING GUIDELINE

A3-1 CODING FOR ERROR DETECTION IN TRANSFER FRAMES

This annex describes the error detection encoding/decoding procedure that is recommended for transfer frame coding.

The code specifies the same generator polynomial used by HDLC (ISO), ADCCP (ANSI), V.41 (CCITT), etc. It has the following capabilities when applied to an encoded block of less than 32,768 (2^{15}) bits:

- (1) All error sequences composed of an odd number of bit errors are detected.
- (2) All error sequences containing at most two bit errors anywhere in the encoded block will be detected.
- (3) If a random error sequence containing an even number of bit errors (greater than or equal to 4) occurs within the block, the probability that the error will be undetected is approximately 2^{-15} (or approximately 3×10^{-5}).
- (4) All single error bursts spanning 16 bits or less will be detected provided no other errors occur within the block.

A3-1.1 Encoding Procedure

The encoding procedure accepts an (n-16)-bit data block and generates a systematic binary (n,n-16) block code by appending a 16-bit Frame Check Sequence (FCS) as the final 16 bits of the codeblock. This FCS is inserted into the frame error control word of the transfer frame trailer. The equation for the FCS is:

$$\text{FCS} = [X^{16} \cdot M(X) + X^{(n-16)} \cdot L(X)] \text{ modulo } G(X)$$

where

$M(X)$ is the (n-16)-bit message to be encoded expressed as a polynomial with binary coefficients

$L(X)$ is the presetting polynomial given by:

15

$$L(X) = \sum_{i=0}^{15} x^i \text{ (all "1" polynomial of order 15)}$$

G(X) is the generating polynomial given by:

$$G(X) = X^{16} + X^{12} + X^5 + 1$$

n is the number of bits in the encoded message

\oplus is the modulo 2 addition operator (Exclusive OR)

Note that the encoding procedure differs from that of a conventional cyclic block encoding operation in that:

The $X^{(n-16)} \cdot L(X)$ term has the effect of presetting the shift register to an all "1" state prior to encoding.

A3-1.2 Decoding Procedure

The error detection syndrome, S(X), is given by

$$S(X) = [X^{16} \cdot C^*(X) \oplus X^n \cdot L(X)] \text{ modulo } G(X)$$

where $C^*(X)$ is the received block in polynomial form and S(X) is the syndrome polynomial which will be zero if no error is detected and non-zero if an error is detected.

A3-2 POSSIBLE IMPLEMENTATION

A possible implementation of the above-defined encoding/decoding procedure is described below.

A3-2.1 Encoding

Figure A3-1 shows an arrangement for encoding using the shift register. To encode, the storage stages are set to "one," gates A and B are enabled (closed), gate C is inhibited (open), and (n-16) message bits are clocked into the input. They will appear simultaneously at the output. After the bits have been entered, the output of gate A is clamped to "zero," gate B is inhibited, gate C is enabled, and the register is clocked a further 16 counts. During these counts the required check bits will appear in succession at the output.

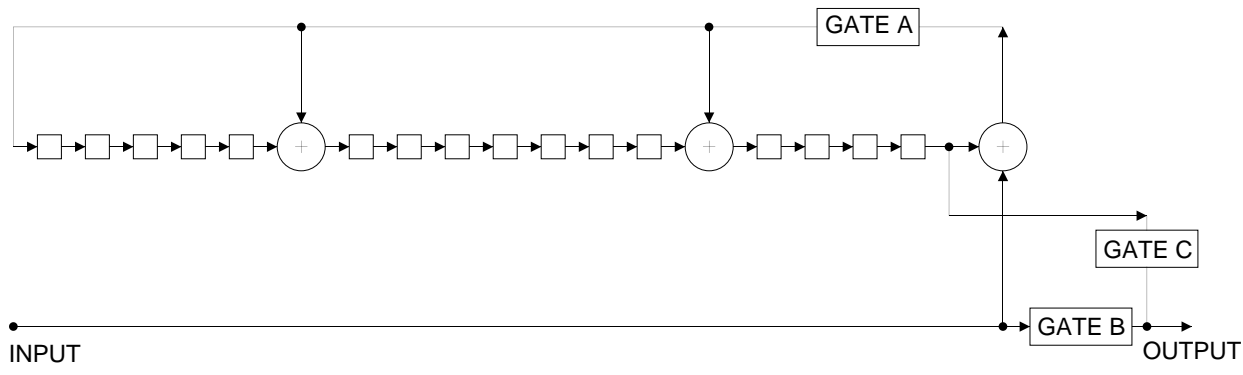


Figure A3-1. Encoder.

A3-2.2 Decoding

Figure A3-2 shows an arrangement for decoding using the shift register. To decode, the storage stages are set to "one" and gate B is enabled. The received n-bits [the (n-16) message bits plus the 16 bits of the FCS] are then clocked into the input. After (n-16) counts, gate B is inhibited. The 16 check bits are then clocked into the input, and the contents of the storage stages are then examined. For an error-free block, the contents will be zero. A non-zero content indicates an erroneous block.

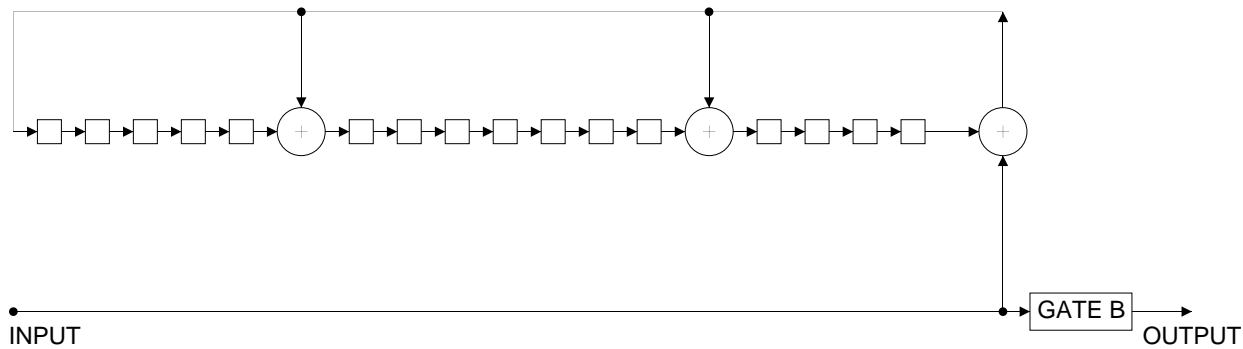


Figure A3-2. Decoder.