CHAPTER 10

DIGITAL RECORDING STANDARD

TABLE OF CONTENTS

CHANGES

Numerous updates to Chapter 10 have been made. Highlighting the changes using different font colors, highlights or other means was not practical and would make reading the document difficult. Therefore, a summary of changes is provided below.

CHANGES TO THIS EDITION OF CHAPTER 10

Paragraph		Description
10.2	Definitions/Acronyms	Added definitions for Memory Clear, Memory Declassification, Memory Sanitization, and Setup Record.
10.3.9.1.2	UDP Transfer Header	Updated Figure 10-3 and Version description.
10.4.3	Ethernet Recorder Interface	Defined a protocol (TELNET) for ground based recorder Ethernet interface.
10.6.1.1	Packet Header	Updated the “Data Type Version” field to include IRI G 106-09.
10.6.4.3	MIL-STD-1553 Bus Data Packets, Format 2	Clarified 16PP194 transaction and word format.
10.6.7.2	Computer Generated Data Packets Format 1	Various changes to support XML TMATS format; Added a note for the Setup Record Configuration Change (SRCC) bit usage.
10.6.x	Intra-Packet Time Stamp (in many 10.6.x)	Corrected the bit in the Packet Flags for the absolute time.
10.6.10.1	Video Packets, Format 0	Added “Byte Alignment (BA)” bit. Clarified Figure 10-75.
10.7.1.2	Optional TELNET Control	Added 10.7.1.2
10.8	Declassification	Added a paragraph on declassification (associated documents, definition and the responsibility).
10.9.7	Mandatory Commands for Processor Devices	Updated Table 1-22 Mandatory Processor Commands.
10.9.12	Declassification Supporting Commands	Added the description of each command.
10.9.14.2	ORB Format	Added more description for the Send and Receive commands.
10.11.2.1	Modified Recording File Annotation	More explanation on “b” on R-x\RI3 and R-x\RI6.

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

ANSI	American National Standards Institute
API	Application Programming Interface
ARINC	Aeronautical Radio, Incorporated
CCM	Command and Control Mnemonic
CD	Collision Detection
CDB	Command Descriptor Block
CSDW	Channel Specific Data Word
CSMA	Carrier-Sense Multiple Access
CSR	Control and Status Register
DCRsi	Digital Cartridge Recording System (a recording method and digital data *interface*)
DITS	Digital Information Transfer System
DoD	Department of Defense
ECL	Emitter-coupled Logic
FCP	Fibre Channel Protocol
FCPL	Channel Private Loop
FC-PLDA	Fibre Channel Private Loop SCSI Direct Attach
GCC	GNU Compiler Collection
GPS	Global Positioning System
HSDB	High Speed Data Bus
I/O	Input/Output
IC	Intelligence Community
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IOCTL	I/O control code
IP	Internet Protocol
IRIG	Inter-range Instrumentation Group
iSCSI	Internet Small Computer Systems Interface
iSNS	Internet Storage Name Service
ISO	International Organization for Standards
IT	Information Technology
ITU	International Telecommunications Union
IU	Information Unit
KLV	Key-Length-Value
LBA	Logical Block Address
LSB	Least Significant Bit
LUN	Logical Unit Number
MAC	Media Access Control
MAS	Military Agency for Standardization
MBR	Master Boot Record
MISB	Motion Imagery Standards Board
MISP	Motion Imagery Standards Profile
MPEG	Moving Picture Experts Group
MUX	Multiplexer

NADSI	NATO Advanced Data Storage Interface
NATO	North Atlantic Treaty Organization
NSG	National System for Geospatial Intelligence
NSG	National System for Geospatial Intelligence
NSIF	NATO Secondary Imagery Format
NSIF	NATO Secondary Imagery Format
ORB	Operation Request Block
OS	Operating System
PCM	Pulse Code Modulation
PDU	Protocol Data Unit
PS	Program Streams
RCC	Range Commanders Council
RFC	Request For Comment
RIU	Remote Interface Unit
RMM	Removable Memory Module
RS	Recommended Standard
RTC	Relative Time Counter
SAM	SCSI Architecture Model
SBC	SCSI Block Commands
SBP	Serial Bus Protocol
SCSI	Small Computer Systems Interface
SD	Standard Definition
SLP	Service Location Protocol
SPC	SCSI Primary Commands
SPT	SCSI Pass Through
SRB	SCSI Request Block
STANAG	Standardization Agreement
TCP	Transmission Control Protocol
TM	Telemetry
TMATS	Telemetry Attributes Transfer Standard
TS	Transport Stream
UART	Universal Asynchronous Receiver and Transmitter
UDP	User Datagram Protocol
WMUX	Weapons MUX

CHAPTER 10

DIGITAL RECORDING STANDARD

10.1 General

A large number of unique and proprietary data structures have been developed for specific data recording applications which required unique decoding software programs. The activities of writing unique decoding software, checking the software for accuracy, and decoding the data tapes are extremely time consuming and costly. In the late 1990s, the test ranges started to see the implementation of non-tape-based, high-data-rate recorders, the most predominate of which were solid-state memory devices. Then, as high-data-rate digital recorders were fielded and as solid state technology began to emerge, the Telemetry Group (TG) saw the need and formed an ad hoc committee for a computer-compatible digital data acquisition and recording standard.

There is a need for a digital data acquisition and recording standard (see the functional layout at Figure 10-1) that supports a broad range of requirements, including:

a. Data download and interface.
b. One or more multiplexed data streams.
c. One or more single data streams.
d. Data format definitions.
e. Recorder control.
f. Media declassification.
g. Data interoperability.

Specifically, this digital recording standard shall be compatible with the multiplexing of both synchronous and asynchronous digital inputs such as pulse code modulation (PCM) and MIL-STD-1553 data bus, time, analog, video, Aeronautical Radio, Inc. (ARINC) 429, discrete, and RS-232/422 communication data. This digital recording standard will allow use of a common set of playback/data reduction software to take advantage of emerging random access recording media.

NOTE	Within this standard, where text and/or figures are used to provide descriptions, meaning, and/or explanations, the text shall take precedence over figures.

10.1.1 Interface Levels. The purpose of this chapter is to establish a common interface standard for the implementation of digital data acquisition and recording systems by the organizations participating in the Range Commanders Council (RCC). This standard does not imply hardware architecture such as the coupling of data acquisition, multiplexing, and media storage. The required interface levels are contained in this standard (see a through e below). In addition, declassification requirements are discussed in paragraph 10.8, ground based recording is discussed in paragraph 10.10, and data interoperability requirements are discussed in paragraph 10.11.

a. Data Download and Electrical Interface, which is the physical interface for data access, is defined in paragraph 10.4.

b. Interface File Structure, which defines data access structure, is described in paragraph 10.5.

c. Data Format Definition, which defines data types and packetization requirements, is defined in paragraph 10.6.

d. Recorder Control and Status, which defines command and control mnemonics, status, and their interfaces, is described in paragraph 10.7.

e. IEEE 1394B Interface to Recorder Removable Media is defined in paragraph 10.9.

f. Ground Recorder Interface, which defines unique interoperability requirements of a ground recorder, is described in paragraph 10.10.

g. Data Interoperability, which defines requirements for the annotation, modification, and exchange of recorded data, is described in paragraph 10.11.

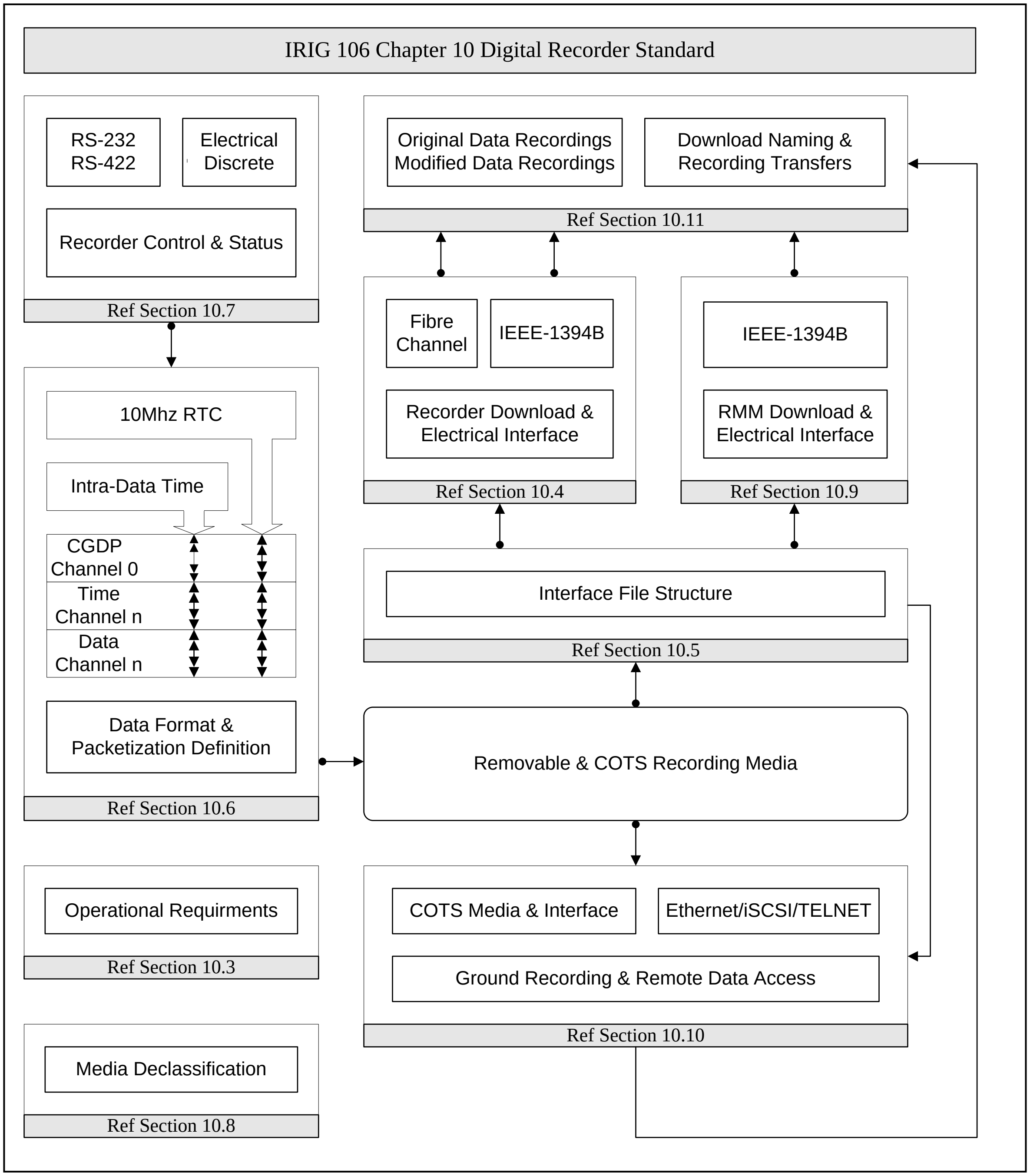


Figure 10-1. Functional layout of standard.

10.2 Definitions/Acronyms

The following are definitions and acronyms which are used in this standard and are provided as a means of removing ambiguities within the standard.

Absolute Time: Is a hypothetical time that either runs at the same rate for all the observers in the universe or the rate of time each observer can be scaled to by multiplying the observer's rate by a constant.

AET: Audio Encoding Type

ARINC: Aeronautical Radio, Inc

ATA: Advanced Technology Attachment

Basic Character Set (BCS): Based on ISO/IEC 10646-1, the Universal Multiple-Octet Coded Character Set (UCS). The North Atlantic Treaty Organization (NATO) Imagery Interoperability Architecture (NIIA) limits characters to a subset rather than allowing all characters. The subset will be single octets, known as the Basic Character Set (BCS).

Block: The smallest unit of addressable memory that can be written to, read from, and/or erased.

Bad Block: A block determined to be unreliable for storing user data.

Bad Block Table: A table of bad block entries for a memory board. The data stored in the entry identifies the chip and block number of the bad block. The table entry also contains a flag field. The flag field is used to determine the circumstance in which the bad block was detected. It also provides a flag indicating whether the corresponding bad block has previously been "Secure Erased."

Byte: A contiguous set of 8 bits that are acted on as a unit.

Channel ID: A unique value assigned to each channel in a system. Each channel must have a unique Channel ID (data channels and playback channels).

Channel Specific Data Word: A required word for each data type channel that has data specific information.

Commercial Off-The-Shelf (COTS) Media: Any recording media (such as hard disks, solid state drives, RAID, and JBOD) that is ready-made and available for sale to the general public. In the context of this standard, all COTS Media shall conform to paragraph 10.3.10 with the exception of Ground Based Recorders COTS Media.

COTS Media Interface: Any recording COTS Media interface (such as PATA, SATA, SCSI, IEEE-1394, USB, Ethernet) that is ready-made and available for sale to the general public.

Data Streaming: Streaming of current value data whether it is being recorded or not, and playback streaming of recorded data from a file. Data Streaming sends the data to one or more destinations simultaneously (e.g., recording media, recorder data interfaces).

DCRsi: Reference AMPEX Data Systems Corp.

Ground Based Recorder: Records IRIG-106 Chapter 10 data and may optionally reproduce the recorded data. It supports the same data format and remote command and control operations as an On-Board Recorder. Instead of an RMM, a ground recorder will generally use COTS media and/or data streaming to a network. Ground Based Recorder requirements to be in 100 percent compliancy with this standard are defined in paragraph 10.3.

IEEE-1588 Time: Time as specified by IEEE Std 1588-2002, "1588 IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems."

FC-PI: Fibre Channel-Physical Interfaces.

FC-FS: Fibre Channel-Framing and Signaling.

Intra-Packet Data Header: A header containing time, status, and/or data information for the tagging of data inside a packet.

IAW: In Accordance With.

JBOD: Just a bunch of disks.

Long Word: A contiguous set of 32 bits that are acted on as a unit.

lsb: The "least significant bit" of a series of bits.

LSB: The "least significant byte" of a series of bytes.

LSW: The "least significant word" of a series of words.

LSLW: The "least significant long word" of a series of long words.

Mandatory: Defines a Mandatory requirement of this standard for full compliancy. Mandatory requirements as defined in this standard are based on the use of "shall" and are defined in paragraph 10.1.

MAS: Military Agency for Standardization

Magic Number: An identifier for the directory block. This identifier is a value chosen to support discovery of lost directory entries and directory reconstruction after a fault.

Memory Board: Printed circuit board containing flash memory devices used to store user data.

Memory Clear: Rendering stored information unrecoverable unless special utility software or techniques are used.

Memory Declassification: Administrative decision or procedure to remove or reduce the security classification of the subject media.

Memory Sanitization: The removal of information from information system media such that data recovery using known techniques or analysis is prevented. Sanitizing includes the removal of data from the media and verification of the action. Properly sanitized media may be subsequently declassified upon observing the organization's respective verification and review procedures.

MISM: Motion Imagery Standards Matrix.

msb: The "most significant bit" of a series of bits.

MSB: The "most significant byte" of a series of bytes.

MSW: The "most significant word" of a series of words.

MSLW: The "most significant long word" of a series of long words.

Multiplexer: The entity that includes all the inputs, control interfaces, and functionality required to properly record data.

NADSI: NATO Advanced Data Storage Interface.

NATO: North Atlantic Treaty Organization

NIIA: NATO Imagery Interoperability Architecture.

Non-volatile: Memory media that retains data when power is removed.

On-Board Recorder: An On-Board Recorder is the basis and original justification for the IRIG-106 Chapter 10 Standard. On-Board Recorder requirements to be in 100 per cent compliancy with this standard are defined in paragraph 10.3.

Optional: Defines an optional requirement of this standard but is not required for full compliancy. Optional requirements as defined in this standard are based on the use of "optional" and are defined in paragraph 10.1. When optional requirements are implemented they shall be IAW with this standard.

Packet: Encapsulates a block of observational and ancillary application data to be recorded.

Packet Generation: The placing of observational and ancillary data into a packet.

Packet Generation Time: The time period from when the first bit of observational and ancillary data was placed into a packet until it is no longer being placed into the packet and the packet is closed.

Packet Header: Identifies the source and characteristics of the data packet and encapsulation environment.

Packet Secondary Header: Contains the Packet Header time.

Page: Storage unit within the flash device. A page is the smallest storage unit that can be written.

PATA: Parallel ATA.

PCM: Pulse code modulation

Quad Word: A contiguous set of 64 bits acted on as a unit.

RAID: Redundant Arrays of Independent Drives/ Redundant Arrays of Inexpensive Disks.

RMM: Removable Memory Module.

RR: RT to RT Transfer.

Recorder: Is used where a function or requirement shall apply to both an On-Board Recorder and a ground based recorder.

Recording: Is defined as the time interval from first packet generated (which by mandatory requirements is a Computer Generated Data Packet Format 1) and committed to the recorder media to the last packet generated and committed to the recorder media. Packet Generation Time and Stream Commit Time, as defined within the standard, apply.

Relative Time Counter (RTC): A free-running binary counter common to all data channels. The counter shall be derived from an internal crystal oscillator and shall remain free-running during each recording. The applicable data bit to which the 48-bit value applies will be defined in each data type section.

Removable Memory Module (RMM): The element of the On-Board Recorder that contains the stored data.

SATA: Serial ATA.

SCSI: Small Computer Systems Interface.

Setup Record: TMATS IAW IRIG-106 Chapter 9 annotated in the Computer Generated Data Format 0 packet.

STANAG: Standardization Agreement, NATO.

Stream: All packets from all enabled channels (including Computer Generated Data) that are generated until the end of a recording.

Stream Commit Time: The time span in which all Generated Packets must be committed to a stream.

TMATS: Telemetry Attributes Transfer Standard as defined in IRIG 106 Chapter 9.

UCS: Universal Multiple-Octet Coded Character Set.

USB: Universal Serial Bus.

Word: A contiguous set of 16 bits acted on as a unit.

10.3 Operational Requirements

10.3.1 Recorder Compliancy Requirements. The following table represents the mandatory recorder requirements to meet 100 per cent compliancy with this standard. Meeting these compliancy requirements guarantees interoperability of recorders, recorder media and recorded data. Optional functions and/or capabilities are not shown but when implemented in a recorder shall be in accordance with the definitions in this standard in order to meet 100 per cent compliancy of this standard.

10.3.1.1 On-Board Recorder Mandatory Compliancy Requirements.

Applicable Compliancy Paragraph	Function/Capability
	RECORDER ELECTRICAL INTERFACES
10.3, 10.4	Fibre Channel and/or IEEE-1394B Data Download Port
10.3, 10.7	Discrete Lines and/or RS-232 and 422 Full Duplex Communication
10.3	External Power Port
	RECORDER DOWNLOAD INTERFACE PROTOCOLS
10.4, 10.9	Fibre Channel SCSI and/or IEEE-1394B SCSI/SBP-2
	RECORDER CONTROL/STATUS INTERFACE PROTOCOLS
10.7	Discrete Control/Status and/or RS-232 and 422 Control/Status
	RMM ELECTRICAL INTERFACE AND POWER
10.3, 10.9	IEEE-1394B Bilingual Socket
	COTS MEDIA ELECTRICAL INTERFACES
10.3	COTS Media Interface
	RMM INTERFACE PROTOCOLS
10.9	IEEE-1394B SCSI/SBP-2
	COTS MEDIA INTERFACE PROTOCOLS
10.3	COTS Media Interface
	RECORDER MEDIA /RMM/COTS MEDIA INTERFACE FILE STRUCTURE
10.5	Directory, File Structures and Data Organization
10.3.7	Directory and File Table Entries
	PACKETIZATION AND DATA FORMAT
10.6	Packet Structures, Generation, Media Commitment and Time Stamping
10.6	Data Type Formats
	DATA INTEROPRABILITY
10.11	Original Recording Files

10.3.1.2 Ground Based Recorder Mandatory Compliancy Requirements.

Applicable Compliancy Paragraph	Function/Capability
	RECORDER ELECTRICAL INTERFACES
10.10	Ethernet
	RECORDER REMOTE INTERFACE PROTOCOLS
10.10, 10.4	iSCSI and/or TELNET
	COTS MEDIA ELECTRICAL INTERFACES
10.10	COTS Media Interface
	COTS MEDIA INTERFACE PROTOCOLS
10.10	COTS Media Interface
	REMOTE DATA ACCESS INTERFACE FILE STRUCTURE
10.5	Directory, File Structures, and Data Organization
10.3.6	Directory and File Table Entries
	PACKETIZATION AND DATA FORMAT
10.6	Packet Structures, Generation, Media Commitment, and Time Stamping
10.6	Data Type Formats
	DATA INTEROPRABILITY
10.11	Original Recording Files

10.3.2 Required Configuration. An On-Board Recorder, as a minimum, shall provide the following functionality:

a. Data Download port.
b. Recorder Control/Maintenance port.
c. External power port.

The required data download port interface shall be IAW paragraph 10.4. This combination will allow data extraction and transfer from any recorder to any paragraph 10.4 compliant intermediate storage unit. The required control port interface shall be IAW paragraph 10.7.

10.3.3 Exclusions to Standard. The physical size, configuration, and form factor for the On-Board Recorder and the RMM are not controlled by this standard. Due to the variation in capacity/rate/cost requirements of the users, this standard does not specify the technology to be used in the RMM or the On-Board Recorder.

10.3.4 Internal System Management. Any processing performed on the stored data by the On-Board Recorder (e.g., for the purposes of internal system management, error detection and correction (EDAC), physical frame formatting, etc.) shall be removed from the stored data when the stored data is downloaded or transferred from storage media.

10.3.5 Data Download. On-Board Recorders may have an RMM capability or the On-Board Recorder can be removed from the acquisition platform and taken to a ground station for data download. Reference paragraph 10.4.1 for recorder download and electrical interface, paragraph 10.9 for RMM interface, and paragraph 10.11 for data transfer and file management.

10.3.6 IEEE-1394b Interface to Recorder Media. Serial Interface to On-Board Recorder media shall be accomplished utilizing IEEE-1394b interface. A nine-pin IEEE-1394b interface shall be provided on the media to allow direct download of data to host computer or storage device.

10.3.7 Required File Tables Entries. Within paragraph 10.5, Table 10-3 “FileSize,” “File Create Date,” “File Create Time,” and “File Close Time” are either optional or can be empty (filled with 0x2D) if data is unavailable. Table 10-3 has been adopted from STANAG 4575 but in the case of IRIG-106 Chapter 10 unless the “Time Type” is 0xFF (Time Data Packet) and the Time Data Packet source is 0xF (None) date and time will always be available.

10.3.7.1 File Table Entry Conditions. If Table 10-2 “Shutdown” value is 0xFF and the “Time Type” is 0xFF and the Time Data Packet source is not 0xF “FileSize,” “File Create Date,” “File Create Time” and “File Close Time” entries shall be filled in their entirety.

If Table 10-2 “Shutdown” value is 0x00 and the “Time Type” is 0xFF and the Time Data Packet source is not 0xF “FileSize,” “File Create Date,” “File Create Time” and “File Close Time” entries shall be filled in their entirety.

10.3.8 Recorder Configuration File. A recorder setup configuration file can reside on the recorder or optionally reside in the RMM. Recorder setup configuration must be IAW IRIG-106 Chapter 9. Recorder setup configurations shall be programmed IAW with paragraph 10.7 of this standard. Optionally the recorder can be configured from a Chapter 10 configuration file residing in the RMM. The RMM recorder configuration file will have priority over setup records residing in the recorder.

10.3.8.1 Recorder Configuration File Location. When a setup record transfer to a recorder is made via the RMM Computer Generated Data, Format 1 Setup Record packet(s) will be used. The RMM shall contain a “Directory” and one “Directory Block File Entry” IAW paragraph 10.5.2.

a. All Directory Block Format fields shall be IAW table 10-2. The field “n File Entries” value shall be 1.
b. All Directory Entry Format fields shall be IAW table 10-3. The field “Time Type” value shall be 0x01, System Time. The field “Name” value shall be:

recorder_configuration_file_SAVE_n

This will notify the recorder to use the Recorder Configuration Transfer File for the next recording and store the setup information contained within the file to non-volatile memory in the recorder pre-defined setup location n, where n is a value of 0-15. This shall be the equivalent of sending .TMATS SAVE [n] and .SETUP [n] commands.

10.3.8.2 Recorder Configuration File Structure. The Recorder Configuration File structure will only contain Computer Generated Data, Format 1 Setup Record packets. More than one packet is allowed only if the required recorder configuration information exceeds the packet size limits in paragraph 10.6.1, thus forcing more than one Computer Generated Data, Format 1 Setup Record packet. The standard method of using the sequence counter will be utilized until all the configuration information has been packetized.

10.3.8.3 Configuration of Recorder from RMM: A setup record may reside in the RMM and be utilized for configuration of the recorder. A Computer Generated Data, Format 1 Setup Record packet(s) will be used. The RMM shall contain a “Directory” and at least one “Directory Block File Entry” IAW paragraph 10.5.2.

a. All Directory Block Format fields shall be IAW table 10-2. The field “n File Entries” value shall be 1.

b. All Directory Entry Format fields shall be IAW table 10-3. The field “Time Type” value shall be 0x01, System Time. The field “Name” value shall be…

recorder_configuration_file_SETUP_RMM

This will notify the recorder to configure from the RMM. The recorder configuration file shall NOT be able to be erased by the recorder .ERASE command or discrete command.

10.3.9 Recorder Data Streaming Transport. Data Streaming Transport may be accomplished across the paragraph 10.4 Recorder Download and Electrical Interfaces using the definitions in paragraph 10.3 and commands in IRIG-106 Chapter 6. For Ground Based Recorders, this will be accomplished across the required Remote Data Access Ethernet Interface.

10.3.9.1 Ethernet. Ethernet is an optional interface for On-Board Recorders; however, it is a required interface for Ground-Based Recorders (10.10). IAW paragraph 10.4. This Ethernet interface can optionally be used for Data Streaming using User Datagram Protocol (UDP). This will be accomplished with the IRIG-106 Chapter 6 PUBLISH command and the following structure added to UDP packets with paragraph 10.6 (Data Format) packets.

10.3.9.1.1 Ethernet Packet Payload Byte Order. The byte ordering within the UDP packet payload shall be IAW paragraph 10.5.3.2. This UDP packet payload shall include the UDP Transfer Header and the IRIG-106 Chapter 10 data.

10.3.9.1.2 UDP Transfer Header. Network broadcasting limitations of paragraph 10.6 packets (up to 128M bytes or 512K bytes) across Ethernet will require use of an added packet transmission UDP Transfer Header. This is required as partial packets or partial out of order packets may be broadcast and will be missing information required for data reconstruction at the subscriber.

The following structure (Figure 10-2) shall be used for UDP Transfer Headers in UDP packets containing one or more full IRIG-106 Chapter 10 data packets:

<table>
<tr><td colspan="3">MSW LSW</td></tr>
<tr><td>WORD 1</td><td colspan="2">WORD 0</td></tr>
<tr><td colspan="3">msb lsb</td></tr>
<tr><td>81 … 8</td><td>7 … 4</td><td>3 … 0</td></tr>
<tr><td>UDP Message Sequence Number</td><td>Type of message</td><td>Version</td></tr>
</table>

Figure 10-2. UDP Transfer Header for non-segmented data.

The following structure (Figure 10-3) shall be used for UDP Transfer Headers in UDP packets containing a segmented IRIG-106 Chapter 10 data packet:

MSW				**LSW**
WORD 1		**WORD 0**		
msb				**lsb**
31	**8**	**7**	**4**	**3 0**
UDP Message Sequence Number		Type of message		Version

WORD 3		**WORD 2**
msb		**lsb**
31 24	**23 16**	**15 0**
Reserved	Channel Sequence Number	Channel ID

WORD 5	**WORD 4**
msb	**lsb**
31	**0**
Segment Offset	

Figure 10-3. UDP Transfer Header for segmented data.

- Version. (4 bits)
  0000: Reserved
  0001: Version 1
  0010-1111: Reserved
- Type of Message. (4 bits)
  0000: Full packets.
  0001: Segmented.
  0010-1111: Reserved.
- UDP Message Sequence Number. (24 bits). Binary value incrementing by one for each UDP message even if segment of IRIG-106 Chapter 10 packet.
- Channel ID. (16 bits). Segmented Packets Only, Channel ID of the data in the IRIG-106 Chapter 10 packet.
- Channel Sequence Number. (8 bits). Segmented Packets Only, Channel Sequence Number of the data in the IRIG-106 Chapter 10 packet.
- Reserved. (8 bits). Reserved.
- Segment Offset. (32 bits). Segmented Packets Only, Position of the data in the IRIG-106 Chapter 10 packet.

10.3.9.1.3 UDP IRIG-106 Chapter 10 Packet Transfer. When more than one complete Chapter 10 packet is contained within a UDP packet, there shall be an integral number of Chapter 10 packets. Chapter 10 packets shall be sent in the same sequence as recording segment of a packet and shall be ordered (segment offset incrementing). Figures 10-4 and 10-5 present the sequence of the general UDP network broadcast of full or segmented packets.

- When using IPv4, total length of message shall be less than 32768 – 42 (IP + UDP header) = 32726.
- When using IPv6, the use of jumbograms removes the need of segmented IRIG-106 Chapter 10 packets so a jumbogram shall always contain an integer number of IRIG-106 Chapter 10 packets.
- Stream Commit Time applies to recorders using Ethernet Data Streaming Transport.

UDP HEADER
UDP TRANSFER HEADER
IRIG-106 CHAPTER 10 PACKET 1
:
IRIG-106 CHAPTER 10 PACKET n

Figure 10-4. General UDP Network Broadcast (full packet).

UDP HEADER
UDP TRANSFER HEADER
IRIG-106 CHAPTER 10 PACKET Segment

Figure 10-5. General UDP Network Broadcast (segmented packet).

10.3.10 COTS Media. In conjunction with an On-Board Recorder and/or a multiplexer when a RMM or internal On-Board Recorder media is not used, COTS media can be used for recording media. The COTS Media shall be accessible at a minimum from the On-Board Recorder data download port IAW paragraph 10.4 and optionally by at least one COTS Media Interface. When accessing COTS Media the Interface File Structure Definition defined in paragraph 10.5 shall be presented at the On-Board Recorder or COTS Media Interface.

10.4 Data Download and Electrical Interface

At a minimum, the required recorder download port interface (see paragraph 10.3.2) shall be Fibre Channel or IEEE1394B and optionally Ethernet (paragraph 10.4.3). The physical, signaling, and command protocols contained in paragraphs 10.4.1 and 10.4.2 are a subset of, and adapted from the North Atlantic Treaty Organization (NATO) Military Agency for Standardization (MAS) Standardization Agreement (STANAG) NATO Advanced Data Storage Interface (NADSI) Number 4575 (STANAG 4575).

10.4.1 Fibre Channel (FC) Recorder Download Interface.

10.4.1.1 Physical and Signaling. The interface shall comply with FC-PH (Physical Interfaces) and FC-FS (Framing and Signaling) in paragraph 10.9, with configuration options as specified.

a. Physical Media. Fibre Channel copper interface will be utilized.

b. Signaling Rate. The transmission signaling rate shall be 1.0625 Gbaud.

10.4.1.2 Command Protocol. The interface shall conform to the requirements of the Fibre Channel Private Loop SCSI Direct Attach (FC-PLDA, ANSI NCITS TR19-1998) interoperability, except as defined herein. Table 17 of FC-PLDA specifies a control protocol using a subset of commands, features, and parameters defined for the Small Computer System Interface (SCSI-3).00 Table 17 of FC-PLDA also defines the command feature and parameter usage categories of “Required,” “Allowed,” “Invokable,” and “Prohibited” between the SCSI Initiator and Target. These definitions assume that the Target is a magnetic disk drive or equivalent device.

The control protocol must support a number of data storage media types. Only the minimum set of SCSI commands needed to download mission data from a memory cartridge are defined as “Required.” FC-PLDA SCSI commands, features, and parameters not defined as “Required” for this standard, are redefined as “Allowed” so that they may be implemented as appropriate. In addition, it is recognized that numerous applications will be required to write to the Removable Memory Module (RMM) as well. Commands required to format and/or write to a RMM are defined as “Recommended.” These commands are not required for any Standardization Agreement (STANAG) 4575 RMM implementation. However, if the functions are incorporated into an application, the “Recommended” commands shall be used to preclude a proliferation of unique commands. All other “Required” FC-PLDA SCSI commands, features, and parameters not defined as “Required” or “Recommended” for STANAG 4575, are redefined as “Allowed” such that they may be implemented as appropriate. Table 10-1 provides the five “Required” STANAG 4575 SCSI commands and two “Recommended” commands and their features and parameter usage definitions. The NATO Advanced Data Storage Interface (NADSI) compliant recorders may respond to the Inquiry command with a 00h SCSI Version code and the ground/shipboard NADSI host must be prepared to accept this response and restrict SCSI commands issued to the STANAG 4575 mandatory set.

The RMM shall provide Fibre Channel Responder functionality and the NATO ground station shall provide Fibre Channel Originator functionality. The RMM shall also provide SCSI Target functionality and the NATO ground station shall provide SCSI Initiator functionality. When an RMM is powered up directly through the NADS Interface, the RMM shall automatically initialize into a mode where the NADSI port is active and is the priority data and control interface.

TABLE 10-1. "REQUIRED" AND "RECOMMENDED" SCSI COMMANDS, FEATURES, AND PARAMETERS

FEATURE (COMMAND)	INITIATOR	TARGET*	NOTES
INQUIRY	I	R	
Standard INQUIRY data (bytes 0-35)	I	R	
EVPD = 1	I	R	
Enable Vital Product Data page codes:			
0x00 (supported vital product pages)	I	R	
0x80 (unit serial number page)	I	R	
0x81 (implemented operations definition pg)	I	A	
0x82 (BCS implemented operations def pg)	I	A	
0x83 (device identification page)	I	R	
READ (10)	I	R	
DPO = 0	I	A	1
DPO = 1	I	A	1
FUA = 0	I	A	2
FUA = 1	I	A	2
RelAdr = 0	R	R	
RelAdr = 1	P	P	3
READ CAPACITY	I	R	
RelAdr = 0	R	R	
RelAdr = 1	P	P	3
PMI = 0	I	R	
PMI = 1	I	A	
TEST UNIT READY	I	R	
REQUEST SENSE	I	R	
WRITE (10)	C	C	4
DPO = 0	I	A	1
DPO = 1	I	A	1
FUA = 0	I	A	2
FUA = 1	I	A	2
RelAdr = 0	C	C	
RelAdr = 1	P	P	3
FORMAT UNIT	C	C	4, 5
FMT DATA = 0	I	A	
CMPLST = 0	I	A	
DEFECT LIST FMT = 0	I	A	
INTERLEAVE = 0	I	A	

TABLE 10-1. "REQUIRED" AND "RECOMMENDED" SCSI COMMANDS, FEATURES, AND PARAMETERS

NOTES:

1. The Disable Page Out (DPO) bit is associated with a device data caching policy.
2. The Force Unit Access (FUA) bit is associated with whether the device may or may not return the requested Read data from its local cache.
3. Relative Offset is prohibited, since this requires the use of linking, which is prohibited.
4. All RMMs not supporting Recommended or Allowed commands shall respond to these commands with an appropriate error response and shall not cease operations.
5. The FORMAT command shall implement an initialization of the target device such that the entire user memory space shall be writable. After performing this command, the content of the memory may be indeterminate.

*__LEGEND__

P = Prohibited: The feature shall not be used between NADSI compliant devices.

R = Required: The feature or parameter value shall be implemented by NADSI compliant devices.

C = Recommended: The feature is recommended and shall be used for applications requiring the functionality of these commands. The initiator determines if a recommended feature/parameter is supported via a required discovery process or a minimal response by the recipient.

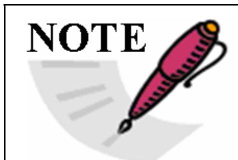
A = Allowed: The feature or parameter may be used between NADSI compliant devices. The initiator determines if an Allowed feature/parameter is supported via a required discovery process or a minimal response by the recipient.

I = Invokable: The feature or parameter may be used between NADSI compliant devices. The recipient shall support "Invokable" features or provide a response that it is not implemented as defined by the appropriate standard.

10.4.2 IEEE-1394B Recorder Interface. The IEEE-1394B recorder download interface shall use the same mechanisms as paragraph 10.9 where applicable.

10.4.2.1 Physical and Signaling. The interface shall allow control of Vendor Specific Recorder devices. The command protocol shall be IAW 10.4.1.2 and Table 10-1.

10.4.2.2 Recorder Communication. The fundamental method of communicating shall be in accordance to the IEEE 1394B protocol. Packets sent and received shall be asynchronous transmissions. IEEE-1394B packets shall encapsulate Serial Bus Protocol (SBP-2) formatted packets for the transport of commands and data. Recorder devices are to use SCSI command set(s) and therefore SCSI commands and status shall be encapsulated in SBP-2 Operation Request Blocks (ORB).



SBP-2 provides for the transport of 6, 10, and 12-byte SCSI Command Descriptor Blocks within a command ORB.

10.4.3 Ethernet Recorder Interface. The On-Board Recorder Ethernet interface shall use iSCSI protocol. This will allow common SCSI protocols across Fibre Channel, IEEE-1394B (SPB-2) and Ethernet (iSCSI) recorder download interfaces. The iSCSI protocol will be implemented as the Host Ground System acting as an *initiator* and the recorder acting as the *target*.

Ground Based Recorder Ethernet interface shall use the TELNET protocol. As a minimum requirement, the TELNET interface will implement Internet Engineering Task Force (IETF) RFC 854 (TELNET Protocol Specification), RFC 855 (TELNET Option Specification), and RFC 1184 (Line Mode Option). The protocol will support IRIG106 Chapter 6 Command and Control Mnemonics (Reference 10.7.8) over a TCP/IP connection on port # 10610. The TELNET interface must respond with a “*” when a connection is made.

10.4.3.1 Target LUN Assignments. The following iSCSI target LUN assignments shall be used:

a. LUN 0 or 32 shall be used for recorder data download via paragraph 10.5 interface.
b. LUN 1 or 33 shall be used for recorder IRIG-106 Chapter 6 Command and Control Mnemonics (Reference 10.7).

10.4.3.2 Naming and Addressing. The Host Ground System (initiator) and Recorder (target) devices on the network must be named with a unique identifier and assigned an address for access. The iSCSI initiators and target nodes can either use an iSCSI qualified name (IQN) or an enterprise unique identifier (EUI). Both types of identifiers confer names that are permanent and globally unique.

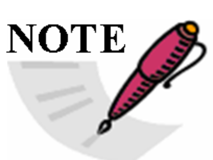
Each node has an address consisting of the IP address, the TCP port number, and either the IQN or EUI name. The IP address can be assigned by using the same methods commonly employed on networks, such as Dynamic Host Control Protocol (DHCP) or manual configuration.

10.4.3.3 Physical and Signaling. The interface shall allow control of Vendor Unique Recorder devices. The command protocol shall be IAW 10.4.1.2 and Table 10-1.

10.4.3.4 Recorder Communication. The fundamental method of communicating shall be in accordance to the iSCSI protocol. Packets sent and received shall be asynchronous transmissions.

10.5 Interface File Structure Definitions

The definitions in this paragraph are a subset of, and were adapted from Section 3 of STANAG 4575, File Structure Definition. This file structure was selected to facilitate host computing platform independence and commonality. By incorporating an independent file structure, backward and forward compatibility is ensured for the life of the standard.

	This file structure definition does not define how data is physically stored on the recorder media but provides a standardized method for access of the stored data at the interface. Data can be organized in any way appropriate to the media, including multiple directories, as long as the file structure IAW paragraph 10.5 is maintained or seen at the interface (paragraph 10.4).

10.5.1 Data Organization. A data recording can contain a single file, which is composed of one (1) or more types of packetized data, or multiple files, in which one (1) or more types of data are recorded simultaneously in separate files. For a recording file to be in compliance with this standard, it must contain as a minimum the following:

a. Computer Generated Packet(s), Setup Record Format 1 IAW paragraph 10.6.7.2 as the first packets in the recording.
b. Time Data Packet(s) IAW paragraph 10.6.3 as the first dynamic packet after the Computer Generated Packet, Setup Record.
c. One (1) or more data format packets IAW paragraph 10.6.

Multiple recordings may reside on the media, and each recording may contain one or more compliant files.

10.5.1.1 Data Hierarchy. The data hierarchy used to define the data stored according to this standard shall have the following structural relationships (highest to lowest):

a. Directory. One or more directory blocks of data comprising a list of all Data Files located under the guidance of this Standard. Also contains supporting data that may be of interest to those manipulating the Data Files. The list of files is made up from "File Entries." The Directory shall always start at logical address zero of each directory block.
b. Directory Block. A memory block containing file entries and other metadata.
c. Directory Block File Entry. A fixed length data structure used to describe files. It contains the name, the starting address, the number of blocks of data assigned to the Data File, the total number of bytes contained in the file, and the file's creation date and time. It also contains a reserved field for future growth and file close time.
d. Data Files. Data files are comprised of user data, presented at the interface in monotonically increasing contiguous logical addresses per file. Thus if a file starts at logical address X, the next location containing file data must be at the next logical address, X+1, and the next location after that must be at the next logical address, X+2, etc.

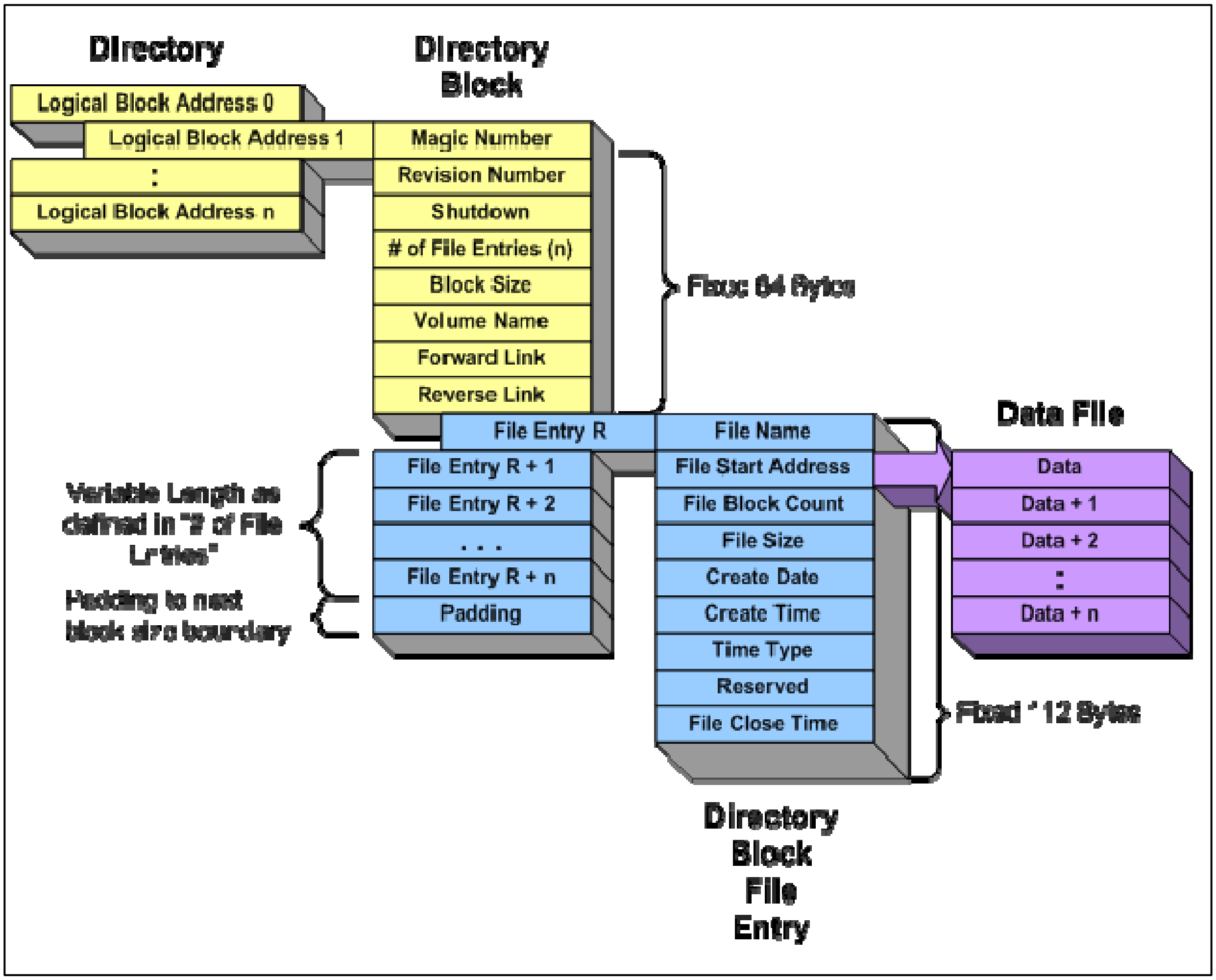


Figure 10-6.   Directory structure.

10.5.2 Directory Definition. The name and location information for all files recorded is illustrated in Figure 10-6). The Directory is composed of one or more directory blocks as shown in Figure 10-7. At least one Directory Block is required and it must be located at SCSI logical block address 1. Logical block address 0 is reserved.

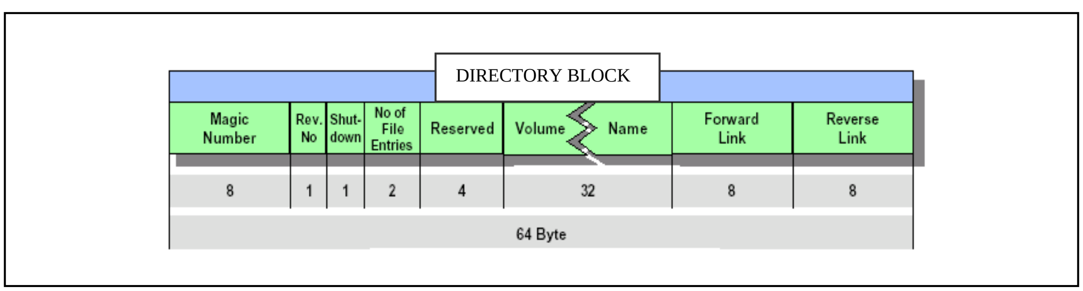


Figure 10-7.   Directory block.

a. Directory Fixed Fields. The fixed fields within a directory block are used to name the volume of data, identify the number of entries, and to provide pointers to other addresses that contain additional directory blocks. Forward and backward links to the next address for the next Directory Block (if any) or the preceding Directory Block (if any) allow for directory expansion beyond a single block. This does not limit the placement of directory information.

b. Block Size. The media types used to implement this standard have varying block lengths. Some will have blocks as small as 512 bytes; others may have blocks as large as 64K bytes or larger. The block size used by a given media can be determined via the SCSI Read Capacity Command (not defined here).

c. Directory to Data File Link. Each Data File on the media has a directory entry within a Directory Block that describes the file, as shown in Table 10-2. The directory entry for a Data File, as shown in Table 10-3, contains a link to the starting location of the data contained in each file and the total number of blocks assigned for the storage of data. This standard does not define the meaning of the data recorded within these Data File blocks.

TABLE 10-2. DIRECTORY BLOCK FORMAT

FIELD NAME	BYTES	DESCRIPTION	DATA TYPE
Magic Number	8	An identifier for a directory block. The value is BCS "FORTYtwo" (0x464F52545974776F)	BCS
Revision Number	1	Revision number of the standard compiled by the recording system. 0x01 = IRIG-106-03 thru IRIG-106-05 0x0F = IRIG-106-07	Unsigned Binary
Shutdown	1	Flag, if cleared to a 0x00 indicates that the volume was not properly dismounted, if seen on power-up is an indication that the directory chain may be faulty. If set = 0xFF, then the file system properly shutdown. This field is only valid in the first directory located in logical block 1; other directory blocks set to 0xFF.	Unsigned Binary
Number of File Entries	2	Defines the number of file entries that follow in this block.	Unsigned Binary
Block Size	4	Bytes per block size referenced in FileBlkCnt in Table-10-3.	Unsigned Binary
VolName	32	Volume name, see character set for restrictions. (Fill any unused VolName byte positions with 0x00.)	BCS
Forward Link	8	Block address of the next block containing directory information. Set equal to address of this block if this is the end of the chain.	Unsigned Binary
Reverse Link	8	Block address of the directory block pointing to this block. Set equal to the address of this block if this is the start of the chain.	Unsigned Binary
(n File Entries)	112 *n	One entry for each file specified in "Number of File Entries." The maximum value of "n" is dependent upon media block size.	See Table 10-3
Unused	Varies with n & block Size	It is possible for bytes to remain between the last byte of the last used File Entry and the end of the Directory Block. These bytes are defined to be Unused and should be filled with 0xFF.	Unsigned Binary
Note: 64 Bytes in fixed fields.			

TABLE 10-3. FILE ENTRY FORMAT			
FIELD NAME	**BYTES**	**DESCRIPTION**	**DATA TYPE**
Name	56	File name (see character set for restrictions). Fill any unused FileName Byte Positions with 0x00.	BCS
FileStartAdd	8	Zero based address of the first block reserved for data associated with this file. Fill with 0xFF for unused directory entries.	Unsigned Binary
FileBlkCnt	8	One based number that is the count of consecutive address blocks reserved for data for this file including the block pointed to by the FileStartAdd field.	Unsigned Binary
FileSize	8	The actual number of Bytes contained in this file. This file size will be equal to or less than the FileBlkCnt multiplied by the block size. This is an optional entry and will be filled with 0xFF if not used.	Unsigned Binary
File Create Date	8	DDMMYYYY BCS Character values, with no embedded spaces or other formatting characters, representing the numeric date on which the file was created (e.g., BCS codes for the decimal digits 02092000 → 0x3032303932303030 represents 2 September 2000). Fill with 0x2D if a value for the field is not available, or for portions of the field where data is not available.	BCS
File Create Time	8	HHMMSSss Character values, with no embedded spaces or other formatting characters, representing the numeric time at which the file was created. HH is the number of the 24 hour based hour, MM is the number of minutes after the hour, SS is the number of seconds after the minute, and ss is the hundredths of seconds after the second. Fill with 0x2D if a value for the field is not available, or for portions of the field where data is not available (e.g., "ss" is not available).	BCS
Time Type	1	A numeric code that qualifies the time and date values recorded in the "Create Date" and "Create Time" and "Close Time" fields. 0x0 = Coordinated Universal Time (Zulu) 0x1 = System Time 0x2 – 0xFE Reserved 0xFF = Time Data Packet	Unsigned Binary
Reserved	7	Bytes in this region are reserved for future growth. Fill with 0xFF.	Unsigned Binary
File Close Time	8	HHMMSSss Character values, with no embedded spaces or other formatting characters, representing the numeric time at which the file was closed. HH is the number of the 24 hour based hour, MM is the number of minutes after the hour, SS is the number of seconds after the minute, and ss is the hundredths of seconds after the second. Fill with 0x2D if a value for the field is not available, or for portions of the field where data is not available (e.g., "ss" is not available).	BCS
Note: 112 Bytes in fixed fields.			

d. File Entry Name. Each file entry in a directory shall have a unique name (See Naming Restrictions in paragraph 10.5.3.4). Default file name is a BCS numeric value incrementally increasing, starting at value "1."

e. File Entry Singularity. Multiple File entries are not permitted to refer to the same regions of memory, partially or completely.

f. Directory Entries and Fields. Directory block fields and entries shall be logically contiguous.

g. Directory and Memory Region Relationships. File Entries shall be entered sequentially into a Directory Block as files are recorded, starting with File Entry #1 in the Primary Directory Block (logical address 1). All File Entry positions in the Primary Directory Block shall be filled before the first Secondary Directory Block is used, and so on. However, there is no a priori relationship between the memory region associated with a file entry and the place-order of the file entry within the overall directory. For example, the very first file entry could refer to the very last logical address region of memory, the second file entry could refer to the beginning logical address of memory, and so on. Similarly, there is no presumed temporal ordering of file entries; the very last entry to be inserted could be inserted in such a fashion so as to be the first entry encountered when traversing the directory chain of blocks.

h. Empty Memory Reads. Reads of regions of memory not containing Directory Blocks or Data File blocks may return unpredictable data values or result in other error conditions.

i. Contiguous Directory Entries. File entries and all fields in a directory block are contiguous.

NOTE	Deleted Files are not applicable to IRIG-106 Chapter 10 as there are no recorder commands that allow or provide file deletion.

j. Deleted Files. In some applications, previously recorded files may be deleted in order to recover media space for new recordings. Deleted files shall be denoted by marking the corresponding file entry's File Block Count field with 0x00 indicating "unused." If the file block count has been set to 0x00, then other fields in that file entry are no longer meaningful.

k. Reserved Field. Reserved fields shall not be used in IRIG-106 Chapter 10 implementations and shall be filled with 0xFF. Reserved fields are intended for future IRIG-106 Chapter 10 use.

l. Number of File Entries. The numerical value placed in the "Number of File Entries" field of a Directory Block shall equal the number of active File Entries plus any File Entries marked as deleted files within that Directory Block.

10.5.3 Data Definitions.

10.5.3.1 Directory Byte Order. The directory structures described in paragraph 10.5 of this standard are defined to have the following bit and byte orientation. The most significant byte of any multi-byte structure is byte 0. The most significant bit of each byte is bit 0. This ordering is commonly referred to as “Big Endian.”

10.5.3.2 Data Format Byte Order. The data format structures (Packet Header, Secondary Packet Header, Channel Specific Data Word, Intra-Packet Header, and Packet Trailer) described in paragraph 10.6 of this standard are defined to have the following bit and byte orientation. The least significant byte shall be transmitted first, the least significant bit of each byte shall be transmitted first, and data is read from the lowest logical address first. This ordering is commonly referred to as “Little Endian.” The Packet data shall remain in its native byte order format.

10.5.3.3 Character Set. The character set for all character fields is based on ISO/IEC 10646-1, the Universal Multiple-Octet Coded Character Set (UCS). The NIIA limits characters to a subset rather than allowing all characters. The subset will be single octets, known as the Basic Character Set (BCS).

10.5.3.4 Naming Restrictions. The following rules shall be applied when forming names in order to assure the highest degree of interchange among other operating systems:

a. Characters. Characters from the first 127 common BCS characters (0x00 through 0x7E) may be used in names except for specific prohibited characters:

1) Any BCS character code value smaller than 0x20 is prohibited, except where the 0x00 is used to terminate the name.

2) The other prohibited characters with their hexadecimal representation are defined in Table 10-4.

TABLE 10-4. PROHIBITED CHARACTERS (HEXADECIMAL REPRESENTATION)

Forbidden Characters In Names	Hexadecimal Value	Forbidden Characters In Names	Hexadecimal Value
”	0x22	=	0x3D
‘	0x27	>	0x3E
*	0x2A	?	0x3F
/	0x2F	\	0x5C
:	0x3A	]	0x5D
;	0x3B	[	0x5B
<	0x3C	|	0x7C

b. Names. Names used for this interface will observe the following rules:

(1) Upper and lowercase characters are considered to be different within file names.

(2) Leading and trailing spaces are not permitted.

(3) Leading periods are not permitted.

(4) Names shall fill their field starting with byte 0 per paragraph 10.5.3.1 and be terminated with a 0x00. Unused Name characters shall be filled with 0x00. Names may utilize the full length of the field, in which case the terminating 0x00 must be omitted. Examples of host provided and default file names are shown in Table 10-5.

TABLE 10-5. FILE NAME EXAMPLES

File Name Byte Address	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	//	50	51	52	53	54	55
Host Provided File Name Example	R	E	C	O	R	D	I	N	G	1	S	E	N	S	O	R	2	0X 00	0X 00	//	0X 00	0X 00	0X 00	0X 00	0X 00	0X 00
Default File Name Example	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	// 0	0	0	0	0	0	1

10.6 Data Format Definition

10.6.1 Common Packet Elements. Data shall have three required parts, a Packet Header, a Packet Body, a Packet Trailer, and an optional part if enabled, a Packet Secondary Header. Single or multiple channel recordings will always conform to the structure outlined in Figure 10-8.

a. A packet has the basic structure shown in Figure 10-9. Note that the width of the structure is not related to any number of bytes or bits. This table is merely to represent relative packet elements and their placement within the packet. See Figure 10-10 for a diagram of the generic packet format. This figure does not depict the bit lengths of each field. Word sizes of 8 bits, 16 bits, and 32 bits are used depending on the data type.

To further clarify the packet layout, Figure 10-10 shows the generic packet in a 32-bit, Little-endian format, and assumes 16-bit data words and data checksum.

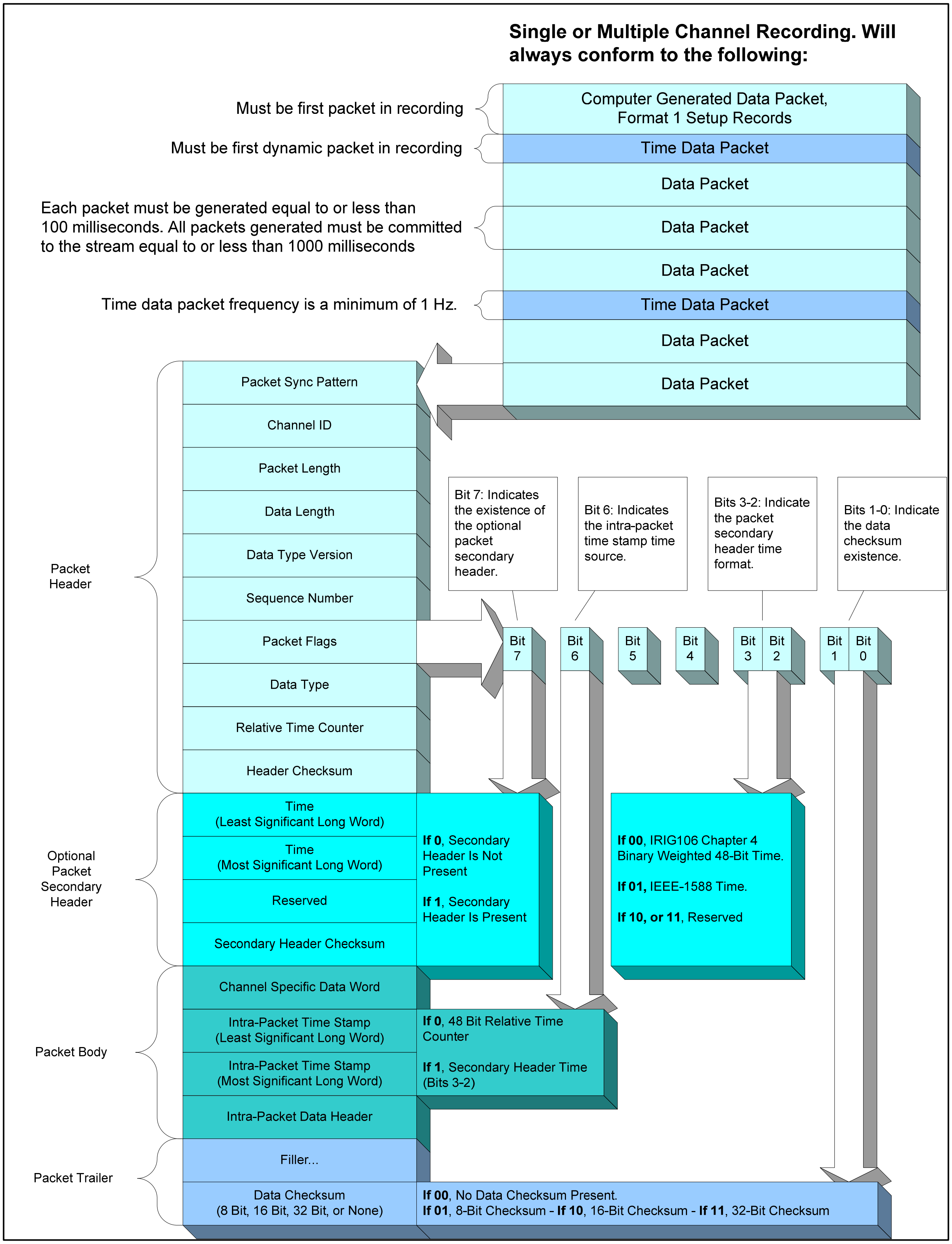


Figure 10-8. Data recording structure.

PACKET SYNC PATTERN	Packet Header
CHANNEL ID	
PACKET LENGTH	
DATA LENGTH	
DATA TYPE VERSION	
SEQUENCE NUMBER	
PACKET FLAGS	
DATA TYPE	
RELATIVE TIME COUNTER	
HEADER CHECKSUM	
TIME	Packet Secondary Header (Optional)
RESERVED	
SECONDARY HEADER CHECKSUM	
CHANNEL SPECIFIC DATA	Packet Body
INTRA-PACKET TIME STAMP 1	
INTRA-PACKET DATA HEADER 1	
DATA 1	
:	
INTRA-PACKET TIME STAMP n	
INTRA-PACKET DATA HEADER n	
DATA n	
DATA CHECKSUM	Packet Trailer

Figure 10-9. General packet format.

<table>
<tr><td colspan="2">msb
31 16</td><td colspan="2">lsb
15 0</td><td></td></tr>
<tr><td colspan="2">CHANNEL ID</td><td colspan="2">PACKET SYNC PATTERN</td><td rowspan="6">Packet Header</td></tr>
<tr><td colspan="4">PACKET LENGTH</td></tr>
<tr><td colspan="4">DATA LENGTH</td></tr>
<tr><td>DATA TYPE</td><td>PACKET FLAGS</td><td>SEQUENCE NUMBER</td><td>DATA TYPE VERSION</td></tr>
<tr><td colspan="4">RELATIVE TIME COUNTER</td></tr>
<tr><td colspan="2">HEADER CHECKSUM</td><td colspan="2">RELATIVE TIME COUNTER</td></tr>
<tr><td colspan="4">TIME (LSLW)</td><td rowspan="3">(Optional) Packet Secondary Header</td></tr>
<tr><td colspan="4">TIME (MSLW)</td></tr>
<tr><td colspan="2">SECONDARY HEADER CHECKSUM</td><td colspan="2">RESERVED</td></tr>
<tr><td colspan="4">CHANNEL SPECIFIC DATA</td><td rowspan="17">Packet Body</td></tr>
<tr><td colspan="4">INTRA-PACKET TIME STAMP 1</td></tr>
<tr><td colspan="4">INTRA-PACKET TIME STAMP 1</td></tr>
<tr><td colspan="4">INTRA-PACKET DATA HEADER 1</td></tr>
<tr><td colspan="2">DATA 1 WORD 2</td><td colspan="2">DATA 1 WORD 1</td></tr>
<tr><td colspan="2">DATA 1 WORD n</td><td colspan="2">:</td></tr>
<tr><td colspan="4">INTRA-PACKET TIME STAMP 2</td></tr>
<tr><td colspan="4">INTRA-PACKET TIME STAMP 2</td></tr>
<tr><td colspan="4">INTRA-PACKET DATA HEADER 2</td></tr>
<tr><td colspan="2">DATA 2 WORD 2</td><td colspan="2">DATA 2 WORD 1</td></tr>
<tr><td colspan="2">DATA 2 WORD n</td><td colspan="2">:</td></tr>
<tr><td colspan="4">:</td></tr>
<tr><td colspan="4">INTRA-PACKET TIME STAMP N</td></tr>
<tr><td colspan="4">INTRA-PACKET TIME STAMP N</td></tr>
<tr><td colspan="4">INTRA-PACKET DATA HEADER N</td></tr>
<tr><td colspan="2">DATA N WORD 2</td><td colspan="2">DATA N WORD 1</td></tr>
<tr><td colspan="2">DATA N WORD n</td><td colspan="2">:</td></tr>
<tr><td colspan="4">[FILLER]</td><td rowspan="2">Packet Trailer</td></tr>
<tr><td colspan="2">DATA CHECKSUM</td><td colspan="2"></td></tr>
</table>

Figure 10-10. A 32-bit packet format layout.

Depending on the data type, the size of the Data Checksum can contain 16 bits, 32 bits, 8 bits, or the checksum can be entirely left out. For a 32-bit Data Checksum, the packet trailer would be as shown in Figure 10-11.

msb 7 lsb 0	
[FILLER]	Packet Trailer
DATA CHECKSUM (LSB)	
DATA CHECKSUM	
DATA CHECKSUM	
DATA CHECKSUM (MSB)	

Figure 10-11. Packet trailer for 32-bit data checksum.

b. For an 8-bit Data Checksum, the packet trailer would be as shown in Figure 10-12.

msb 7 lsb 0	
[FILLER]	Packet Trailer
DATA CHECKSUM	

Figure 10-12. Packet trailer for 8-bit data checksum.

c. The size of a single Packet may be a maximum of 524,288 bytes as shown in Table 10-6. This includes the Packet Header, Packet Body, Packet Trailer, and optional Packet Secondary Header if enabled. The only exception to the packet size limit is the Computer Generated Data Packet, Format 1 Setup Record, which may be a maximum of 134,217,728 bytes. Any Packet that requires more than 524,288 bytes may generate multiple packets by utilizing the packet sequence counter. Some packet types allow a single data set to span multiple packets if the data set size or time does not fall under packet maximums. The specific mechanism allowing packet data spanning for each data type is described within that data type's section.

d. With the exception of Computer Generated Packets, all other Packet Generation Times shall be equal to or less than 100 milliseconds as measured by the 10 MHz Relative Time Counter whenever data is available. This requirement ensures that a packet shall contain equal to or less than 100 milliseconds worth of data, and that a packet containing any data must be generated equal to or less than 100 milliseconds from the time the first data was placed in the packet. This strategy will assure packet granularity and save bandwidth by not forcing or marking empty/idle packets.

e. Packets *can not* contain only filler or *can not* be idle or empty. All packets that are generated *shall contain data*.

f. All reserved bit-fields in packet headers or channel specific data words shall be set to zero (0x0).

g. With the exception of Computer Generated Data Packets, all other packets shall have a Stream Commit Time equal to or less than 1,000 milliseconds as measured by the 10 MHz Relative Time Counter contained in the packet header.

h. Once version bits and packet structure bits have been used to indicate a value or setting for each data type and its associated channel, they shall not change for that data type and its associated channel within the recording.

TABLE 10-6. PACKET REQUIREMENTS

PACKET TYPE	REQUIRED	MAXIMUM PACKET SIZE	REQUIRED PACKET LOCATION
Computer Generated Data Packet, Format 1 Setup Record	Yes	134,217,728 bytes	First Packets in Recording. A single setup record may span across multiple Computer Generated Data Packets, Format 1 Setup Record.
Time Data Packet	Yes	524,288 bytes	First Dynamic Data Packet Following Setup Record Packet(s). Reference the Time Data Packet Description for packet rate.
All other data type packets with the exception of Computer Generated Data Packet, Format 1 Setup Record, Time Data Packets, and Computer Generated Data Packet, Format 3 Recording Index (Root Index)	No	524,288 bytes	After First Time Data Packet and before the last Computer Generated Data Packet Format 3, Recording Index (Root Index) if enabled.
Computer Generated Data Packet, Format 3 Recording Index (Root Index)	Yes, if Recording Events are Enabled. No, if Recording Events are Disabled.	524,288 bytes	If Recording Index Packets are enabled, Root Index Packet Type will be the last packet in a recording.

10.6.1.1 Packet Header. The length of the packet header is fixed at 24 bytes (192 bits). The Packet Header is mandatory and shall consist of ten fields, positioned contiguously, in the following sequence:

a. Packet Sync Pattern. (2 Bytes) contains a static sync value for the every packet. The Packet Sync Pattern value shall be 0xEB25.

b. Channel ID. (2 Bytes) contains a value representing the Packet Channel ID. All channels in a system must have a unique Channel ID for each data source.
(1) Multiplexer Source ID. In a distributed Multiplexer system, a Multiplexer Source ID is used to discern each multiplexer in the system. The Setup Record shall contain a "Number of Source Bits" recorder attribute (R-x\NSB) to specify the number of most significant bits (from the Channel ID) that distinguish the Multiplexer Source ID. The remaining least significant bits of the Channel ID field shall be the Channel ID for each data source acquired by the multiplexer.
(2) Reserved Channel ID. Channel ID 0x0000 is reserved, and is used to insert computer-generated messages into the composite data stream.
(3) Available Channel IDs. All values not comprising the reserved Channel ID are available.

NOTE	For single multiplexer systems: the reserved Channel ID is 0x0000; values 0x0001 thru 0xFFFF are available Channel IDs; Computer Generated Data Packets shall only have a Channel ID of 0x0000; and no other data type can have a Channel ID of 0x0000.

c. Packet Length. (4 Bytes) contains a value representing the length of the entire packet. The value shall be in bytes and is always a multiple of four (bit 1 and bit 0 shall always be zero). This Packet Length includes the Packet Header, Packet Secondary Header (if enabled), Channel Specific Data, Intra-Packet Headers, Data, Filler, and Data Checksum.

d. Data Length. (4 Bytes) contains a value representing the valid data length within the packet. This value shall be represented in bytes. Valid data length includes Channel Specific Data, Intra-Packet Data Headers, Intra-Packet Time Stamp(s), and Data but does not include Packet Trailer Filler and Data Checksum.

e. Data Type Version. (1 Byte) contains a value at or below the release version of standard applied to the data types in table 10-7. The value shall be represented by the following bit patterns:

0x00 = Reserved
0x01 = Initial Release (IRIG-106-04)
0x02 = IRIG-106-05
0x03 = IRIG-106-07
0x04 = IRIG-106-09
0x05 thru 0xFF = Reserved

f. Sequence Number. (1 Byte) contains a value representing the packet sequence number for each Channel ID. This is simply a counter that increments by n + 0x01 to 0xFF for every packet transferred from a particular channel and is not required to start at 0x00 for the first occurrence of a packet for the Channel ID.

NOTE	Sequence number counter value for each channel in a recording will repeat (rollover to 0x00) after the sequence number counter has reached 0xFF.

NOTE	Each Channel in a Recording shall have its own sequence counter providing a unique Sequence Number for that channel.

g. Packet Flags. (1 Byte) contains bits representing information on the content and format of the packet(s).

Bit 7: Indicates the presence or absence of the Packet Secondary Header.
- 0 = Packet Secondary Header is not present.
- 1 = Packet Secondary Header is present.

Bit 6: Indicates the Intra-Packet Time Stamp Time Source.
- 0 = Packet Header 48-bit Relative Time Counter.
- 1 = Packet Secondary Header Time (bit 7 must be 1).

Bit 5: Relative Time Counter Sync Error.
- 0 = No Relative Time Counter sync error.
- 1 = Relative Time Counter sync error has occurred.

Bit 4: Indicates the Data Overflow Error.
- 0 = No data overflow.
- 1 = Data overflow has occurred.

Bits 3-2: Indicate the Packet Secondary Header Time Format.
- 00 = IRIG 106 Chapter 4 binary weighted 48-bit time format. The two LSBs of the 64-bit Packet Secondary Header Time and Intra-Packet Time Stamp shall be zero filled.
- 01 = IEEE-1588 Time format. The Packet Secondary Header Time and each Intra-Packet Time Stamp shall contain a 64-bit timestamp represented in accordance with the Time Representation type as specified by IEEE STD 1588-2002. The 32 bits indicating seconds shall be placed into the Most Significant Long Word portion of the secondary header and the

32 bits indicating nanoseconds shall be placed into the Least Significant Long Word portion.
10 = Reserved
11 = Reserved

Bits 1-0: Indicate Data Checksum existence.
00 = No data checksum present
01 = 8-bit data checksum present
10 = 16-bit data checksum present
11 = 32-bit data checksum present

h. Data Type. (1 Byte) contains a value representing the type and format of the data. All values not used to define a data type are reserved for future data type growth.

TABLE 10-7. DATA TYPE NAMES AND DESCRIPTIONS

PACKET HEADER VALUE	DATA TYPE NAME	DATA TYPE DESCRIPTION	CURRENT DATA TYPE VERSION
0x00	Computer Generated Data, Format 0	(User Defined)	0x02
0x01	Computer Generated Data, Format 1	(Setup Record)	0x03
0x02	Computer Generated Data, Format 2	(Recording Events)	0x03
0x03	Computer Generated Data, Format 3	(Recording Index)	0x03
0x04 – 0x07	Computer Generated Data, Format 4 Format 7	(Reserved for future use)	0x02
0x08	PCM Data, Format 0	(Reserved for future use)	0x02
0x09	PCM Data, Format 1	(IRIG 106 Chapter 4/8)	0x03
0x0A – 0x0F	PCM Data, Format 2 – Format 7	(Reserved for future use)	0x02
0x10	Time Data, Format 0	(Reserved for future use)	0x02
0x11	Time Data, Format 1	(IRIG/GPS/RTC)	0x03
0x12 – 0x17	Time Data, Format 2 – Format 7	(Reserved for future use)	0x02
0x18	MIL-STD-1553 Data, Format 0	(Reserved for future use)	0x02
0x19	MIL-STD-1553 Data, Format 1	(Mil-Std-1553B Data)	0x03
0x1A	MIL-STD-1553 Data, Format 2	(16PP194 Bus)	0x03
0x1B – 0x1F	MIL-STD-1553 Data, Format 3 – Format 7	(Reserved for future use)	0x03
0x20	Analog Data, Format 0	(Reserved for future use)	0x02
0x21	Analog Data, Format 1	(Analog Data)	0x02
0x22 – 0x27	Analog Data, Format 2 – Format 7	(Reserved for future use)	0x02
0x28	Discrete Data, Format 0	(Reserved for future use)	0x02
0x29	Discrete Data, Format 1	(Discrete Data)	0x02
0x2A – 0x2F	Discrete Data, Format 2 – Format 7	(Reserved for future use)	0x02

TABLE 10-7. DATA TYPE NAMES AND DESCRIPTIONS			
PACKET HEADER VALUE	**DATA TYPE NAME**	**DATA TYPE DESCRIPTION**	**CURRENT DATA TYPE VERSION**
0x30	Message Data, Format 0	(Generic Message Data)	0x02
0x31 – 0x37	Message Data, Format 1 – Format 7	(Reserved for future use)	0x02
0x38	ARINC 429 Data, Format 0	(ARINC429 Data)	0x02
0x39– 0x3F	ARINC 429 Data, Format 1 – Format 7	(Reserved for future use)	0x02
0x40	Video Data, Format 0	(MPEG-2/H.264 Video)	0x04
0x41	Video Data, Format 1	(ISO 13818-1 MPEG-2)	0x02
0x42	Video Data, Format 2	(ISO 14496 MPEG-4 Part10) 110 AVC/H.264)	0x03
0x43 – 0x47	Video Data, Format 3 – Format 7	(Reserved for future use)	0x03
0x48	Image Data, Format 0	(Image Data)	0x02
0x49	Image Data, Format 1	(Still Imagery)	0x03
0x4A – 0x4F	Image Data, Format 2 – Format 7	(Reserved for future use)	0x03
0x50	UART Data, Format 0	(UART Data)	0x03
0x51 – 0x57	UART Data, Format 1 – Format 7	(Reserved for future use)	0x02
0x58	IEEE-1394 Data, Format 0	(IEEE-1394 Transaction)	0x02
0x59	IEEE-1394 Data, Format 1	(IEEE-1394 Physical Layer)	0x03
0x5A – 0x5F	IEEE-1394 Data, Format 2 – Format 7	(Reserved for future use)	0x03
0x60	Parallel Data, Format 0	(Parallel Data)	0x02
0x61 – 0x67	Parallel Data, Format 1 – Format 7	(Reserved for future use)	0x02
0x68	Ethernet Data, Format 0	(Ethernet Data)	0x03
0x69 – 0x6F	Ethernet Data, Format 1 – Format 7	(Reserved for future use)	0x03

i. Relative Time Counter. (6 Bytes) contains a value representing the 10 MHz Relative Time Counter (RTC). This is a free-running 10 MHz binary counter represented by 48 bits which are common to all data channels. The counter shall be derived from a 10 MHz internal crystal oscillator and shall remain free-running during each recording.

   Note: The applicable data bit to which the 48-bit value applies shall correspond to the first bit of the data in the packet body (unless it is defined in each data type section).

j. Header Checksum. (2 Bytes) contains a value representing a 16-bit arithmetic sum of all 16-bit words in the header excluding the Header Checksum Word.

10.6.1.2 Packet Secondary Header (Optional). The length of the Packet Secondary Header is fixed at 12 bytes (96 bits). The Packet Secondary Header is optional and when enabled shall consist of the three fields, positioned contiguously, in the following sequence:

a. Time. (8 Bytes) contain the value representing Time in the format indicated by bits 2 and 3 of the Packet Flags in paragraph 10.6.1.1g.
b. Reserved. (2 Bytes) are reserved and shall be zero filled.
c. Secondary Header Checksum. (2 Bytes) contain a value representing a 16-bit arithmetic sum of all Secondary Header bytes excluding the Secondary Header Checksum Word.

10.6.1.3 Packet Body. The format of the data in the packet body is unique to each data type. Detailed descriptions of the type-specific data formats found in packet bodies are described in subsequent sections of this document.

a. Channel Specific Data. (Variable Bytes) contains the contents of the Channel Specific Data field(s) depending on the Data Type field in the Packet Header. Channel Specific Data is mandatory for each data type and channel. The occurrence of Channel Specific Data is once per packet and precedes packet channel data.

b. Intra-Packet Time Stamp. (8 Bytes) contains Time in either 48-bit Relative Time Counter format (plus 16 high-order zero bits) or 64-bit absolute format as specified in the Packet Flags in the Packet Header. The Intra-Packet Time Stamps are only mandatory where defined by the data formats.

c. Intra-Packet Data Header. (Variable Bytes) contains additional status and format information pertaining to the data items that follow. The Intra-packet Data Headers are only mandatory where defined by the data formats.

d. Data. (n Bytes) contains valid data from a particular channel as defined within the data formats contained within this standard.

NOTE	The Intra-Packet Time Stamp and the Intra-Packet Data Header are collectively called the Intra-Packet Header. In some cases, an Intra-Packet Header may only have a Time Stamp (zero-length Data Header), while in other cases, the Intra-Packet Header only has a Data Header (zero-length Time Stamp). Some data types have no Intra-Packet Header. The Intra-Packet Header requirements are specified separately for each Data Type.

NOTE	The Intra-Packet Data Header (IPDH) presence, once set, shall be the same state for the entire recording per each channel

10.6.1.4 Packet Trailer. The packet trailer may contain filler, a data checksum, both filler and a data checksum, or neither filler nor a data checksum. In the latter case, the packet trailer has zero length. The reason a packet trailer would have a zero length is best explained by understanding the reason for inserting filler. The purpose of the filler is twofold:

a. To keep all packets aligned on 32-bit boundaries (i.e., make all packet lengths a multiple of 4 bytes), and
b. To optionally keep all packets from a particular channel the same length.

If both of the above requirements are already met without adding filler, then filler shall not be added.

The inclusion of the data checksum is optional as well and is indicated by the Packet Flags setting. When included, the packet trailer contains either an 8-bit, 16-bit, or 32-bit Data Checksum. Depending on the Packet Flags option selected, the Data Checksum is the arithmetic sum of all of the bytes (8 bits), words (16 bits), or long words (32 bits) in the packet excluding the 24 bytes of Packet Header, Packet Secondary Header (if enabled) and the Data Checksum. Stated another way, the Data Checksum includes everything in the packet body plus all added filler.

a. Filler. (variable Bytes) All filler shall be set to 0x00 or 0xFF.
b. 8-Bit Data Checksum. (1 Byte) contains a value representing an 8-bit arithmetic sum of the bytes in the packet. Only inserted if Packet Flag bits are set (see paragraph 10.6.1.1.g).
c. 16-Bit Data Checksum. (2 Bytes) contains a value representing a 16-bit arithmetic sum of the words in the packet. Only inserted if Packet Flag bits are set (paragraph 10.6.1.1.g).
d. 32-Bit Data Checksum. (4 Bytes) contains a value representing a 32-bit arithmetic sum of the long words in the packet and is only inserted if Packet Flag bits are set (paragraph 10.6.1.1.g).

10.6.2 PCM Data Packets.

10.6.2.1 PCM Data Packets Format 0. Reserved.

10.6.2.2 PCM Data Packets Format 1 (IRIG 106 Chapter 4 and 8). A packet with IRIG-106 Chapter 4 or IRIG-106 Chapter 8 PCM data has the basic structure as shown in Figure 10-13. Note that the width of the structure is not related to any number of bits. This table merely represents relative placement of data in the packet.

PACKET HEADER
CHANNEL SPECIFIC DATA
(Optional) INTRA-PACKET TIME STAMP
(Optional) INTRA-PACKET DATA HEADER
MINOR FRAME DATA
(Optional) INTRA-PACKET TIME STAMP
(Optional) INTRA-PACKET DATA HEADER
MINOR FRAME DATA
(Optional) INTRA-PACKET TIME STAMP
(Optional) INTRA-PACKET DATA HEADER
MINOR FRAME DATA
(Optional) INTRA-PACKET TIME STAMP
(Optional) INTRA-PACKET DATA HEADER
MINOR FRAME DATA
:
(Optional) INTRA-PACKET TIME STAMP
(Optional) INTRA-PACKET DATA HEADER
MINOR FRAME DATA
PACKET TRAILER

Figure 10-13. General PCM data packet, format 1.

The user may separately enable or disable word unpacking on each active PCM channel. Word unpacking will force the least significant bit of each word to be aligned on a 16-bit boundary. High-order filler bits are added to words as necessary to force alignment.

The user may separately enable or disable frame synchronizing on each active PCM channel. This provides a "Throughput Mode" that will transfer data to the packet without frame synchronization. Throughput Mode essentially disables all setup and packing/unpacking options for the packet, and places data in the packet as it is received.

a. PCM Packet Channel Specific Data. The packet body portion of each PCM packet begins with the channel specific data, which is formatted as shown in Figure 10-14.

msb 31	30	29	28	27 24	23 18	17 lsb 0
R	IPH	MA	MI	LOCKST	MODE	SYNCOFFSET

Figure 10-14. PCM packet channel specific data format.

- Reserved. (bit 31) is reserved.
- Intra-Packet Header (IPH). (bit 30) indicates if Intra-Packet Headers (Intra-Packet Time Stamp and Intra-Packet Data Header) are inserted before each minor frame. Intra-Packet Headers are only optional because of the mode selection. This determines whether Intra-Packet Headers are included or omitted.
  0 = Intra-Packet Headers are omitted for Throughput Mode.
  1 = Intra-Packet Headers are required for Packed Data and Unpacked Data modes.
- Major Frame Indicator (MA). (bit 29) indicates if the first word in the packet is the beginning of a major frame. Not valid for Throughput Mode.
  0 = First word is not the beginning of a major frame.
  1 = First word is the beginning of a major frame.
- Minor Frame Indicator (MI). (bit 28) indicates if the first word in the packet is the beginning of a minor frame. Not valid for Throughput Mode.
  0 = First word is not the beginning of a minor frame.
  1 = First word is the beginning of a minor frame.
- Lock Status (LOCKST). (bits 27-24) indicates the lock status of the frame synchronizer. Not valid for Throughput Mode.

NOTE	Minor Frame Definition. The minor frame is defined as the data structure in time sequence from the beginning of a minor frame synchronization pattern to the beginning of the next minor frame synchronization pattern. Please reference IRIG-106 Chapter 4, paragraph 4.3.2 for minor/major frame definition.

Bits 27-26: Indicate Minor Frame Status.
00 = Reserved.
01 = Reserved.
10 = Minor Frame Check (after losing Lock).
11 = Minor Frame Lock.

Bits 25-24: Indicate Major Frame Status.
00 = Major Frame Not Locked.
01 = Reserved.
10 = Major Frame Check (after losing Lock).
11 = Major Frame Lock.

- Mode (MODE). (bits 23-18) indicates the data packing mode.

Bits 23-22: Reserved.

Bit 21: Alignment Mode.
0 = 16 Bit Alignment Mode enabled.
1 = 32 Bit Alignment Mode enabled.

Bit 20: Indicates Throughput Data Mode.
0 = Throughput Data Mode not enabled.
1 = Throughput Data Mode enabled.

Bit 19: Indicates Packed Data Mode.
0 = Packed Data Mode not enabled.
1 = Packed Data Mode enabled.

Bit 18: Indicates Unpacked Data Mode.
0 = Unpacked Data Mode not enabled.
1 = Unpacked Data Mode enabled.

- Sync Offset (SYNCOFFSET). (bits 17-0) contains an 18-bit binary value representing the Word offset into the major frame for the first data word in the packet. Not valid for Packed or Throughput Mode.

b. PCM Packet Body. After the Channel Specific Data, the Intra-Packet Headers and the PCM data are inserted in the packet in integral numbers of minor or major frames, unless the packet is in Throughput Mode. In Throughput Mode, there is no frame or word alignment to the packet data and no Intra-Packet Headers are inserted in the data. In both Packed and Unpacked Modes, minor frame alignment is dependent on the MODE field in the Channel Specific Data. In 16 Bit Alignment Mode, PCM minor frames begin and end on 16-bit boundaries. In 32 Bit Alignment Mode, PCM minor frames begin and end on 32-bit boundaries. In either case, Alignment Mode does not affect the format of PCM data words themselves. However, depending on perspective, word order is affected and a zero-filled data word may be required to maintain alignment.

c. PCM Data in Unpacked Mode. In Unpacked Mode, packing is disabled and each data word is padded with the number of filler bits necessary to align the first bit of each word with the next 16-bit boundary in the packet. For example, 4 pad bits are added to 12-bit words, 6 pad bits are added to 10-bit words, etc. In 32 Bit Alignment Mode, a zero-filled 16-bit word is required to maintain alignment when an odd number of 16-bit words exist in the minor frame.

Minor frame sync patterns larger than 16 bits are divided into two words of packet data. If the sync pattern has an even number of bits, then it will be divided in half and placed in two packet words. For example, a 24-bit sync pattern is broken into two (2) 12-bit words with 4 bits of pad in each word. If the sync pattern has an odd number of bits, it is broken into two (2) words with the second word having one-bit more of the sync pattern. For example, if the minor sync pattern is 25 bits, then the first sync word is 12 bits of sync pattern plus 4 bits of pad, and the second sync word is 13 bits of sync pattern plus 3 bits of pad.

Minor frame sync patterns larger than 32 bits are divided into (Number of bits+15) / 16 words in 16 Bit Alignment Mode, or (Number of bits+31) / 32 in 32 Bit Alignment Mode. If the sync word doesn't fill the words completely, the first word shall contain the lesser number of bits with the later words containing one bit more (in the manner described above in splitting frame sync pattern words into two words). For example, a 35-bit sync word shall be split into 11+12+12-bit words in 16 Bit Alignment Mode, or into 17+18-bit words in 32 Bit Alignment Mode.

Given PCM frames with a 24-bit minor sync pattern and n data words where the bit-lengths of data words 1, 2, and 3 are 12, 16, and 8 respectively, the resultant PCM packets are as shown in Figure 10-15.

d. PCM Data in Packed Mode. In Packed Mode, packing is enabled and pad is not added to each data word. However, filler bits may be required to maintain minor frame alignment. The number of filler bits is dependent on the Alignment Mode, where N is either 16 or 32. If the number of bits in the minor frame is not an integer multiple of N, then Y PAD bits will be added to the end of each minor frame of bit-length L. Either Y = N – MOD(L,N), or N minus the integer remainder when L is divided by N. In packed mode, the PCM stream is minor frame synchronized so the first data bit in the packet is the first data bit of a minor frame. If X = N – Y when N is 32 Bit Alignment Mode, then the resultant PCM packets are as shown in Figure 10-16.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
INTRA-PACKET TIME STAMP (BITS 15-0)	
INTRA-PACKET TIME STAMP (BITS 31-16)	
INTRA-PACKET TIME STAMP (BITS 47-32)	
INTRA-PACKET TIME STAMP (BITS 63-48)	
INTRA-PACKET DATA HEADER (BITS 15-0)	
INTRA-PACKET DATA HEADER (BITS 31-16) (32 Bit Alignment Mode ONLY)	
4-BITS PAD	12-BITS SYNC (BITS 23-12)
4-BITS PAD	12-BITS SYNC (BITS 11-0)
4-BITS PAD	12-BITS WORD 1 DATA
16-BITS WORD 2 DATA	
8-BITS PAD	8-BITS WORD 3 DATA
:	
WORD n DATA BITS + PAD IF NEEDED	
ZERO FILLED PAD WORD IF NEEDED (32 Bit Alignment Mode ONLY)	
INTRA-PACKET TIME STAMP (BITS 15-0)	
INTRA-PACKET TIME STAMP (BITS 31-16)	
INTRA-PACKET TIME STAMP (BITS 47-32)	
INTRA-PACKET TIME STAMP (BITS 63-48)	
INTRA-PACKET DATA HEADER (BITS 15-0)	
INTRA-PACKET DATA HEADER (BITS 31-16)	
:	
REPEAT FOR EACH MINOR FRAME	
:	
PACKET TRAILER	

Figure 10-15. PCM Data – unpacked mode sample packet.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
INTRA-PACKET TIME STAMP (BITS 15-0)	
INTRA-PACKET TIME STAMP (BITS 31-16)	
INTRA-PACKET TIME STAMP (BITS 47-32)	
INTRA-PACKET TIME STAMP (BITS 63-48)	
INTRA-PACKET DATA HEADER (BITS 15-0)	
DATA (BITS 15 – 0)	
DATA (BITS 31 – 16)	
DATA (BITS 47 – 32)	
:	
Y FILLER BITS	
Y FILLER BITS	X DATA BITS
INTRA-PACKET TIME STAMP (BITS 15-0)	
INTRA-PACKET TIME STAMP (BITS 31-16)	
INTRA-PACKET TIME STAMP (BITS 47-32)	
INTRA-PACKET TIME STAMP (BITS 63-48)	
INTRA-PACKET DATA HEADER (BITS 15-0)	
:	
REPEAT FOR EACH MINOR FRAME	
:	
PACKET TRAILER	

Figure 10-16. PCM Data – packed mode sample packet.

e. PCM Data in Throughput Mode. In Throughput Mode, the PCM data are not frame synchronized so the first data bit in the packet can be any bit in the major frame. The resultant PCM packets are as shown in Figure 10-17. Only bit 20 of the Channel Specific Data word is set to one (1), indicating Throughput Mode. All other bits of the Channel Specific Data word are undefined and shall be set to zero (0).

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
DATA (BITS 15 – 0)	
DATA (BITS 31 – 16)	
DATA (BITS 47 – 32)	
:	
PACKET TRAILER	

Figure 10-17. PCM Data – Throughput Mode sample packet.

f. PCM Data Word Order in 32 Bit Alignment Mode. When recording in 32 Bit Alignment Mode, the resultant data word ordering will differ from 16 Bit Alignment Mode. The serial PCM data stream is shifted into 32-bit words from right to left, with bit 31 on the left, bit 0 on the right, and addresses ascending from top to bottom. Word order is affected depending on the reader's addressing perspective. For example, 16-bit data words when addressed as 32-bit words appear in order when read from left to right, and top to bottom. However, when addressed as 16-bit words, each pair of data words will appear swapped. Figure 10-18 and Figure 10-19 depict the anomaly of perspective.

msb 31	16	15	lsb 0	addr
byte 3	byte 2	byte 1	byte 0	
DATA WORD 1		DATA WORD 2		0
DATA WORD 3		DATA WORD 4		1
:				
DATA WORD N-1		DATA WORD N		N/2-1

Figure 10-18. 32-bit Alignment Mode example, 16-bit data words (32-bit word addressing).

msb 15	lsb 0	addr
byte 1	byte 0	
DATA WORD 2		0
DATA WORD 1		1
DATA WORD 4		2
DATA WORD 3		3
:		:
DATA WORD N-1		N-1

Figure 10-19. 32-bit Alignment Mode example, 16-bit data words (16-bit word addressing).

g. PCM Intra-Packet Header. When recording in Packed or Unpacked mode, all PCM minor frames shall include an Intra-Packet Header containing a 64-bit Intra-Packet Time Stamp and a 16 or 32-bit Intra-Packet Data Header, as indicated by MODE in the Channel Specific Data. This header is inserted immediately before the minor frame sync pattern. Depending on Alignment Mode, the length of the Intra-Packet Header is either 10 or 12 bytes (80 or 96 bits) positioned contiguously, as depicted in Figure 10-20. In 16 Bit Alignment Mode, the Intra-Packet Data Header length is fixed at 2 bytes. A 32 Bit Alignment Mode requires a 4 byte Intra-Packet Data Header, and the two most significant bytes are zero-filled.

msb 31	16	15	12	11	lsb 0
TIME (LSLW)					
TIME (MSLW)					
zero filled		LOCKST		RESERVED	

Figure 10-20. PCM Intra-packet header.

- Intra-Packet Time Stamp. (8 Bytes) indicates the time tag of the PCM minor frame. Not valid for Throughput Mode. First Long Word Bits and Second Long Word Bits indicate the following values:
  - The 48-bit Relative Time Counter that corresponds to the first data bit of the minor frame with bits 31 to 16 in the second long word zero filled; or
  - Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format corresponds to the time format indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and to the first data bit of the minor frame.

- Intra-Packet Data Header.
  - 32 Bit Alignment (32-Bit Alignment Mode ONLY). Bits 31-16 are zero filled.
  - Lock Status (LOCKST). (bits 15-12) indicates the lock status of the frame synchronizer for each minor frame.

  Bits 15-14: Indicates Minor Frame Status.
  00 = Reserved.
  01 = Reserved.
  10 = Minor Frame Check (after losing Lock).
  11 = Minor Frame Lock.

  Bits 13-12: Indicates Major Frame Status.
  00 = Major Frame Not Locked.
  01 = Reserved.
  10 = Major Frame Check (after losing Lock).
  11 = Major Frame Lock.

  - Reserved. (bits 11-0) are reserved.

10.6.3 Time Data Packets.

10.6.3.1 Time Data Packets, Format 0. Reserved.

10.6.3.2 Time Data Packets, Format 1 (IRIG/GPS/RTC). Time is treated like another data channel. If a time source other than "none" is used (Figure 10-22), the time packet will be generated at a minimum frequency of 1 Hz.

- IRIG Time Type Formats. The 10 MHz RTC shall be captured for insertion into the Time Packet Data header IAW IRIG 200 Serial Time Code Formats; On-Time Reference Marker definition.

- All Non-IRIG Time Type Formats. The 10 MHz RTC shall be captured for insertion into the Time Packet Data header consistent with the resolution with the Time Packet Body format (10 milliseconds as measured by the 10 MHz RTC).

NOTE	A Time Data Packet shall be the first dynamic data packet at the start of each recording. Only static Computer Generated Data, Format 1, packets may precede the first Time Data Packet in the recording.

NOTE	If the Time Data Packet Source is None, at least one Time Data Packet is required IAW the previous NOTE.

A packet with time data has the basic structure shown in Figure 10-21. Note that the width of the structure is not related to any number of bits. This drawing is merely to represent relative placement of data in the packet. Time Packets do not have Intra-Packet Headers.

PACKET HEADER
CHANNEL SPECIFIC DATA
TIME DATA
PACKET TRAILER

Figure 10-21. General time data packet, format 1.

a. Time Packet Channel Specific Data. The Packet Body portion of each Time Data Packet begins with a Channel Specific Data word formatted as shown in Figure 10-22.

msb 31 12	11 8	7 4	lsb 3 0
RESERVED	DATE	FMT	SRC

Figure 10-22. Time Packet channel specific data format.

- Reserved. (bits 31-12) are reserved.
- Date Format (DATE). (bits 11-8) indicates the Date format. All bit patterns not used to define a date format type are reserved for future growth.

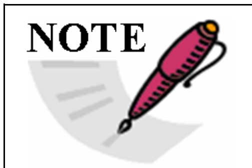
  Bits 11-10: Reserved.

  Bit 9: Indicates Date Format.
  0 = IRIG day available (Figure 10-23)
  1 = Month and Year available (Figure 10-24)

  Bit 8: Indicates if this is a leap year.
  0 = Not a leap year

1 = Is a leap year

- Time Format (FMT). (bits 7-4) indicates the Time Data Packet format.
  - 0x0 = IRIG-B
  - 0x1 = IRIG-A
  - 0x2 = IRIG-G
  - 0x3 = Real-Time Clock
  - 0x4 = UTC Time from GPS
  - 0x5 = Native GPS Time
  - 0x6 thru 0xE = Reserved
  - 0xF = None (time packet payload invalid)
- Time Source (SRC). (bits 3-0) indicates the source of the time in the payload of each time packet.
  - 0x0 = Internal (Time derived from a clock in the Recorder)
  - 0x1 = External (Time derived from a clock not in the Recorder)
  - 0x2 = Internal from RMM (Time derived from the clock in the RMM)
  - 0x3 – 0xE = Reserved
  - 0xF = None



If the Time Source (SRC) is External (0x1) and lock on the external source is lost then the Time Source (SRC) shall indicate Internal (0x0). Once lock on the external Time Source is regained, Time Source (SRC) shall once again indicate External (0x1).

b. Time Packet Body. After the Channel Specific Data word, the time data words are inserted in the packet in Binary Coded Decimal (BCD) format as shown in Figure 10-23 and Figure 10-24 (units of measure presented in Table 10-8).

<table>
<tr><td colspan="8">msb</td><td colspan="8">lsb</td></tr>
<tr><td>15</td><td></td><td></td><td>12</td><td>11</td><td></td><td></td><td>8</td><td>7</td><td></td><td></td><td>4</td><td>3</td><td></td><td></td><td>0</td></tr>
<tr><td>0</td><td colspan="3">TSn</td><td colspan="4">Sn</td><td colspan="4">Hmn</td><td colspan="4">Tmn</td></tr>
<tr><td>0</td><td>0</td><td colspan="2">THn</td><td colspan="4">Hn</td><td>0</td><td colspan="3">TMn</td><td colspan="4">Mn</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td colspan="2">HDn</td><td colspan="4">TDn</td><td colspan="4">Dn</td></tr>
</table>

Figure 10-23. Time data - packet format, day format.

<table>
<tr><td colspan="8">msb</td><td colspan="8">lsb</td></tr>
<tr><td>15</td><td></td><td></td><td>12</td><td>11</td><td></td><td></td><td>8</td><td>7</td><td></td><td></td><td>4</td><td>3</td><td></td><td></td><td>0</td></tr>
<tr><td>0</td><td colspan="3">TSn</td><td colspan="4">Sn</td><td colspan="4">Hmn</td><td colspan="4">Tmn</td></tr>
<tr><td>0</td><td>0</td><td colspan="2">THn</td><td colspan="4">Hn</td><td>0</td><td colspan="3">TMn</td><td colspan="4">Mn</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>TOn</td><td colspan="4">On</td><td colspan="4">TDn</td><td colspan="4">Dn</td></tr>
<tr><td>0</td><td>0</td><td colspan="2">OYn</td><td colspan="4">HYn</td><td colspan="4">TYn</td><td colspan="4">Yn</td></tr>
</table>

Figure 10-24. Time data - packet format, day, month, and year format.

TABLE 10-8. UNITS OF MEASURE

Tmn	Tens of milliseconds	TDn	Tens of days
Hmn	Hundreds of milliseconds	HDn	Hundreds of Days
Sn	Units of seconds	On	Units of Months
TSn	Tens of Seconds	TOn	Tens of Months
Mn	Units of minutes	Yn	Units of Years
TMn	Tens of minutes	TYn	Tens of Years
Hn	Units of hours	HYn	Hundreds of Years
THn	Tens of Hours	OYn	Thousands of Years
Dn	Units of Days	0	Always zero

10.6.4 MIL-STD-1553.

10.6.4.1 MIL-STD-1553 Bus Data Packets, Format 0. Reserved

10.6.4.2 MIL-STD-1553 Bus Data Packets, Format 1 (MIL-STD-1553B Bus Data). MIL-STD-1553 BUS data is packetized in Message Mode, with each 1553 bus “transaction” recorded as a “message.” A transaction is a BC-to-RT, RT-to-BC, or RT-to-RT word sequence, starting with the command word and including all data and status words that are part of the transaction, or a mode code word broadcast. Multiple messages may be encoded into the data portion of a single packet.

a. MIL-STD-1553 Packet Channel Specific Data. The Packet Body portion of each MIL-STD-1553 data packet begins with a Channel Specific Data word formatted as shown in Figure 10-25.

msb 31 30	29 24	23 lsb 0
TTB	RESERVED	MSGCOUNT

Figure 10-25. MIL-STD-1553 Packet channel specific data format.

- Time Tag Bits (TTB). (bits 31-30) indicate which bit of the MIL-STD-1553 message the Intra-Packet Header time tags.
  - 00 = Last bit of the last word of the message
  - 01 = First bit of the first word of the message
  - 10 = Last bit of the first (command) word of the message
  - 11 = Reserved
- Reserved. (bits 29-24) are reserved.
- Message Count (MSGCOUNT). (bits 23-0) indicates the binary value of the number of messages included in the packet. An integral number of complete messages will be in each packet.

b. MIL-STD-1553 Packet Body. A packet with **n** MIL-STD-1553 messages has the basic structure shown in Figure 10-26. Note that the width of the structure is not related to any number of bits. This drawing is merely intended to represent relative placement of data in the packet.

PACKET HEADER
CHANNEL SPECIFIC DATA
INTRA-PACKET TIME STAMP FOR MESSAGE 1
INTRA-PACKET DATA HEADER FOR MESSAGE 1
MESSAGE 1
INTRA-PACKET TIME STAMP FOR MESSAGE 2
INTRA-PACKET DATA HEADER FOR MESSAGE 2
MESSAGE 2
:
INTRA-PACKET TIME STAMP FOR MESSAGE n
INTRA-PACKET DATA HEADER FOR MESSAGE n
MESSAGE n
PACKET TRAILER

Figure 10-26. MIL-STD-1553 data packet, format 1.

c. MIL-STD-1553 Intra-Packet Header. After the Channel Specific Data, the MIL-STD-1553 data are inserted into the packet in messages. Each MIL-STD-1553 message is preceded by an Intra-Packet Header consisting of an Intra-Packet Time Stamp and an Intra-Packet Data Header.

(1) MIL-STD-1553 Intra-Packet Time Stamp. (8 Bytes) indicates the time tag of the MIL-STD-1553 message as follows.

- The 48-bit Relative Time Counter that corresponds to the data bit indicated in the MIL-STD-1553 Channel Specific Data, Time Tag Bits (paragraph 10.6.4.2a) with bits 31 to 16 in the second long word zero filled; or

- The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format corresponds to the time format indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and to the data bit indicated in the MIL-STD-1553 Channel Specific Data, Time Tag Bits (paragraph 10.6.4.2a).

(2) MIL-STD-1553 Intra-Packet Data Header. The length of the Intra-Packet Data Header is fixed at 6 bytes (48 bits) positioned contiguously, in the following sequence (Figure 10-27).

msb 15	lsb 0
BLOCK STATUS WORD	
GAP TIMES WORD	
LENGTH WORD	

Figure 10-27. MIL-STD-1553 intra-packet data header.

- Block Status Word (BSW). (bits 15-0) contain the Block Status Word for both the message type and any 1553 bus protocol errors that occurred during the message transfer. The Block Status Word bit definitions are in Figure 10-28.

msb															lsb
15	14	13	12	11	10	9	8		6	5	4		2		0
R		BID	ME	RR	FE	TM	RESERVED			LE	SE	WE	RESERVED		

Figure 10-28. Block status word format.

  - Reserved (R). (bits 15-14) are reserved.
  - Bus ID (BID). (bit 13) indicates the bus ID for the message.
    0 = Message was from Channel A
    1 = Message was from Channel B
  - Message Error (ME). (bit 12) indicates a message error was encountered.
    0 = No message error
    1 = Message error
  - RT to RT Transfer (RR). (bit 11) indicates a RT to RT transfer; message begins with two command words.
    0 = No RT to RT transfer
    1 = RT to RT transfer
  - Format Error (FE). (bit 10) indicates any illegal gap on the bus other than Response Time Out.
    0 = No format error
    1 = Format error
  - Response Time Out (TM). (bit 9) indicates a response time out occurred. The bit is set if any of the Status Word(s) belonging to this message didn't arrive within the response time of 14μs defined by MIL-STD-1553B.
    0 = No response time out
    1 = Response time out
  - Reserved. (bits 8-6) are reserved.
  - Word Count Error (LE). (bit 5) indicates that the number of data words transmitted is different than identified in the command word. A MIL-STD-1553B Status Word with the busy bit set to true will not cause a Word Count Error. A transmit command with a response timeout will not cause a Word Count Error.
    0 = No word count error
    1 = Word count error

- Sync Type Error (SE). (bit 4) indicates an incorrect sync type occurred.
  0 = No sync type error
  1 = Sync type error
- Invalid Word Error (WE). (bit 3) indicates an invalid word error occurred. This includes Manchester decoding errors in the synch pattern or word bits, invalid number of bits in the word, or parity error.
  0 = No invalid word error
  1 = Invalid word error
- Reserved. (bits 2-0) are reserved.

NOTE	Gap Times (Response Time): The Gap Times Word indicates remote terminal response times as defined by MIL-STD-1553. The resolution of the response time shall be in tenths of microseconds. A maximum of two Response Time Words can exist. Remote Terminal to Remote Terminal type messages shall have two Response Time Words if both terminals respond; all other messages will have one Response Time Word, or none for broadcast type messages or messages with no Remote Terminal response.

- Gap Times Word (bits 15-0). The Gap Times Word indicates the number of tenths of microseconds in length of the internal gaps within a single transaction. For most messages, only GAP1 is meaningful. It measures the time between the command or data word and the first (and only) status word in the message. For RT-to-RT messages, GAP2 measures the time between the last data word and the second status word. The Gap Times Word bit definitions are as shown in Figure 10-29.

msb 15 8	7 lsb 0
GAP2	GAP1

Figure 10-29. Gap Times word format.

NOTE	Gap measurements shall be made IAW MIL-STD-1553 response time measurements from the mid-bit zero crossing of the parity bit of the last word to the mid-zero crossing of the sync of the status word.

- Length Word Bits (15-0). The Length of the message is the total number of bytes in the message. A message consists of command words, data words, and status words.

d. Packet Format. Unless an error occurred, as indicated by one of the error flags in the Block Status Word, the first word following the Length Word should always be a command word. The resultant packets have the format shown in Figure 10-30.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 15-0)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 31-16)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 47-32)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 63-48)	
INTRA-PACKET DATA HEADER FOR MSG 1 (BITS 15-0)	
INTRA-PACKET DATA HEADER FOR MSG 1 (BITS 31-16)	
INTRA-PACKET DATA HEADER FOR MSG 1 (BITS 47-32)	
COMMAND WORD	
COMMAND, STATUS, OR DATA WORD	
DATA OR STATUS WORD	
:	
DATA OR STATUS WORD	
INTRA-PACKET TIME STAMP FOR MSG 2 (BITS 15-0)	
INTRA-PACKET TIME STAMP FOR MSG 2 (BITS 31-16)	
INTRA-PACKET TIME STAMP FOR MSG 2 (BITS 47-32)	
INTRA-PACKET TIME STAMP FOR MSG 2 (BITS 63-48)	
INTRA-PACKET DATA HEADER FOR MSG 2 (BITS 15-0)	
INTRA-PACKET DATA HEADER FOR MSG 2 (BITS 31-16)	
INTRA-PACKET DATA HEADER FOR MSG 2 (BITS 47-32)	
COMMAND WORD	
COMMAND, STATUS, OR DATA WORD	
DATA OR STATUS WORD	
:	
DATA OR STATUS WORD	
:	
INTRA-PACKET TIME STAMP FOR MSG N (BITS 15-0)	
INTRA-PACKET TIME STAMP FOR MSG N (BITS 31-16)	
INTRA-PACKET TIME STAMP FOR MSG N (BITS 47-32)	
INTRA-PACKET TIME STAMP FOR MSG N (BITS 63-48)	
INTRA-PACKET DATA HEADER FOR MSG N (BITS 15-0)	
INTRA-PACKET DATA HEADER FOR MSG N (BITS 31-16)	
INTRA-PACKET DATA HEADER FOR MSG N (BITS 47-32)	
COMMAND WORD	
COMMAND OR DATA, WORD	
DATA OR STATUS WORD	
:	
DATA OR STATUS WORD	
PACKET TRAILER	

Figure 10-30. MIL-STD-1553 data packet, format 1.

10.6.4.3 MIL-STD-1553 Bus Data Packets, Format 2 (Bus 16PP194 Weapons Bus Data). This data type provides packetization for F-16 MIL-STD-1553 weapons multiplex bus as defined in document 16PP362A (Weapons MUX (WMUX) protocols). A 16PP194 transaction consists of 6 each 32-bit words consisting of a 16PP194 COMMAND (1), COMMAND (1) ECHO, COMMAND (2), COMMAND (3, GO NOGO), COMMAND (4, GO NOGO), STATUS as illustrated in Figure 10-31. Multiple transactions may be encoded into the data portion of a single packet.

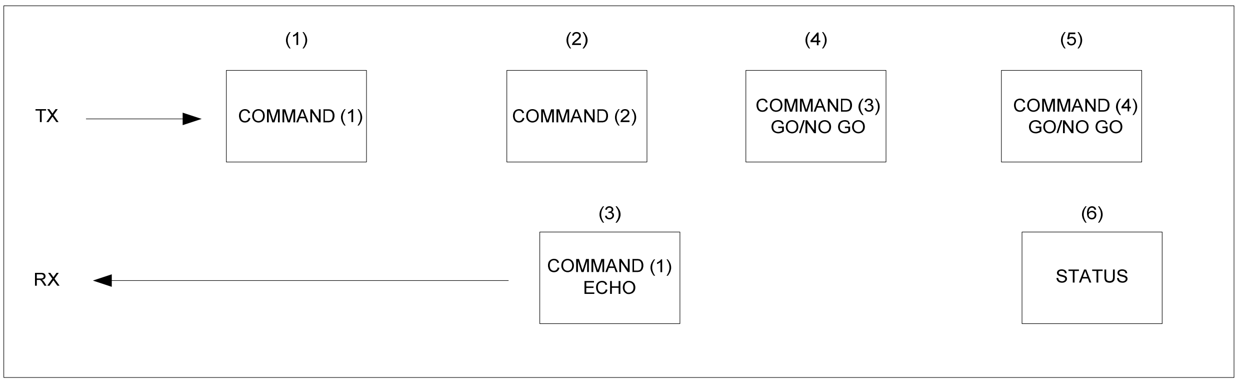


Figure 10-31. 16PP194 message transaction.

a. MIL-STD-1553 16PP194 Packet Channel Specific Data Word. The Packet Body portion of each 16PP MIL-STD-1553 data packet begins with a Channel Specific Data word formatted as shown in Figure 10-32.

msb 31	lsb 0
MSGCOUNT	

Figure 10-32. MIL-STD-1553 16PP194 packet channel specific data format.

- Message Count (MSGCOUNT). (bits 31-0) indicates the binary value of the number of messages included in the packet. An integral number of complete transaction messages will be in each packet.

b. MIL-STD-1553 16PP194 Packet Body. A packet with n MIL-STD-1553 16PP194 transactions has the basic structure shown in Figure 10-33 below. This drawing is merely to represent relative placement of data in the packet.

msb 31	lsb 0
PACKET HEADER	
16PP194 CHANNEL SPECIFIC DATA WORD	
INTRA-PACKET TIME STAMP (LSLW)	
INTRA-PACKET TIME STAMP (MSLW)	
INTRA PACKET DATA HEADER LENGTH WORD	INTRA PACKET DATA HEADER STATUS WORD
DATA 1	
.	
.	
INTRA-PACKET TIME STAMP (LSLW)	
INTRA-PACKET TIME STAMP (MSLW)	
INTRA PACKET DATA HEADER LENGTH WORD	INTRAPACKET DATA HEADER STATUS WORD
DATA N	
.	
.	
PACKET TRAILER	

Figure 10-33. MIL-STD-1553 16PP194 data packet.

c. <u>MIL-STD-1553 16PP194 Intra-Packet Header</u>. The intra-packet header consists of the intra-packet data header (LENGTH and STATUS) and the intra-packet time stamp.

- <u>MIL-STD-1553 16PP194 Intra-Packet Data Header LENGTH</u>. The length word contains the length in bytes of the intra-packet data.

NOTE	The Intra-packet length is fixed to 0x18, 24 bytes.

- <u>MIL-STD-1553 16PP194 Intra-Packet Data Header STATUS</u>. The status word contains error and special handling information about the data. The error indicator bits when set to a '1' reflect that such an error is present in the data or occurred during data reception. The format of the status word is shown in Figure 10-34.

msb 15	14	13	12 7	6	5 4	3	2 lsb 0
TE	RE	TM	RESERVED	SE	R	EE	RESERVED

Figure 10-34. MIL-STD-1553 16PP194 intra-packet data header format.

- Transaction Error (TE): (bit 15) Error condition found in 16PP194 transaction.
  0 = No Errors found in current transaction
  1 = ERROR Condition Found in transaction
- Reset (RE): (bit 14) Indicates a 16PP194 Bus Master Reset.
  0 = No Master Reset
  1 = Master reset Detected on
- Message Time Out (TM). (bit 13) Indicates a transaction time out occurred.
  0 = No message time out
  1 = Message time out
- Reserved. (bits 12-7) are reserved
- Status Error (SE). (bit 6) Indicates Status Word missing in transaction.
  0 = Status word present
  1 = Status word missing
- Reserved (R). (bits 5-4) are reserved.
- Echo Error (EE). (bit 3) Indicates Echo Word missing in transaction.
  0 = Echo word present
  1 = Missing Echo Word
- Reserved. (bits 2-0) are reserved.

- MIL-STD-1553 16PP194 Intra-Packet Time Stamp. The intra-packet time stamp (64 bits total) contains a 48-bit relative timestamp in the low-order bits. The 16 high-order bits are zero.

d. Packet Format. Unless an error occurred, as indicated by one of the error flags in the Intra-Packet Data Header, the first word following the Length should always be the first transaction command word. The resultant packets have the format shown in Figure 10-35.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
INTRA PACKET TIME STAMP (BITS 0-15)	
INTRA PACKET TIME STAMP (BITS 31-16)	
INTRA PACKET TIME STAMP (BITS 32-47)	
INTRA PACKET TIME STAMP (BITS 48-63)	
INTRA PACKET DATA HEADER STATUS	
INTRA PACKET DATA HEADER LENGTH	
COMMAND (1) (BITS 31-16)	
COMMAND (1)(BITS 15-0)	
COMMAND (1) ECHO (BITS 31-16)	
COMMAND (1) ECHO (BITS 15-0)	
COMMAND (2) (BITS 31-16)	
COMMAND (2) (BITS 15-0)	
COMMAND (3) GO NOGO (BITS 31-16)	
COMMAND (3) GO NOGO (BITS 15-0)	
COMMAND (4) GO NOGO ECHO (BITS 31-16)	
COMMAND (4) GO NOGO ECHO (BITS 15-0)	
STATUS (BITS 31-16)	
STATUS (BITS 15-0)	
INTRA PACKET TIME STAMP (BITS 0-15)	
INTRA PACKET TIME STAMP (BITS 31-16)	
INTRA PACKET TIME STAMP (BITS 32-47)	
INTRA PACKET TIME STAMP (BITS 48-63)	
INTRA PACKET DATA HEADER STATUS	
INTRA PACKET DATA HEADER LENGTH	
COMMAND (1) (BITS 31-16)	
COMMAND (1) (BITS 15-0)	
COMMAND (1) ECHO (BITS 31-16)	
COMMAND (1) ECHO (BITS 15-0)	
COMMAND (2) (BITS 31-16)	
COMMAND (2) (BITS 15-0)	
COMMAND (3) GO NOGO (BITS 31-16)	
COMMAND (3) GO NOGO (BITS 15-0)	
COMMAND (4) GO NOGO ECHO (BITS 31-16)	
COMMAND (4) GO NOGO ECHO (BITS 15-0)	
STATUS (BITS 31-16)	
STATUS (BITS 15-0)	
•	
•	
PACKET TRAILER	

Figure 10-35. MIL-STD-1553 16PP194 data packet.

e. MIL-STD-1553 16PP194 Data Format. Each 26-bit 16PP194 word in a 16PP194 transaction shall be formatted into two 16-bit words (Figure 10-36). The corresponding 16PP194 Sync and Parity bits will not be formatted into the 16PP194 words.

msb 15 14 13	12 11 10	9	8	7 lsb 0
BUS ID	GAP	W	P	16PP194Data Word (bits 24-17)
16PP194 Data Word (bits 16-1)				

Figure 10-36. MIL-STD-1553 26-bit 16PP194 word format.

- MIL-STD-1553 16PP194 BUS ID (BUS ID). A three-bit field shall be used to indicate bus identification as follows.

111	CIU Left Bus A
110	CIU Left Bus B
101	CIU Right Bus A
100	CIU Right Bus B
011	Response Bus A and B
010	Response Bus A
001	Response Bus B
000	Incomplete Transaction

- MIL-STD-1553 16PP194 GAP (GAP). A three-bit field shall be used to indicate GAP between transactions as follows.

111	GAP> 9.15us
110	7.55uS <GAP <=9.15 uS
101	5.95 uS<GAP <=7.55uS
100	4.35 uS<GAP <=5.95
011	2.75 uS<GAP <=4.35uS
010	2.75 uS<GAP <=4.35uS
001	1.15 uS<GAP <=2.75uS
000	Not Applicable

NOTE	Gap time is measured from mid-crossing of parity bit from previous received word to the mid-crossing of the sync bit of the current word in 1uS counts.

- MIL-STD-1553 16PP194 Word Bit Error (W). If the bit is set to “1,” this indicates that a Manchester Error was detected in the word.
- MIL-STD-1553 16PP194 Word Parity Error (P). If the bit is set to “1,” this indicates that a parity error occurred in the word.
- 16PP194 Data Word (bits 16-1): 16PP194 data field as in Figure 10-37.

- 16PP194 Data Word (bits 24-17): 16PP194 RIU Address and RIU Sub Address as in Figure 10-37.

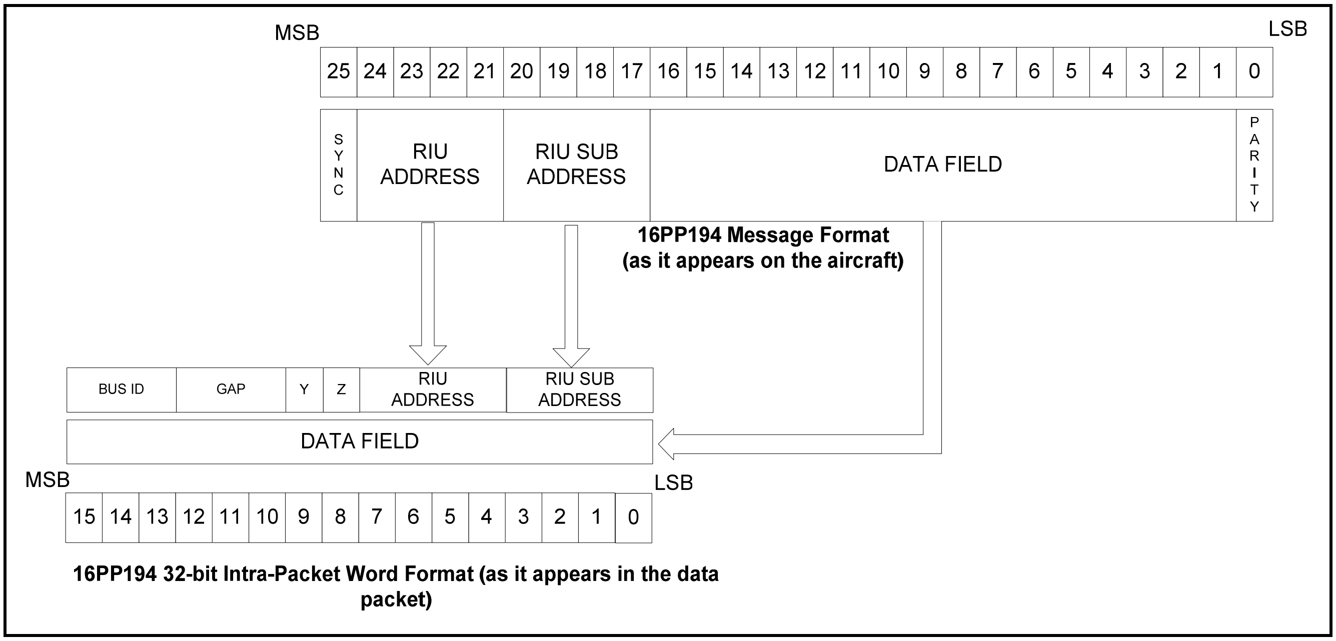


Figure 10-37. 16PP194 Word Format.

10.6.5 Analog Data Packets.

10.6.5.1 Analog Data Packets, Format 0. Reserved.

10.6.5.2 Analog Data Packets, Format 1. The generic packet structure for analog data is illustrated in Figure 10-38.

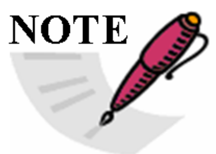
An Analog Data Packet will contain a Channel Specific Data word for each subchannel of analog data sampled within that packet if the SAME bit is set to 0, or it will contain a single Channel Specific Data Word for the entire analog packet if the SAME bit is set to 1. This will be followed by at least one complete sampling schedule of data.

A sampling schedule is defined as a sampling sequence in which each subchannel, described by a Channel Specific Data word, is sampled at least once. In many cases, due to simultaneous sampling rules and varied sampling rates, a particular subchannel will be sampled more than once during a sampling schedule. In addition, multiple complete sampling schedules may be included in a single packet. For these reasons, the number of Channel Specific Data words will usually be less than the number of samples.

Figure 10-38 depicts the generic packet data structure for M data subchannels and a single sampling schedule that has a length N. Note that the width of the structure is not related to any number of bits and is merely intended to represent relative placement of words within the packet.

PACKET HEADER
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 1
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 2
:
:
:
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL M
SAMPLE 1
SAMPLE 2
: : :
SAMPLE N
PACKET TRAILER

Figure 10-38. Generic analog data packet, format 1.



The Packet Header Time in an Analog Data packet shall correspond to the first data sample in the packet. There are no Intra-Packet Headers in Analog Data Packets.

a. Analog Packet Channel Specific Data. The Packet Body portion of each Analog packet begins with the Channel Specific Data word(s). Each subchannel that is sampled with the packet sampling schedule must have a Channel Specific Data word within the packet. Only one Channel Specific Data word is required if subchannels are sampled at the same sampling rate (“FACTOR”), and have the same bits per sample (“LENGTH”) and same packing mode ("MODE"). Bit 28 of the channel specific data word shall be used to indicate same sampling data rate for subchannels.

Channel Specific Data words for Analog Data Packets are formatted as shown in Figure 10-39.

msb 31 29	28	27 24	23 16	15 8	7 2	1 lsb 0
RESERVED	SAME	FACTOR	TOTCHAN	SUBCHAN	LENGTH	MODE

Figure 10-39. Analog packet channel specific data word.

- Reserved. (bits 31-29) are reserved.
- Same. (bit 28) specifies if this Channel Specific Data Word applies for all the channels included in the packet, or if each channel has its own Channel Specific Data Word.

0 = each analog channel has its own Channel Specific Data Word
1 = The Channel Specific Data Word is valid for all analog channels stored in this packet

- Factor. (bits 27-24) is the exponent of the power of 2 sampling rate factor denominator for the corresponding subchannel in the range 0 to 15. (The sampling rate factor numerator is always 1.)
  0x0 = sampling rate factor denominator 2^0 = 1 => factor = $^1/_1$
  0x1 = sampling rate factor denominator 2^1 = 2 => factor = ½
  0x2 = sampling rate factor denominator 2^2 = 4 => factor = ¼
  :
  0xF = sampling rate factor denominator 2^{15} = 32768 => factor = $^1/_{32768}$

- Totchan. (bits 23-16) indicates the total number of analog subchannels in the packet (and the number of Channel Specific Data words in the packet).

  This Totchan field must be the same value in all Channel Specific Data words in a single packet. The Totchan value may be less than the largest Subchan value. This can happen when a multi-channel analog input device has some of its subchannels disabled (turned off) for a specific recording. For example, if an analog input device has eight subchannels and not all eight are active, an analog data packet may have three subchannels (Totchan=3) numbered 4, 7, and 8 (enabled Subchan = 4, 7, 8). The number of subchannels (Totchan) and the subchannel number for each active subchannel (Subchan) in a packet are identified in the accompanying TMATS (Computer Generated Data, Format 1) packet.
  0x00 = 256 Subchannels
  0x01 = 1 Subchannel
  0x02 = 2 Subchannels

- Subchan. (bits 15-8) indicates a binary value representing the number of the analog subchannel.

  When an Analog Packet contains data from more than one subchannel and the Channel Specific Data Words are not the same for all channels (see field Same, bit 28), the Channel Specific Data words must be inserted into the packet in ascending subchannel number as identified by this Subchan field. The Subchan values in these Channel Specific Data words need not be contiguous (see Totchan), but they must be in ascending decimal numerical order with the exception that subchannel 0 (256) is last. If the "Same" bit is set, the Subchan field shall be set to zero.
  0x01 = Subchannel 1
  0x02 = Subchannel 2
  :
  0x00 = Subchannel 256
  :

- Length. (bits 7-2) indicates a binary value representing the number of bits in the Analog to Digital converter (A/D).
  - 000000 = sixty-four bit samples
  - 000001 = one bit samples
  - :
  - 001000 = eight bit samples
  - :
  - 001100 = twelve bit samples
  - :

- Mode. (bits 1-0) indicates alignment and packing modes of the analog data. Bit 0 is the packing bit and bit 1 is the alignment bit. When TOTCHAN, defined, is more than 1, the Mode must be the same for all subchannels in a single packet.
  - 00 = data is packed
  - 01 = data is unpacked, lsb padded
  - 10 = Reserved for future definition
  - 11 = data is unpacked, msb padded

b. Analog Samples. To preserve timing relationships and allow for accurate reconstruction of the data, a simultaneous sampling scheme shall be employed. The highest sampling rate required shall define the primary simultaneous sampling rate within the packet. The primary simultaneous sampling rate is identified in the Telemetry Attributes Transfer Standard (TMATS) file describing the attributes of the analog data packet. The rate at which the other subchannels are sampled is then defined by the sampling factor (1, ½, ¼, $^1/_8$, $^1/_{16}$, $^1/_{32768}$) for each subchannel. As an example, a sampling factor of ¼ would yield that subchannel being sampled at one-fourth the primary simultaneous sampling rate and a sampling factor of 1 would yield that subchannel being sampled at the primary simultaneous sampling rate.

Directly following the Channel Specific Data word(s), at least one complete sampling schedule shall be inserted in the packet. The samples, within the sampling sequence, may be inserted either unpacked, MSB Packed, or LSB Packed as described in paragraph 10.6.5.2.b(1) and 10.6.5.2.b(2). In either case, one or more subchannels may be included in a single packet. When multiple subchannels are encapsulated into a single packet, the subchannel with the highest sampling rate requirement defines the primary simultaneous sampling rate. The rate at which the other subchannels are sampled is defined by the sampling factor (contained within the Channel Specific Data words). Sampling factors are defined as:

$$\left(\frac{1}{2^K}\right) * X \quad ; K = 0, 1, 2, 3, 4, 5, \ldots$$

of the Primary Simultaneous Sampling Rate X.

The subchannels are then sampled and ordered such that:

- The highest sample rate $1 * X$ subchannel(s) appear in every simultaneous sample,
- The $\left(\frac{1}{2}\right) * X$ subchannel(s) appear in every 2nd simultaneous sample,
- The $\left(\frac{1}{4}\right) * X$ subchannel(s) appear in every 4th simultaneous sample,

… and so on until all the subchannels are sampled, resulting in a complete sampling schedule of all subchannels described by the Channel Specific Data words. In doing so, the total number of simultaneous samples (not the total number of samples) will equal the denominator of the smallest sampling factor and all subchannels will be sampled in the last simultaneous sample.

For example, a packet with six subchannels with Sampling Factors ½, $^1/_8$, 1, ½, 1, and $^1/_8$ respectively will yield a sampling sequence within the data packet as:

Simultaneous Sample 1: Subchannel 3
Simultaneous Sample 1: Subchannel 5

Simultaneous Sample 2: Subchannel 1
Simultaneous Sample 2: Subchannel 3
Simultaneous Sample 2: Subchannel 4
Simultaneous Sample 2: Subchannel 5

Simultaneous Sample 3: Subchannel 3
Simultaneous Sample 3: Subchannel 5

Simultaneous Sample 4: Subchannel 1
Simultaneous Sample 4: Subchannel 3
Simultaneous Sample 4: Subchannel 4
Simultaneous Sample 4: Subchannel 5

Simultaneous Sample 5: Subchannel 3
Simultaneous Sample 5: Subchannel 5

Simultaneous Sample 6: Subchannel 1
Simultaneous Sample 6: Subchannel 3
Simultaneous Sample 6: Subchannel 4
Simultaneous Sample 6: Subchannel 5

Simultaneous Sample 7: Subchannel 3
Simultaneous Sample 7: Subchannel 5

Simultaneous Sample 8: Subchannel 1
Simultaneous Sample 8: Subchannel 2
Simultaneous Sample 8: Subchannel 3
Simultaneous Sample 8: Subchannel 4
Simultaneous Sample 8: Subchannel 5
Simultaneous Sample 8: Subchannel 6

Notice that the denominator of the smallest sampling factor defines the number of simultaneous samples within the packet (in this example, 8). However, the total number of samples within the sampling schedule does not have to equal the number of simultaneous samples (in this example, 26). Also notice that all subchannels are sampled during the last Simultaneous Sample. The order of the subchannel samples in each simultaneous sample is ascending by subchannel number.

Any number of complete sampling schedules may be placed within a packet so that the maximum packet length is not exceeded. The TMATS file identifies the number of samples contained within each packet.

(1) Unpacked Mode. In Unpacked Mode, packing is disabled and each sample is padded with the number of bits necessary to align each word with the next 16-bit boundary in the packet. Four (4) pad bits are added to 12 bit words, eight (8) pad bits are added to 8-bit words, etc. All pad bits shall equal zero.

Samples less than 8 bits go into a 16-bit word boundary.

To illustrate msb padding, given M analog subchannels mapping into N samples for the special case of all samples having bit lengths of 12 bits, the resultant Analog packets with msb padding have the form shown in Figure 10-40.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 1 (BITS 15-0)	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 1 (BITS 31-16)	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 2 (BITS 15-0)	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 2 (BITS 31-16)	
:	
:	
:	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL M (BITS 15-0)	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL M (BITS 31-16)	
4-PAD BITS	SAMPLE 1, 12-DATA BITS
4-PAD BITS	SAMPLE 2, 12- DATA BITS
4-PAD BITS	SAMPLE 3, 12- DATA BITS
:	
4-PAD BITS	SAMPLE N, 12- DATA BITS
PACKET TRAILER	

Figure 10-40. Analog data packet - unpacked mode, msb padding.

To illustrate LSB Packing, given M analog subchannels mapping into N samples for the special case of all samples having bit lengths of 12 bits, the resultant Analog packets with LSB padding have the form shown in Figure 10-41.

<table>
<tr><td colspan="2">msb
15 lsb
0</td></tr>
<tr><td colspan="2">PACKET HEADER</td></tr>
<tr><td colspan="2">CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 1 (BITS 15-0)</td></tr>
<tr><td colspan="2">CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 1 (BITS 31-16)</td></tr>
<tr><td colspan="2">CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 2 (BITS 15-0)</td></tr>
<tr><td colspan="2">CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 2 (BITS 31-16)</td></tr>
<tr><td colspan="2">:</td></tr>
<tr><td colspan="2">:</td></tr>
<tr><td colspan="2">:</td></tr>
<tr><td colspan="2">CHANNEL SPECIFIC DATA WORD, SUBCHANNEL M (BITS 15-0)</td></tr>
<tr><td colspan="2">CHANNEL SPECIFIC DATA WORD, SUBCHANNEL M (BITS 31-16)</td></tr>
<tr><td colspan="2">:</td></tr>
<tr><td>SAMPLE 1, 12- DATA BITS</td><td>4-PAD BITS</td></tr>
<tr><td>SAMPLE 2, 12- DATA BITS</td><td>4-PAD BITS</td></tr>
<tr><td>SAMPLE 3, 12- DATA BITS</td><td>4-PAD BITS</td></tr>
<tr><td colspan="2"></td></tr>
<tr><td>SAMPLE N, 12- DATA BITS</td><td>4-PAD BITS</td></tr>
<tr><td colspan="2">PACKET TRAILER</td></tr>
</table>

Figure 10-41. Analog data packet - unpacked mode, lsb padding.

(2) Packed Mode. In Packed Mode, packing is enabled and padding is not added to each data word. However, if the number of bits in the packet are not an integer multiple of 16, then Y filler bits will be used to msb fill the last data word, forcing alignment on a 16-bit boundary. The value of Y is sixteen (16) minus the integer remainder of L, the total number of data bits in the packet, divided by 16 and is mathematically expressed as:

$$Y = 16 - (\text{MODULUS}\{L,16\}).$$

To illustrate msb padding, given M Analog subchannels mapping into N samples for the special case of all samples having bit lengths of 12 bits, the resultant Analog packets with padding bits at the end of the Nth sample have the form shown in Figure 10-42.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 1 (BITS 15-0)	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 1 (BITS 31-16)	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 2 (BITS 15-0)	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL 2 (BITS 31-16)	
:	
:	
:	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL M (BITS 15-0)	
CHANNEL SPECIFIC DATA WORD, SUBCHANNEL M (BITS 31-16)	
SAMPLE 2 (BITS 3-0)	SAMPLE 1 (BITS 11-0)
SAMPLE 3 (BITS 7-0)	SAMPLE 2 (BITS 11-4)
:	:
:	:
:	:
Y PADDING BITS	SAMPLE N (BITS 11-0)
:	
PACKET TRAILER	

Figure 10-42. Analog data packet - packed mode packet.

10.6.6 Discrete Data Packets.

10.6.6.1 Discrete Data Packets, Format 0. Reserved.

10.6.6.2 Discrete Data Packets, Format 1. A packet with Discrete data has the basic structure shown in Figure 10-43. Note that the width of the structure is not related to any number of bits. This drawing is merely intended to represent relative placement of data in the packet. One to 32 discrete states may be recorded for each event.

PACKET HEADER
CHANNEL SPECIFIC DATA
INTRA-PACKET HEADER FOR EVENT 1
EVENT 1 STATES
INTRA-PACKET HEADER FOR EVENT 2
EVENT 2 STATES
:
INTRA-PACKET HEADER FOR EVENT N
EVENT N STATES
PACKET TRAILER

Figure 10-43. General discrete data packet, format 1.

a. Discrete Packet Channel Specific Data Word. The Packet Body portion of each Discrete packet begins with the Channel Specific Data Word, which is formatted as shown in Figure 10-44.

msb 31 8	7 3	2 lsb 0
RESERVED	LENGTH	MODE

Figure 10-44. Discrete packet channel data word format.

- Reserved. (bits 31-8)
- Length (bits 7-3). Indicates a binary value representing the number of bits in the event.
- Mode (bits 2-0). Indicates the mode of accessing the discrete data.

  Bit 0: indicates the Record State.
  0 = discrete data is recorded when the state changes
  1 = discrete data is recorded on a time interval basis

  Bit 1: indicates the alignment of the data.
  0 = lsb
  1 = msb

  Bit 2: reserved.

b. Discrete Data. After the Channel Specific Data, Discrete data (Figure 10-45) is inserted in the packet. Discrete data are described as Events. Each Event includes the Event State for each discrete input and the corresponding Intra packet Time. The Event State is a 32-bit word that provides the logical state of each discrete input.

msb 31	30		1	lsb 0
D31	D30	…	D1	D0

Figure 10-45. Discrete data format.

- Discrete Event Bits. (Bits 31-0) indicate the states of the discrete event bits.

  Bit 31: indicates Discrete 31 (D31) State.
  0 = discrete 31 is at state 0
  1 = discrete 31 is at state 1

  Bit 30: indicates Discrete 30 (D30) State.
  0 = discrete 30 is at state 0
  1 = discrete 30 is at state 1

  Bit 1: indicates Discrete 1 (D1) State.
  0 = discrete 1 is at state 0
  1 = discrete 1 is at state 1

Bit 0: indicates Discrete 0 (DO) State.
0 = discrete 0 is at state 0
1 = discrete 0 is at state 1

c. Discrete Event Intra-Packet Header. All discrete events shall include an Intra-Packet Header consisting of an Intra-Packet Time Stamp only, which is inserted immediately before the discrete event. The length of the Intra-Packet Header is fixed at 8 bytes (64 bits) positioned contiguously, arranged in the sequence shown in Figure 10-46.

msb 31	lsb 0
TIME (LSLW)	
TIME (MSLW)	

Figure 10-46. Discrete event intra-packet header.

- Intra-Packet Time Stamp. (8 Bytes) indicate the time tag of the discrete event. First Long Word Bits 31-0 and Second Long Word Bits 31-0 indicate the following values:

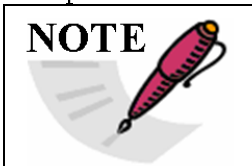
  (1) The 48-bit Relative Time Counter that corresponds to the first data bit of the discrete event with bits 31 to 16 in the second long word zero filled or;

  (2) The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format corresponds to the time format indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and to the first data bit of the discrete event. The discrete data packet format is presented in Figure 10-47.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
INTRA-PACKET TIME STAMP FOR EVENT 1 (BITS 15–0)	
INTRA-PACKET TIME STAMP FOR EVENT 1 (BITS 31–16)	
INTRA-PACKET TIME STAMP FOR EVENT 1 (BITS 47-32)	
INTRA-PACKET TIME STAMP FOR EVENT 1 (BITS 63-48)	
STATES FOR EVENT 1 (BITS 15–0)	
STATES FOR EVENT 1 (BITS 31–16)	
:	
INTRA-PACKET TIME STAMP FOR EVENT n (BITS 15–0)	
INTRA-PACKET TIME STAMP FOR EVENT n (BITS 31–16)	
INTRA-PACKET TIME STAMP FOR EVENT n (BITS 47-32)	
INTRA-PACKET TIME STAMP FOR EVENT n (BITS 63-48)	
STATES FOR EVENT n (BITS 15–0)	
STATES FOR EVENT n (BITS 31–16)	
PACKET TRAILER	

Figure 10-47. Discrete data packet format.

10.6.7 Computer Generated Data Packets. Packets with Computer Generated Data have the basic structure shown in Figure 10-48. Formats 0, 1, 2, and 3 are used to add information packets to recorded data. This information contains annotation data, setup records, events, or index information for the data that has been recorded. The width of the structure is not related to any number of bits. This drawing is merely intended to represent relative placement of data in the packet.



Computer Generated Data is defined as non-external data or data generated within the recorder. After the Channel Specific Data Word, the Computer Generated Data is inserted in the packet. The organization and content of the Computer Generated Data is determined by the specific Format type.

PACKET HEADER
CHANNEL SPECIFIC DATA
COMPUTER GENERATED DATA
PACKET TRAILER

Figure 10-48. General computer generated data packet format.

10.6.7.1 Computer Generated Data Packets Format 0, User Defined. Format 0 enables the insertion of user-defined Computer Generated Data. Data can not be placed in this packet if the data type is already defined within this standard nor can data be inserted in this packet if it is generated from an external input to the recorder.

- Computer Generated Packets Format 0 Channel Specific Data Word. The Packet Body portion of each Format 0 Packet begins with the Channel Specific Data word, which is formatted as shown in Figure 10-49.

msb 31	lsb 0
RESERVED	

Figure 10-49. Computer generated format 0 channel specific data word format.

  - Reserved. (bits 31-0) are reserved.

10.6.7.2 Computer Generated Data Packets Format 1, Setup Records. Format 1 defines a setup record that describes the hardware, software, and data channel configuration used to produce the other data packets in the file. The organization and content of a Format 1 Setup Record is indicated in the Channel Specific Data Word FRMT field.

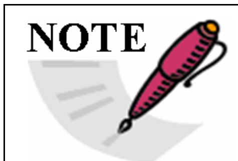
It is mandatory for a setup TMATS record to be utilized to configure the recorder. A Format 1 Computer Generated Data Packet containing the setup TMATS record utilized to configure the recorder shall be the first packet in each data file. A single setup record may span multiple consecutive packets. When spanning multiple packets, the sequence counter shall increment in the order of segmentation of the setup record, n+1.

a. Format 1 - Channel Specific Data Word. The Packet Body portion of each Format 1 Packet begins with the Channel Specific Data word, which is formatted as shown in Figure 10-50.

msb 31 10	9	8	7 lsb 0
RESERVED	FRMT	SRCC	CH10VER

Figure 10-50. Computer generated format 1 channel specific data word format.

- Reserved. (bits 31-10) are reserved.
- FRMT (bit 9): Setup record format.
  0 = Setup Record IAW IRIG-106 Chapter 9 ASCII Format
  1 = Setup Record IAW IRIG-106 Chapter 9 XML Format



It is not permissible to have both ASCII and XML IRIG-106 Chapter 9 TMATS attributes in the same recording or recording sessions.

- Setup Record Configuration Change (SRCC). (bit 8) indicates if the recorder configuration contained in the previous Setup Record packet(s) of the current recording session (defined as .RECORD to .STOP) has changed.

0 = Setup Record Configuration has Not Changed
1 = Setup Record Configuration has Changed

NOTE When a setup record configuration change has taken place, bit 8 (SRCC) shall be set to 1 and the new setup record packet will be committed to the stream **prior** to any new or changed data packets being committed to the stream. The next setup record packet(s) committed to the stream, if not changed from this new setup record, shall clear the SRCC bit to 0.

NOTE Prior to the new setup record being committed to the stream, a “Setup Record Configuration Change” event packet shall be inserted into the stream.

NOTE Each new setup record packet must adhere to all applicable setup record requirements including, but not limited to, the minimum required TMATS attributes.

- IRIG-106 Chapter 10 Version (CH10VER). A 1-Byte indicator of which IRIG-106 Chapter 10 release version the recorder requirements and following recorded data are applicable to and comply with. The value shall be represented by the following bit patterns:

0x00 thru 06	=	Reserved
0x07	=	IRIG-106-07
0x08	=	IRIG-106-09
0x09 thru 0xFF	=	Reserved

  Individual paragraph 10.6 Data Types and their format/content compliancy and applicability with the IRIG-106 Chapter 10 release version are defined in paragraph 10.6.1.1e.

10.6.7.3 Computer Generated Data Packets Format 2, Recording Event. Format 2 defines a recording event packet that contains the occurrence and information of one or more individual events that have been defined within the Format 1 Setup Record IAW “Recording Events” attribute. If the Recording Events information is larger than the maximum packet size of 512K bytes, the Recording Events information may be contained in multiple packets using the Major Packet Header Sequence Number.

Events associated with the .EVENT command defined in Table 10-9 can only be directly accessed from the recorder itself and are not contained within the recording data. This does not preclude defining an event driven by the .EVENT command to also be defined within the Recording Event setup record information and placed in the appropriate event entry within an event packet. The .EVENT recorder command and the Recording Event packets will not be directly linked in this standard and any linking between the two will be an implementation of this standard within a recorder.

NOTE	It is not the intent for the event packets within the data to be directly coupled with recorder events per the .EVENT command in Table 10-9.

a. Event Packet Location. Recording Event packets may be placed at any location within the recording after the first Time Data packet and before the last Root Index Packet. This may be at the time each event occurs, after multiple events have occurred or at an interval of time or packets. The complete event log of a recording (defined in Event Period of Capture paragraph 10.6.7.3c) is constituted by the contents of all event packets in a recording concatenated in order of which the event(s) occurred in time.

NOTE	Index Packets *will be enabled* if Recording Event Packets are *enabled*.

b. Channel Specific Data Word. The Packet Body portion of each Format 2 Packet begins with the Channel Specific Data word, which is formatted as shown in Figure 10-51.

msb 31	30 ... 12	11 ... lsb 0
IPDH	RESERVED	REEC

Figure 10-51. Computer generated format 2 channel specific data word.

- Recording Event Intra-Packet Data Header (IPDH). (bit 31) indicates the presence of the Intra-Packet Data Header.
  0 = Recording Event Intra-Packet Data Header Not Present
  1 = Recording Event Intra-Packet Data Header Present

- Reserved. (bits 30-12) are reserved.

- Recording Event Entry Count (REEC). (bits 11-0) are an unsigned binary that identifies the count of recording event entries included in the packet.

c. Event Period of Capture. Denotes the period of capture (Figure 10-52), and is defined to encompass the events occurring from the time a .RECORD command (paragraph 10.7.9) is issued (if it is the first recording) until the time a .STOP command (paragraph 10.7.9) is issued. If there is a previous recording, the period of capture is described as encompassing those events that occur from the previous recording's .STOP command until the .STOP command of the current recording. This ensures that any events that occurred between recordings will be captured and will include special indicators that the event occurred between .STOP and .RECORD commands.

Priority conditions and event limit counts are defined in the setup record attributes for each defined event. The ability to put finite limits on events during periods of capture precludes overflowing buffers or media capacities. These priority conditions and event limit counts are as follows:

Priority 1: Defined event will always be captured during and in between recordings.
Priority 2: Defined event will always be captured during recordings and up to a limit count between recordings.
Priority 3: Defined event will always be captured during recordings and not captured between recordings.
Priority 4: Defined event will be captured up to a limit count during recordings and between recordings.
Priority 5: Defined event will be captured up to a limit count for each defined event during recordings and not captured between recordings.

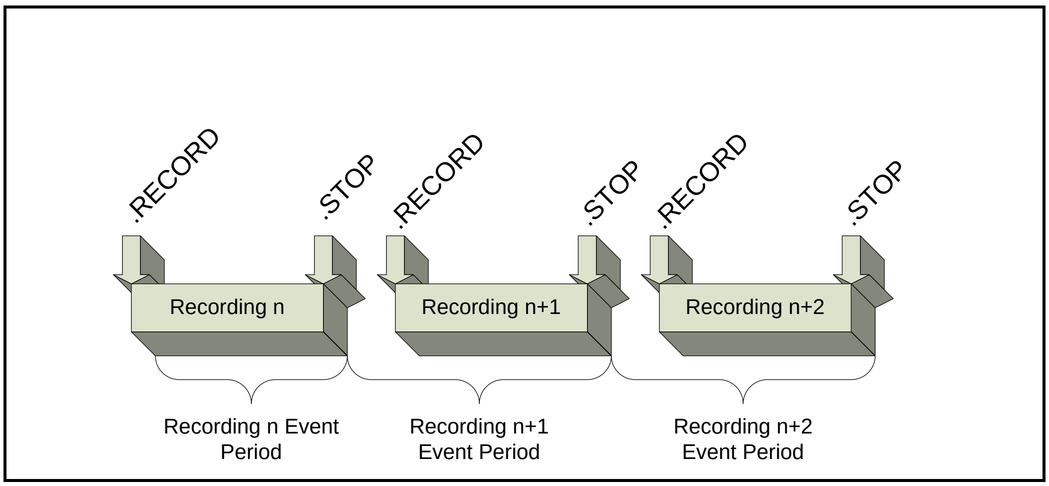


Figure 10-52. Events recording period.

d. Event Condition of Capture. Event trigger mode conditions during the Event Period of Capture are defined in the setup record attributes for each defined event. These MEASUREMENT DISCRETE or MEASUREMENT LIMIT trigger mode conditions are as follows:

Mode 1: Capture MEASUREMENT DISCRETE event at each On (1) and Off (0) mode transition sampling.
Mode 2: Capture MEASUREMENT DISCRETE event at each On (1) mode transition sampling.
Mode 3: Capture MEASUREMENT DISCRETE event at each Off (0) mode transition sampling.
Mode 4: Capture MEASUREMENT LIMIT event at each High and Low value transition sampling.
Mode 5: Capture MEASUREMENT LIMIT event at each High value transition sampling.
Mode 6: Capture MEASUREMENT LIMIT event at each Low value transition sampling.

NOTE	If Event Type is MEASUREMENT DISCRETE or MEASUREMENT LIMIT, the trigger measurement must be fully described using the Setup Record attributes for PCM, Bus, Analog, or Discrete channels. The trigger measurement source and measurement name shall be identified in the Event Definition.

e. Event Initial Capture. Event initial capture conditions are defined in the setup record attributes for each defined event. This determines if an event will be captured initially prior to the transition mode set for the event if the transition has already occurred prior to the Event Period of Capture.

f. Event Trigger Measurement Description. If Event Type is MEASUREMENT DISCRETE or MEASUREMENT LIMIT, the trigger measurement must be fully described using the Setup Record attributes for PCM, Bus, Analog, or Discrete channels. This shall include at a minimum the following attributes for the Trigger measurement:

(1) Measurement Source (via data link name).
(2) Measurement Name.
(3) Applicable Measurement Value Definition or Bit Mask.
(4) High Measurement Value (if MEASUREMENT LIMIT at or above the high limit is used to trigger the event).
(5) Low Measurement Value (if MEASUREMENT LIMIT at or below the low limit is used to trigger the event).
(6) Applicable Measurement Name Engineering Unit Conversion.

g. Recording Event Intra-Packet Time Stamp. (8 Bytes) indicates the time tag of the Recording Event Entry as follows:

(1) The 48-bit Relative Time Counter that corresponds to the Event Entry with bits 31 to 16 in the second long word zero filled. For Event Types that are MEASUREMENT DISCRETE or MEASUREMENT LIMIT, the time tag will correspond to the data packet timing mechanism containing the trigger measurement. This will either be the Packet Header Relative Time Counter value or, if enabled, the Intra-Packet Time Stamp — whichever most accurately provides the time the event occurred; or

(2) The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format corresponds to the time format indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and to the Event Entry. For Event Types that are MEASUREMENT DISCRETE or MEASUREMENT LIMIT, the time tag will correspond to the data packet timing mechanism containing the trigger measurement. This will either be the Packet Secondary Header (if enabled and using an absolute time value) or, if enabled and using an absolute time value, the Intra-Packet Time Stamp —whichever most accurately provides the time the event occurred.

The format of the recording event intra-packet header is presented in Figure 10-53. Figure 10-54 and Figure 10-55 present the general recording event packet format and recording event entry layout.

h. (Optional) Recording Event Intra-Packet Data Header. (8 Bytes) contains the absolute time of the event occurrence. The time source and format shall be derived from the Time Data Packet, Format 1. Unused high-order bits will be zero filled as needed, depending on the time type of the Time Data Packet.

msb 31	lsb 0
INTRA-PACKET TIME STAMP (LSLW)	
INTRA-PACKET TIME STAMP (MSLW)	
(OPTIONAL) INTRA-PACKET DATA HEADER (LSLW)	
(OPTIONAL) INTRA-PACKET DATA HEADER (MSLW)	

Figure 10-53. Recording event intra-packet header.

i. Event Packet Entry Format. Figures 10-54 and 10-55 present the general recording event packet format and recording event entry layout.

PACKET HEADER
(Optional) PACKET SECONDARY HEADER
CHANNEL SPECIFIC DATA
INTRA-PACKET TIME STAMP FOR EVENT 1
(Optional) INTRA-PACKET DATA HEADER FOR EVENT 1
RECORDING EVENT 1
INTRA-PACKET TIME STAMP FOR EVENT 2
(Optional) INTRA-PACKET DATA HEADER FOR EVENT 2
RECORDING EVENT 2
:
INTRA-PACKET TIME STAMP FOR EVENT n
(Optional) INTRA-PACKET DATA HEADER FOR EVENT n
RECORDING EVENT n
PACKET TRAILER

Figure 10-54. General recording event packet format.

msb 31 29	28	27 12	lsb 11 0
RESERVED	EO	EVENT COUNT	NUMBER

Figure 10-55. Recording event entry layout.

- Reserved. (bits 31-29) Reserved for future growth and shall be zero filled.
- Event Occurrence (EO). (bit 28) Indicates Event Occurrence State.
  - 0 = Indicates the event occurred after the .STOP command and before the .RECORD command.
  - 1 = Indicates the event occurred after the .RECORD command and before the .STOP command.
- Event Count. (bits 27-12) An unsigned binary that identifies the count of up to 65,535 occurrences of an individually defined event (as defined by Event Number in the following paragraph). Event occurrence counts shall begin at 0x0 for the first occurrence of an individual event type (identified by the Event Number). The event count can roll over to 0x0 and begin to count up again. The event count applicability is for each Event Period of Capture as defined in paragraph 10.6.7.3c. The Event Count shall start from 0x0 at the beginning of each Event Period of Capture incrementing at n+0x1 to 0xFFFF for each event occurrence. If the event count reaches 0xFFFF within the Event Period of Capture it shall roll over to 0x0.
- Event Number. (bits 11-0) An unsigned binary that identifies 4096 individual events types defined in the corresponding setup record Recording Event Number. The Event Number shall begin at 0x0 for the first event

type defined in the setup record and increment n+1 for the next event type defined in the setup record.

10.6.7.4 Computer Generated Data Packets Format 3, Recording Index. This defines an index packet for an individual recording file used for direct access into the recording file. Recording Index packets will be enabled when Recording Event packets are enabled. There are two types of index packets:

NOTE	Recording Index packets *will be enabled* when Recording Event packets are *enabled*.

- Root Index Packets: contain zero based byte offset entries that are the beginning of Node Index packets. The last entry will be an offset to the beginning of the previous root index packet if there are more than one Root Index packets, or to the beginning of the Root Index packet itself, if this root index packet is either the first root index packet of the recording or the only root index packet. Root Index packets will not contain Filler in the Packet Trailer and will contain a 32-bit Data Checksum in the Packet Trailer.

NOTE	Root Index packets shall not contain Filler in the Packet Trailer and shall contain a 32-bit Data Checksum in the Packet Trailer.

NOTE	Each recording file with indexes enabled shall have at a minimum one (1) Root Index Type packet.

- Node Index Packets: contain Node Items structures containing information about the location of data packets throughout the recording.

NOTE	At a minimum, an index entry *shall exist* for each Time Data Packet in the recording and, at a minimum, an index entry *shall exist* for each Recording Event Packet in the recording.

NOTE	If the Recording Index Type uses a Count rather than Time, the Time Data Packets and Computer Generated Data Packets are not included in the count interval. If the Recording Index Type uses a Time rather than Count, the Time Data Packets are not included in the time interval. If the Time Count value coincides with the Time Packet rate, then one index entry shall be created.

NOTE	If the Recording Indexes are *enabled* the Computer Generated Data Packet Format 1 Setup Record Count or Time interval value cannot be zero.

a. Recording Index Packet Location. If indexes are enabled, a Root Index packet (Figure 10-56) will be the last packet in any recording file. More than one (1) Root Index Type packet may be created and may be located within the recording. Root Index packets are not required to be contiguous. Node Index Types may be placed at any location within the recording after the first Time Data packet and before the last Root Index packet. This may be at an interval of time or packets. If Indexes are based on a time interval, the time interval shall be referenced to and based on 10MHz RTC counts. When a time-based index time interval expiration takes place and all packet(s) are open (not generated), the index offset and time stamp will be based on the first of the opened packets generated. Packet generation and packet generation time shall apply per the definitions in paragraph 10.2.

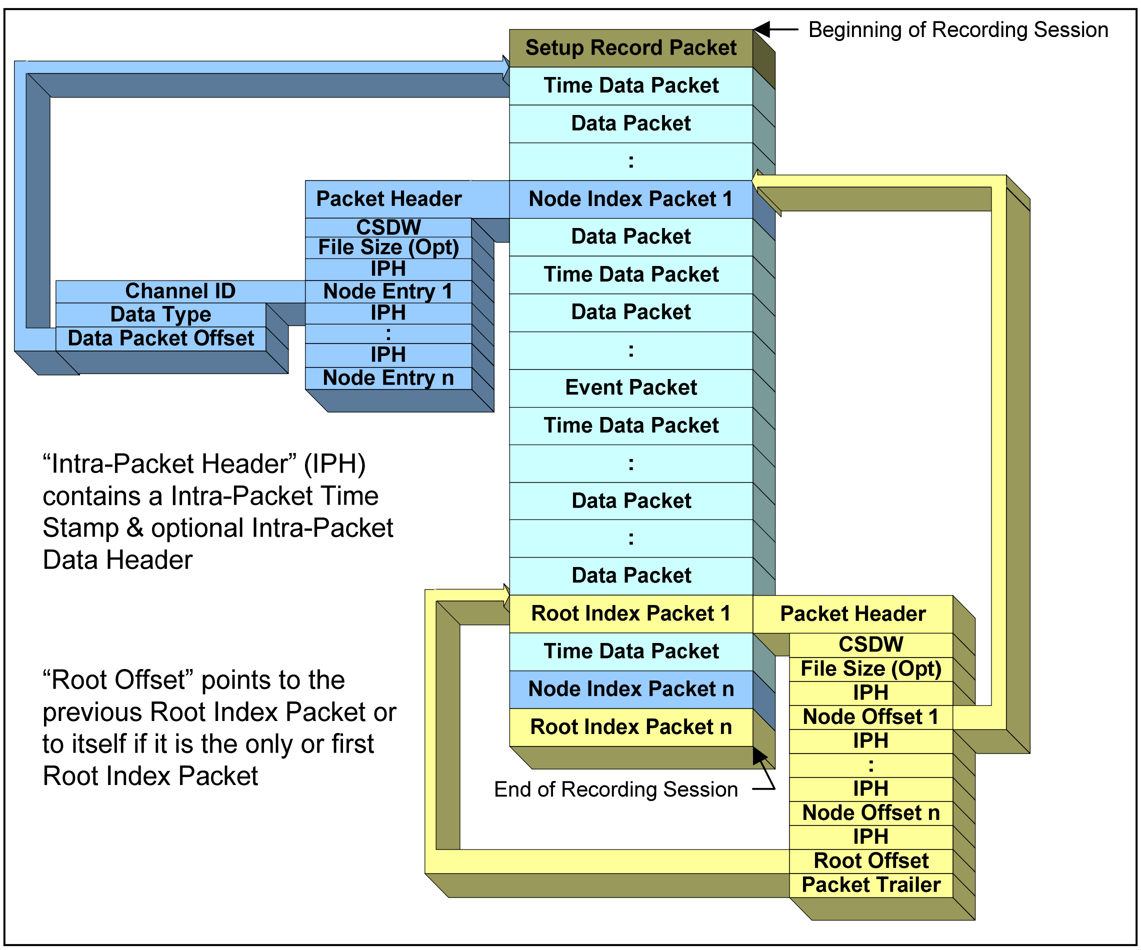


Figure 10-56. Format showing root index packet.

b. Channel Specific Data Word. The Packet Body portion of each Format 3 Packet begins with the Channel Specific Data word, which is formatted as shown in Figure 10-57.

msb 31	30	29	28 16	15 lsb 0
IT	FSP	IPDH	RESERVED	INDEX ENTRY COUNT

Figure 10-57. Channel specific data word format.

- Index Type (IT). (bit 31) indicates the type of index packet.
  0 = Root Index
  1 = Node Index
- File Size Present (FSP). (bit 30) indicates if the file size at the time the index packet was created is present.
  0 = File Size Not Present
  1 = File Size Present
- Index Intra-Packet Data Header (IPDH). (bit 29) indicates the presence of the Intra-Packet Data Header.
  0 = Index Intra-Packet Data Header Not Present
  1 = Index Intra-Packet Data Header Present
- Reserved. (bits 28-16) are reserved.
- Index Entry Count. (bits 15-0) indicates the unsigned binary value of the number of index entries included in the packet. An integral number of complete index entries will be in each packet.

NOTE	The Intra-Packet Data Header (IPDH) presence once set by bit 29 shall be the same state for the entire recording.

c. Recording Index Intra-Packet Time Stamp. (8 Bytes) indicates the time tag of the Recording Index Entry as follows:

- The 48-bit Relative Time Counter that corresponds to the Index Entry, with bits 31 to 16 in the second long word zero filled. For Node Index Packets this corresponds to the first bit in the packet body of the packet associated with the Node Index Item; or
- The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format corresponds to the time format indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and to the Index Entry. For Node Index Packets this corresponds to the first bit in the packet body of the packet associated with the Node Index Item.

d. (Optional) Recording Index Intra-Packet Data Header. (8 Bytes) contains the absolute time of the Index Entry. The time source and format shall be derived from the Time Data Packet, Format 1. Unused high-order bits will be zero filled as needed, depending on the time type of the Time Data Packet. Figure 10-58 presents the format of the recording index intra-packet header.

msb 31	lsb 0
INTRA-PACKET TIME STAMP (LSLW)	
INTRA-PACKET TIME STAMP (MSLW)	
(Optional) INTRA-PACKET DATA HEADER (LSLW)	
(Optional) INTRA-PACKET DATA HEADER (MSLW)	

Figure 10-58. Recording index intra-packet header.

e. Root Index Packet Entry Format. A Root Index Packet contains a Node Index Offset entry or entries using the format shown in Figure 10-59 and Figure 10-60.

- (Optional) Root Index File Size. (8 Bytes) An unsigned binary that identifies the current size in bytes of the file at the time the Root Index Packet was created and placed into the recording. This value should be the same as the Root Index Offset. The file size is required when a recording is split across multiple media, individual or multiple channels are split from the original recording file, or time slices are extracted from the original recording. In all cases the original recording file size will allow recalculation and/or replacement of the index offsets when required.

- Node Index Offset. (8 Bytes) An unsigned binary that identifies the zero based byte offset from the beginning of the recording file to the point in the file at which the Node Index Packet Sync Pattern (0xEB25) begins.

- Root Index Offset. (8 Bytes) An unsigned binary that identifies the zero based byte offset from the beginning of the recording file to the point in the file at which the previous Root Index Packet begins, if there are more than one (1) Root Index Packets or to itself, if it is the first or only Root Index Packet.

PACKET HEADER
(Optional) PACKET SECONDARY HEADER
CHANNEL SPECIFIC DATA
(Optional) ROOT INDEX FILE SIZE
INTRA-PACKET TIME STAMP FOR NODE INDEX 1
(Optional) INTRA-PACKET DATA HEADER FOR NODE INDEX 1
NODE INDEX OFFSET 1
:
INTRA-PACKET TIME STAMP FOR NODE INDEX n
(Optional) INTRA-PACKET DATA HEADER FOR NODE INDEX n
NODE INDEX OFFSET n
INTRA-PACKET TIME STAMP FOR ROOT INDEX
(Optional) INTRA-PACKET DATA HEADER FOR ROOT INDEX
ROOT INDEX OFFSET
PACKET TRAILER

Figure 10-59. General recording root index packet.

msb 31	lsb 0
(Optional) FILE SIZE (LSLW)	
(Optional) FILE SIZE (MSLW)	
INTRA-PACKET TIME STAMP FOR NODE INDEX 1 (LSLW)	
INTRA-PACKET TIME STAMP FOR NODE INDEX 1 (MSLW)	
(Optional) INTRA-PACKET DATA HEADER FOR NODE INDEX 1 (LSLW)	
(Optional) INTRA-PACKET DATA HEADER FOR NODE INDEX 1 (MSLW)	
NODE INDEX OFFSET 1 (LSLW)	
NODE INDEX OFFSET 1 (MSLW)	
:	
INTRA-PACKET TIME STAMP FOR NODE INDEX n (LSLW)	
INTRA-PACKET TIME STAMP FOR NODE INDEX n (MSLW)	
(Optional) INTRA-PACKET DATA HEADER FOR NODE INDEX n (LSLW)	
(Optional) INTRA-PACKET DATA HEADER FOR NODE INDEX n (MSLW)	
NODE INDEX OFFSET n (LSLW)	
NODE INDEX OFFSET n (MSLW)	
INTRA-PACKET TIME STAMP FOR ROOT INDEX (LSLW)	
INTRA-PACKET TIME STAMP FOR ROOT INDEX (MSLW)	
(Optional) INTRA-PACKET DATA HEADER FOR ROOT INDEX (LSLW)	
(Optional) INTRA-PACKET DATA HEADER FOR ROOT INDEX (MSLW)	
ROOT INDEX OFFSET (LSLW)	
ROOT INDEX OFFSET (MSLW)	

Figure 10-60. Recording root index entry layout.

f. Node Index Packet Entry Format. A Node Index Packet contains an index entry or entries using the format shown in Figure 10-61 and Figure 10-62.

PACKET HEADER
(Optional) PACKET SECONDARY HEADER
CHANNEL SPECIFIC DATA
(Optional) NODE INDEX FILE SIZE
INTRA-PACKET TIME STAMP FOR NODE INDEX 1
(Optional) INTRA-PACKET DATA HEADER FOR NODE INDEX 1
RECORDING NODE INDEX 1
INTRA-PACKET TIME STAMP FOR NODE INDEX 2
(Optional) INTRA-PACKET DATA HEADER FOR NODE INDEX 2
RECORDING NODE INDEX 2
:
INTRA-PACKET TIME STAMP FOR NODE INDEX n
(Optional) INTRA-PACKET DATA HEADER FOR NODE INDEX n
RECORDING NODE INDEX n
PACKET TRAILER

Figure 10-61. General recording node index packet.

msb 31 24	23 16	15 lsb 0
RESERVED	DATA TYPE	CHANNEL ID
DATA PACKET OFFSET (LSLW)		
DATA PACKET OFFSET (MSLW)		

Figure 10-62. Recording node index entry layout.

- (Optional) Node Index File Size. (8 Bytes) An unsigned binary that identifies the current size in bytes of the file at the time the Node Index Packet was created and placed into the recording. This value should be the same as the Node Index Offset. The file size is required when a recording is split across multiple media, individual or multiple channels are split from the original recording file, or time slices are extracted from the original recording. In all cases the original recording file size will allow recalculation and/or replacement of the index offsets when required.

- Channel ID. (2 Bytes) An unsigned binary that identifies a value representing the Packet Channel ID for the data packet associated with this Node Index Item.

- Data Type. (1 Byte) An unsigned binary that identifies a value representing the type and format of the data packet associated with this Node Index Item.

- Data Packet Offset. (8 Bytes) An unsigned binary that identifies the zero based byte offset from the beginning of the recording file to the point in the file at which the Data Packet Sync Pattern (0xEB25) begins for this Node Index Packet item.

10.6.8 ARINC-429 Data Packets.

10.6.8.1 ARINC-429 Data Packets, Format 0. Data shall be packetized in Word Mode: each 32-bit word of an ARINC-429 bus shall be preceded by an Intra-Packet Header containing an Intra-Packet Data Header only with an identifier (ID Word) that provides type and status information. The Intra-Packet Header does not contain an Intra-Packet Time Stamp. The Packet Time in the Packet Header is the time of the first ARINC data word in the packet, and the time of successive ARINC data words is determined from the first word time using the gap times in the ID words that precede each of the data words. Multiple words of multiple ARINC-429 buses can be inserted into a single packet. The resultant packets shall have the following format as shown in Figure 10-63.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
WORD 1 INTRA PACKET DATA HEADER	
WORD 1 INTRA PACKET DATA HEADER	
ARINC-429 DATA WORD 1 (BITS 15-0)	
ARINC-429 DATA WORD 1 (BITS 31-16)	
WORD 2 INTRA PACKET DATA HEADER	
WORD 2 INTRA PACKET DATA HEADER	
ARINC-429 DATA WORD 2 (BITS 15-0)	
ARINC-429 DATA WORD 2 (BITS 31-16)	
:	
WORD n INTRA PACKET DATA HEADER	
WORD n INTRA PACKET DATA HEADER	
ARINC-429 DATA WORD n (BITS 15-0)	
ARINC-429 DATA WORD n (BITS 31-16)	
PACKET TRAILER	

Figure 10-63. ARINC-429 data packet format.

NOTE

Time tagging of ARINC-429 shall correspond to the first data bit of the packet.

a. ARINC-429 Packet Channel Specific Data Word. The Packet Body portion of each ARINC-429 data packet shall begin with a Channel Specific Data word formatted as shown in Figure 10-64.

msb 31 16	15 lsb 0
RESERVED	MSGCOUNT

Figure 10-64. ARINC-429 packet channel specific data word format.

- Reserved. (bits 31-16) are reserved
- Message Count (MSGCOUNT). (Bits 15-0) indicates the binary value of the number of ARINC-429 words included in the packet.

b. Intra-Packet Data Header. (bits 31-0) contains the ARINC-429 ID Word. Each ARINC-429 bus data word is preceded by an identification word and the bit definitions are as shown in Figure 10-65.

msb 31 24	23	22	21	20	19 lsb 0
BUS	FE	PE	BS	R	GAP TIME

Figure 10-65. Intra-packet data header format.

- Bus. (bits 31-24) a binary value identifying the ARINC-429 bus number associated with the following data word (0 indicates the first bus. A maximum of 256 buses can be placed in one packet).
- Format Error (FE). (bit 23) indicates an ARINC-429 Format Error.
  0 = No format error has occurred
  1 = Format error has occurred
- Parity Error (PE). (bit 22) indicates an ARINC-429 Parity Error.
  0 = No parity error has occurred
  1 = Parity error has occurred
- Bus Speed (BS). (bit 21) indicates the ARINC-429 bus speed the data is from.
  0 = Indicates Low-Speed ARINC-429 bus (12.5 kHz)
  1 = Indicates High-Speed ARINC-429 bus (100 kHz)
- Reserved (R). (bit 20) is reserved.
- Gap Time (GAP TIME). (bits 19-0) contains a binary value that represents the gap time from the beginning of the preceding bus word (regardless of bus) to the beginning of the current bus word in 0.1 microsecond increments. The gap time of the first word in the packet is GAP TIME=0. When the gap time is longer than 100 milliseconds, a new packet must be started.

c. ARINC-429 Packet Data Words. ARINC-429 Data: The data words shall be inserted into the packet in the original 32-bit format as acquired from the bus.

10.6.9 Message Data Packets.

10.6.9.1 Message Data Packets, Format 0. The data from one or more separate serial communication interface channels can be placed into a Message Data Packet (Figure 10-66).

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 15–0)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 31–16)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 47-32)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 63-48)	
INTRA-PACKET DATA HEADER FOR MSG 1 (BITS 15-0)	
INTRA-PACKET DATA HEADER FOR MSG 1 (BITS 31-16)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
:	
INTRA-PACKET TIME STAMP FOR MSG n (BITS 15–0)	
INTRA-PACKET TIME STAMP FOR MSG n (BITS 31–16)	
INTRA-PACKET TIME STAMP FOR MSG n (BITS 47-32)	
INTRA-PACKET TIME STAMP FOR MSG n (BITS 63-48)	
INTRA-PACKET DATA HEADER FOR MSG n (BITS 15-0)	
INTRA-PACKET DATA HEADER FOR MSG n (BITS 31-16)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
PACKET TRAILER	

Figure 10-66. Message data packet format.

a. Message Packet Channel Specific Data Word. The Packet Body portion of each Message Data Packet begins with a Channel Specific Data word. It indicates if the Packet Body contains several short messages (Type: Complete) or one segment of a long message (Type: Segmented).

b. Complete Message Channel Specific Data Word. The Channel Specific Data word is formatted for the Complete type of packet body as shown in Figure 10-67.

msb 31	18	17 16	15	lsb 0
RESERVED		TYPE	COUNTER	

Figure 10-67. Complete message channel specific data word format.

- Reserved. (bits 31-18) are reserved.
- Type. (bits 17-16) indicates the type of Serial Packet.
  00 = One or more complete messages
  01 = Reserved
  10 = Reserved
  11 = Reserved
- Counter. (bits 15-0) contains a binary value indicating the number of messages included in the packet.

c. Segmented Message Channel Specific Data Word. The Channel Specific Data word is formatted for the Segmented type of packet body as shown in Figure 10-68.

msb 31	18	17 16	15	lsb 0
RESERVED		TYPE	COUNTER	

Figure 10-68. Segmented message channel specific data word format.

- Reserved. (bits 31-18) are reserved.
- Type. (bits 17-16) indicates the type of Serial Packet.
  00 = Reserved
  01 = Packet is a beginning of a long message from a single source
  10 = Whole packet is the last part of a long message from a single source
  11 = Whole packet is a middle part of a long message from a single source
- Counter. (bits 15-0) contains a binary value indicating the segment number of a long message. The number must start with 1 and must be incremented by one after each packet. The maximum length of a single long message is 4 GBytes (combined with the 16-bit Message Length field, see description at paragraph 10.6.9.1d below).

d. Message Data Intra-Packet Header. After the Channel Specific Data, Message Data is inserted into the packet. Each Message is preceded by an Intra-Packet Header that has both an Intra-Packet Time Stamp and an Intra-Packet Data Header containing a Message ID Word. The length of the Intra-Packet Header is fixed at 12 bytes (96 bits) positioned contiguously, in the sequence shown in Figure 10-69.

msb 31	lsb 0
TIME (LSLW)	
TIME (MSLW)	
MESSAGE ID WORD	

Figure 10-69. Message data intra-packet header.

- Intra-Packet Time Stamp. (8 Bytes) indicate the time tag of the Message Data. First long word bits 31-0 and second long word bits 31-0 indicate the following values:
  (1) The 48-bit Relative Time Counter that corresponds to the first data bit in the Message with bits 31 to 16 in the second long word zero filled or;
  (2) The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format corresponds to the time format indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and to the first data bit in the Message.

- Intra-Packet Data Header. The Intra-Packet Data Header is an identification word (Message ID Word) that precedes the message and is inserted into the packet with the format shown in Figure 10-70.

msb 31	30	29 16	15 lsb 0
DE	FE	SUBCHANNEL	MESSAGE LENGTH

Figure 10-70. Intra-packet data header format.

- Data Error (DE). (bit 31) indicates bad data bits as determined by parity, checksums, or CRC words received with the data.
  0 = No Data error has occurred
  1 = Data error has occurred

- Format Error (FE). (bit 30) indicates a protocol error, such as out-of-sequence data or length errors.
  0 = No Format Error
  1 = Format Error encountered

- Subchannel. (bit 29-16) contains a binary value that represents the subchannel number belonging to the message that follows the ID Word when the Channel ID in the packet header defines a group of subchannels. Zero means first and/or only subchannel.

- Message Length. (bits 15-0) contains a binary value representing the length of the message in bytes (n) that follows the ID Word. The maximum length of a message (complete) or a message segment (segmented) is 64K bytes.

10.6.10 Video Packets.

10.6.10.1 Video Packets, Format 0 (MPEG-2/H.264). Format 0 MPEG-2/H.264 encoding will be IAW Department of Defense (DoD) Motion Imagery Standards Profile (MISP) Standard 9701– Standard Definition Digital Motion Imagery, Compression Systems. The MPEG-2/H.264 format will be Transport Streams (TS) per MISP Recommended Practice (RP) 0101. The TS will be encapsulated at a Constant Bit Rate (CBR) within the limits of MPEG-2 MP@ML and H.264 MP@L3 specifications per MISP 9720d (Motion Imagery Standards Matrix (MISM), Standard Definition Motion Imagery) for further standardization and telemeter/transmission requirements of the video.

These MPEG-2/H.264 algorithm features are combined to produce an encoded video stream that will be encapsulated in Format 0 packets. H.264 can be carried over the MPEG-2 transport streams using ITU-T Rec. H.222, Amendment 3, 2004: Transport of AVC data over ISO/IEC 13818-1/ H.222.0 for MPEG2 TS containment for MPEG4 AVC. MISP has adapted this with 9720 and 9701.

Transport streams are limited to a single program stream using Program Elementary Stream (PES) packets that share the same common time base. A transport stream must contain the Program Association Table (PAT) and Program Map Table (PMT) that define the Program ID (PID) for the Program Clock Reference (PCR) stream. Program streams also must contain at least one packet header.

A packet with Format 0 MPEG-2/H.264 Video data has the basic structure shown in Figure 10-71. Note that the width of the structure is not related to any number of bits. This figure is merely intended to represent relative placement of data in the packet.

PACKET HEADER
CHANNEL SPECIFIC DATA
(Optional) INTRA-PACKET HEADER
188 BYTE TS DATA
(Optional) INTRA-PACKET HEADER
188 BYTE TS DATA
:
(Optional) INTRA-PACKET TIME HEADER
188 BYTE TS DATA
(Optional) INTRA-PACKET TIME HEADER
188 BYTE TS DATA
PACKET TRAILER

Figure 10-71. General MPEG-2/H.264 video packet, format 0.

a. Video Packet Audio. When recording video using Format 0, if audio is present it will be inserted into the TS per ISO/IEC 13818-3 for MPEG-2 and ISO/IEC 14496-3 for H.264. A separate analog channel to specifically record audio will not be required as MPEG-2/H.264 supports audio insertion into the TS. By combining video and audio, recording bandwidth and memory capacity will be increased.

b. Video Packet Channel Specific Data Word. The packet body portion of each Format 0 packet begins with the Channel Specific Data word, formatted as shown in Figure 10-72.

msb 31	30	29	28	27 24	23	22 lsb 0
ET	IPH	SRS	KLV	PL	BA	RESERVED

Figure 10-72. Video packet channel specific data word format.

- Embedded Time (ET). (bit 31) indicates if embedded time is present in the MPEG-2 video data.
  - 0 = No embedded time present
  - 1 = Embedded time is present

  MPEG-2 stream embedded time if utilized will be IAW MISP Standard 9708 - Embedded Time Reference for Motion Imagery Systems and Standard 9715 - Time Reference Synchronization. Embedded time is used for the synchronization of core MPEG-2 data when extracted from the IRIG-106 Chapter 10 domain (i.e., an export to MPEG-2 files).

- Intra-Packet Header (IPH). (bit 30) indicates if Intra-Packet Time Stamps are inserted before each Transport Packet.
  - 0 = Intra-Packet Times Not Present
  - 1 = Intra-Packet Times Present

- SCR/RTC Sync (SRS). (bit 29) indicates if the MPEG-2 SCR is RTC.
  - 0 = SCR is not synchronized with the 10 MHz RTC
  - 1 = SCR is synchronized with the 10 MHz RTC

  Transport streams contain their own embedded time base used to facilitate the decoding and presentation of video and/or audio data at the decoder. Within a Program stream, all streams are synchronized to a single time source referred to as the System Clock Reference (SCR). Within a Transport stream, each embedded program contains its own PCR, requiring that each Format 0 encoded MPEG-2/H.264 Transport stream contains only a single program allowing each format to be treated in a similar manner using a single global clocking reference.

  The 10 MHz RTC is for the purposes of synchronizing and time stamping the data acquired from multiple input sources. For input sources that don’t define an explicit timing model for data presentation, superimposing this timing model can be accomplished. For MPEG-2/H.264, however, an explicit synchronization model based on a 27MHz clock is defined for the capture, compression, decompression, and presentation of MPEG-2/H.264 data. In order to relate the two different timing models, the MPEG-2/H.264 SCR/PCR timestamps (if enabled) will be derived from the 10 MHz RTC timing reference source (by generating the 27 MHz MPEG-2/H.264 reference clock slaved to the 10 MHz RTC).

MPEG-2/H.264 defines the SCR/PCR timestamp as a 42-bit quantity, consisting of a 33-bit base value and a 9-bit extension value. The exact value is defined as:

SCR = SCR_base * 300 + SCR_ext

where:

SCR_base = ((system_clock_frequency * t) / 300) MOD 2^{33}
SCR_ext = ((system_clock_frequency * t) / 1) MOD 300

For recording periods of less than 26.5 hours, the SCR can be directly converted into the 10 MHz RTC using the equation:

10 MHz RTC time base = SCR * 10 / 27 (rounded to nearest integer)

For recording periods longer than this, the Format 0 packet header time stamp can be used to determine the number of times the MPEG-2/H.264 SCR has rolled over and calculate the upper 8 bits of the free-running counter's value.

- KLV. (bit 28) indicates if KLV Metadata is present in the MPEG-2 video data.
  - 0 = No KLV Metadata present
  - 1 = KLV Metadata is present

MPEG-2/H.264 stream KLV Metadata, if utilized, will be IAW MISP Standard 9711 - Intelligence Motion Imagery Index, Geospatial Metadata, Standard 9712 - Intelligence Motion Imagery Index, Content Description Metadata (Dynamic Metadata Dictionary Structure and Contents), 9713 - Data Encoding Using Key-Length-Value, Recommended Practice 9717 - Packing KLV Packets into MPEG-2 Systems Streams, and Standard 0107 – Bit and Byte Order for Metadata in Motion Imagery Files and Streams.

- PL. (bit 27-24) indicates the payload type within the MPEG-2 stream per MISP Xon2:
  - 0000 = MPEG-2 MP@ML
  - 0001 = H.264 MP@L2.1
  - 0010 = H.264 MP@L2.2
  - 0011 = H.264 MP@L3
  - 0100-1111 = Reserved.

- BA. (bit 23) Byte Alignment indicates the MPEG-2 data payload byte alignment within 16-bit words.

  0 = Little endian as referenced in Figure 10-74.
  1 = Big endian as referenced in Figure 10-75.

- Reserved. (bits 22-0) are reserved.

c. Intra-Packet Header. If enabled, the Intra-Packet Header shall include a 64-bit Intra-Packet Time Stamp, which is inserted immediately before the TS sync pattern. The length of the Intra-Packet Header is fixed at 8 bytes (64 bits) positioned contiguously, in Figure 10-73.

msb 31	lsb 0
TIME (LSLW)	
TIME (MSLW)	

Figure 10-73. Intra-packet header.

- Intra-Packet Time Stamp. (8 Bytes) indicate the time tag of the individual Transport Stream packets. First Long Word (LSLW) bits 31-0 and Second Long Word (MSLW) bits 31-0 indicate the following values:
  (1) The 48-bit Relative Time Counter that will correspond to the first bit of the TS. Bits 31 to 16 in the second long word (MSLW) will be zero filled; or
  (2) The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and the first bit of the TS.

d. Video Packet Data: A Format 0 packet shall contain an integral number of 188 Byte (1,504 bits) TS packets as illustrated in Figure 10-74 and Figure 10-75 depending on Byte Alignment (BA) bit. Intra-Packet Headers can be inserted in Format 0 Video Data Packets. The 10MHz RTC Packet Header time is the time of the first bit of the first TS in the packet.

The CBR of the encoding will be user selectable and within the MPEG-2 MP@ML and H.264 MP@L3 specification. Per ISO/IEC 13818-1 the TS format will be fixed-length 188 byte (1,504 bits) frames containing an 8-bit sync pattern or “sync byte” (starting at bit 0 and ending at bit 7 of the TSF). The sync bytes value is 01000111 (0x47). The rest of the TS 187 data bytes will follow (Figure 10-76).

msb 15	lsb 0
TS SYNC BYTE (BITS 0 TO 7)	TS DATA (BITS 8 TO 15)
TS DATA (BITS 16 TO 23)	TS DATA (BITS 24 TO 31)
:	
TS DATA (BITS 1488 TO 1495)	TS DATA (BITS 1496 TO 1503)

Figure 10-74. Format 0 MPEG-2/H.264 video frame sync and word format, 16-bit little-endian aligned.

msb 15	lsb 0
TS DATA (BITS 8 TO 15)	TS SYNC BYTE (BITS 0 TO 7)
TS DATA (BITS 24 TO 31)	TS DATA (BITS 16 TO 23)
:	
TS DATA (BITS 1496 TO 1503)	TS DATA (BITS 1488 TO 1495)

Figure 10-75. Format 0 MPEG-2/H.264 video frame sync and word format, 16-bit big-endian (native) aligned.

msb 15 — lsb 0
PACKET HEADER
CHANNEL SPECIFIC DATA (BITS 15-0)
CHANNEL SPECIFIC DATA (BITS 31-16)
(Optional) INTRA-PACKET TIME STAMP
TS SYNC BYTE DATA (BITS 15 TO 0)
TS DATA (BITS 31 TO 16)
:
TS DATA (BITS 1487 TO 1472)
TS DATA (BITS 1503 TO 1488)
(Optional) INTRA-PACKET TIME STAMP
TS SYNC BYTE DATA (BITS 15 TO 0)
TS DATA (BITS 31 TO 16)
:
TS DATA (BITS 1487 TO 1472)
TS DATA (BITS 1503 TO 1488)
:
(Optional) INTRA-PACKET TIME STAMP
REPEAT FOR EACH TS
:
PACKET TRAILER

Figure 10-76. Format 0 MPEG-2/H.264 Video Data Packet (Example is 16-bit Aligned).

10.6.10.2 Video Packets, Format 1 (ISO 13818-1 MPEG-2 Bit Stream). Unlike Video Packets, Format 0 (MPEG-2) the Format 1 packets encapsulate the complete MPEG-2 ISO/IEC 13818-1:2000 bit streams for both Program and Transport with constant or variable bit rates. Also any of the Profiles and Level combinations as set forth by MPEG-2 ISO/IEC 13818-1:2000 may be utilized in the encoding process. Transport streams are limited to a single program stream using Program Elementary Stream (PES) packets that share the same common time base. A transport stream must contain the Program Association Table (PAT) and Program Map Table (PMT) that define the Program ID (PID) for the Program Clock Reference (PCR) stream. Program streams also must contain at least one pack header.

a. MPEG-2 Stream Packet Body. The Format 1 packet with n MPEG-2 packets has the basic structure shown in Figure 10-77. Note that the width of the structure is not related to any number of bits. This drawing is merely intended to represent relative placement of data in the packet.

PACKET HEADER
CHANNEL SPECIFIC DATA
(Optional) INTRA-PACKET HEADER
MPEG-2 Packet 1
(Optional) INTRA-PACKET HEADER
MPEG-2 Packet 2
:
(Optional) INTRA-PACKET HEADER
MPEG-2 Packet n
PACKET TRAILER

Figure 10-77. General MPEG-2 video packet, format 1.

b. Video Packet Audio. When recording video using Format 1, if audio is present it will be inserted into the TS per ISO/IEC 13818-3. A separate analog channel to specifically record audio will not be required as MPEG-2 supports audio insertion into the TS or PS. By combining video and audio, recording bandwidth and memory capacity will be increased.

c. MPEG-2 Channel Specific Data Word. The Packet Body portion of each MPEG-2 bit stream begins with a Channel Specific Data word formatted as shown in Figure 10-78.

msb 31 22	21	20	19	18 15	14	13	12	11 lsb 0
RESERVED	KLV	SRS	IPH	EPL	ET	MD	TP	PC

Figure 10-78. MPEG-2 channel specific data word format.

- Reserved. (bits 31-22) are reserved for future use.

- KLV. (bit 21) indicates if KLV Metadata is present in the MPEG-2 video data.
  0 = No KLV Metadata present
  1 = KLV Metadata is present.

MPEG-2 stream KLV Metadata (if utilized) will be IAW MISP Standard 9711 - Intelligence Motion Imagery Index, Geospatial Metadata, Standard 9712 - Intelligence Motion Imagery Index, Content Description Metadata (Dynamic Metadata Dictionary Structure and Contents), 9713 - Data Encoding Using Key-Length-Value, Recommended Practice 9717 - Packing KLV Packets into MPEG-2 Systems Streams, and Standard 0107 - Bit and Byte Order for Metadata in Motion Imagery Files and Streams.

- SCR/RTC Sync (SRS). (bit 20) indicates whether the MPEG-2 SCR is RTC.
  0 = SCR is not synchronized with the 10 MHz RTC.
  1 = SCR is synchronized with the 10 MHz RTC.

Transport streams contain their own embedded time base used to facilitate the decoding and presentation of video and/or audio data at the decoder. Within a Program stream, all streams are synchronized to a single time source referred to as the System Clock Reference (SCR). Within a Transport stream, each embedded program contains its own PCR, requiring that each Format 1 encoded MPEG-2 Transport stream contain only a single program allowing each format to be treated in a similar manner using a single global clocking reference.

The 10 MHz RTC is used to synchronize and time stamp the data acquired from multiple input sources. For input sources that don't define an explicit timing model for data presentation, superimposing this timing model can be accomplished. For MPEG-2, however, an explicit synchronization model based on a 27 MHz clock is defined for the capture, compression, decompression, and presentation of MPEG-2 data. In order to relate the two different timing models, the MPEG-2 SCR/PCR timestamps (if enabled) will be derived from the 10 MHz RTC timing reference source (by generating the 27 MHz MPEG-2 reference clock slaved to the 10 MHz RTC).

MPEG-2 defines the SCR/PCR timestamp as a 42-bit quantity, consisting of a 33-bit base value and a 9-bit extension value. The exact value is defined as:

SCR = SCR_base * 300 + SCR_ext

where:

SCR_base = ((system_clock_frequency * t) / 300) MOD 233
SCR_ext = ((system_clock_frequency * t) / 1) MOD 300

For recording periods of less than 26.5 hours, the SCR can be directly converted into the 10 MHz RTC using the equation:

10 MHz RTC time base = SCR * 10 / 27 (rounded to the nearest integer)

For recording periods longer than this, the Format 1 packet header time stamp can be used to determine the number of times the MPEG-2 SCR has rolled over and calculate the upper 8 bits of the free-running counter's value.

- Intra-Packet Header (IPH). (bit 19) indicate whether Intra-Packet Time Stamps are inserted before each Program or Transport Packet.

- Encoding Profile and Level (EPL). (bits 18-15) indicate the MPEG-2 Profile and Level of the encoded bit stream.
  - 0000 = SimpleProfile@MainLevel
  - 0001 = MainProfile@LowLevel
  - 0010 = MainProfile@MainLevel
  - 0011 = MainProfile@High-1440Level
  - 0100 = MainProfile@HighLevel
  - 0101 = SNRProfile@LowLevel
  - 0110 = SNRProfile@MainLevel
  - 0111 = SpatialProfile@High-1440Level
  - 1000 = HighProfile@MainLevel
  - 1001 = HighProfile@High-1440Level
  - 1010 = HighProfile@HighLevel
  - 1011 = 4:2:2Profile@MainLevel
  - 1100 = Reserved
  - 1101 = Reserved
  - 1110 = Reserved
  - 1111 = Reserved

- Embedded Time (ET). (bit 14) indicates whether embedded time is present in the MPEG-2 video data.
  - 0 = No embedded time present
  - 1 = Embedded time is present

MPEG-2 stream embedded time, if utilized, will be IAW MISP Standard 9708 - Embedded Time Reference for Motion Imagery Systems and Standard

9715 - Time Reference Synchronization. Embedded time is used for the synchronization of core MPEG-2 data when extracted from the IRIG-106 Chapter 10 domain (i.e., an export to MPEG-2 files).

- Mode (MD). (bit 13) indicates whether the MPEG-2 bit stream was encoded using a variable or constant bit rate parameter setting.
  - 0 = Constant Bit Rate stream
  - 1 = Variable Bit Rate stream
- Type (TP). (bit 12) indicates the type of data the packetized MPEG-2 bit stream contains.
  - 0 = Transport data bit stream
  - 1 = Program data bit stream
- Packet Count (PC). (bits 11-0) indicate the binary value of the number of MPEG-2 packets included in the Format 1 packet.

An integral number of complete packets will be in each Format 1 packet. If the MPEG-2 hardware implementation is unable to determine the value of this number, the value of 0 is used by default. If TYPE=0, then this number represents the number of Transport stream packets within the Format 1 packet. If TYPE=1, then this number represents of the number of Program stream packs within the Format 1 packet.

d. Intra-Packet Header. If enabled, the Intra-Packet Header shall include a 64-bit Intra-Packet Time Stamp, which is inserted immediately before the MPEG-2 packet (transport or program). The length of the Intra-Packet Header is fixed at 8 bytes (64 bits) positioned contiguously, in the following sequence (Figure 10-79):

msb 31	lsb 0
TIME (LSLW)	
TIME (MSLW)	

Figure 10-79. Intra-packet header.

- Intra-Packet Time Stamp. (8 Bytes) indicate the time tag of the individual MPEG-2 packets (transport or program). First Long Word (LSLW) bits 31-0 and Second Long Word (MSLW) bits 31-0 indicate the following values:
  - o The 48-bit Relative Time Counter that will correspond to the first bit of the MPEG-2 packet (transport or program). Bits 31 to 16 in the second long word (MSLW) will be zero filled; or
  - o The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format is indicated by bits 2 and 3 in the Packet Flags paragraph 10.6.1.1.g) and the first bit of the MPEG-2 packet (transport or program).

10.6.10.3 Video Packets, Format 2 (ISO 14496 MPEG-4 Part 10 AVC/H.264). Format 2 video encoding will be IAW ISO 14496 MPEG-4 Part 10 Advanced Video Coding (AVC). The carrier format for Format 2 AVC/H.264 will be MPEG-2 ISO/IEC 13818-1:2000 bit streams for both Program and Transport with constant or variable bit rates. AVC/H.264 can be carried over the MPEG-2 streams IAW ITU-T Rec. H.222, Amendment 3, 2004: Transport of AVC data over ITU-T Rec. / H.222.0 | ISO/IEC 13818-1 streams.

Unlike Format 0 Video Packets (MPEG-2\H.264), which only support a fixed MPEG-2 transport and fixed MPEG-2/H.264 profiles and levels, the Format 2 AVC/H.264 packets encapsulate the complete MPEG-2 transport/program streams, provide for a fixed/variable bit rate (Format 1), and include all H.264 video encoding profiles and levels.

Format 2 AVC/H.264 streams are limited to a single program or transport stream using Program Elementary Stream (PES) packets that share the same common time base. The transport or program stream must contain the Program Association Table (PAT) and Program Map Table (PMT) that define the Program ID (PID) for the Program Clock Reference (PCR) stream. Program streams also must contain at least one pack header.

a. AVC/H.264 Stream Packet Body. The Format 2 packet with n AVC/H.264 packets has the basic structure shown in Figure 10-80. Note that the width of the structure is not related to any number of bits. This drawing is merely intended to represent relative placement of data in the packet.

PACKET HEADER
CHANNEL SPECIFIC DATA
(Optional) INTRA-PACKET HEADER
AVC/H.264 Packet 1
(Optional) INTRA-PACKET HEADER
AVC/H.264 Packet 2
:
(Optional) INTRA-PACKET HEADER
AVC/H.264 Packet n
PACKET TRAILER

Figure 10-80. General AVC/H.264 video packet, format 2.

b. Video Packet Audio. When recording video using Format 2 AVC/H.264, if audio is present it will be inserted into the per ISO/IEC 13818-3 Audio or 13818-7 AAC. A separate analog channel to specifically record audio will not be required as AVC/H.264 supports audio insertion into the AVC/H.264 transport stream. By combining video and audio, recording bandwidth and memory capacity will be increased.

c. AVC/H.264 Channel Specific Data Word. The Packet Body portion of each AVC/H.264 packet begins with a Channel Specific Data word formatted as shown in Figure 10-81.

msb 31 27	26	25 22	21	20	19	18 15	14	13	12	11 lsb 0
R	AET	EL	KLV	SRS	IPH	EP	ET	MD	TP	PC

Figure 10-81. AVC/H.264 channel specific data word format.

- Reserved (R). (bits 31-26) are reserved for future use.
- AVC/H.264 Audio Encoding Type (AET). (bit 26) indicate the AVC/H.264 audio encoding type.
  0 = ISO/IEC 13818-3 Audio
  1 = ISO/IEC 13818-7 AAC
- AVC/H.264 Encoding Level (EL). (bits 25-22) indicate the AVC/H.264 Level of the encoded video bit stream.

0000 = 1	0001 = 1b	0010 = 1.1	0011 = 1.2	0100 = 1.3
0101 = 2	0110 = 2.1	0111 = 2.2	1000 = 3	1001 = 3.1
1010 = 3.2	1011 = 4	1100 = 4.1	1101 = 4.2	1110 = 5
1111 = 5.1				

- KLV. (bit 21) indicates if KLV Metadata is present in the MPEG-2 video data.
  0 = No KLV Metadata present
  1 = KLV Metadata is present.

  MPEG-2 stream KLV Metadata (if utilized) will be IAW MISP Standard 9711 - Intelligence Motion Imagery Index, Geospatial Metadata, Standard 9712 - Intelligence Motion Imagery Index, Content Description Metadata (Dynamic Metadata Dictionary Structure and Contents), 9713 - Data Encoding Using Key-Length-Value, Recommended Practice 9717 - Packing KLV Packets into MPEG-2 Systems Streams, and Standard 0107 – Bit and Byte Order for Metadata in Motion Imagery Files and Streams.

- SCR/RTC Sync (SRS). (bit 20) indicates whether the AVC/H.264 MPEG-2 SCR is RTC.
  0 = SCR is not synchronized with the 10 MHz RTC.
  1 = SCR is synchronized with the 10 MHz RTC.

  Transport streams contain their own embedded time base used to facilitate the decoding and presentation of video and/or audio data at the decoder. Within a Program stream, all streams are synchronized to a single time source referred to as the System Clock Reference (SCR). Within a Transport stream, each embedded program contains its own PCR, requiring that each Format 0 encoded MPEG-2 Transport stream contain only a single program allowing each format to be treated in a similar manner using a single global clocking reference.

The 10 MHz RTC is provided to synchronize and time stamp the data acquired from multiple input sources. For input sources that don't define an explicit timing model for data presentation, superimposing this timing model can be accomplished. For MPEG-2, however, an explicit synchronization model based on a 27 MHz clock is defined for the capture, compression, decompression, and presentation of MPEG-2 data. In order to relate the two different timing models, the MPEG-2 SCR/PCR timestamps (if enabled) will be derived from the 10 MHz RTC timing reference source (by generating the 27 MHz MPEG-2 reference clock slaved to the 10 MHz RTC).

MPEG-2 defines the SCR/PCR timestamp as a 42-bit quantity, consisting of a 33-bit base value and a 9-bit extension value. The exact value is defined as:

SCR = SCR_base * 300 + SCR_ext

where:

SCR_base = ((system_clock_frequency * t) / 300) MOD 233
SCR_ext = ((system_clock_frequency * t) / 1) MOD 300

For recording periods of less than 26.5 hours, the SCR can be directly converted into the 10 MHz RTC using this equation:

10 MHz RTC time base = SCR * 10 / 27 (rounded to nearest integer).

For recording periods longer than this, the Format 0 packet header time stamp can be used to determine the number of times the MPEG-2 SCR has rolled over and calculate the upper 8 bits of the free-running counter's value.

- Intra-Packet Header (IPH). (bit 19) indicate whether Intra-Packet Time Stamps are inserted before each Program or Transport Packet.

- AVC/H.264 Encoding Profile (EP). (bits 18-15) indicate the AVC/H.264 Profile of the encoded video bit stream.
  0000 = Baseline Profile (BP) 0001 = Main Profile (MP)
  0010 = Extended Profile (EP) 0011 = High Profile (HiP)
  0100 = High 10 Profile (Hi10P) 0101 = High 4:2:2 Profile (Hi422P)
  0110 = High 4:4:4 Profile (Hi444P) 0111 – 1111 = Reserved

- Embedded Time (ET). (bit 14) indicates whether embedded time is present in the AVC/H.264 MPEG-2 video data.

  0 = No embedded time present
  1 = Embedded time is present.

AVC/H.264 MPEG-2 stream embedded time (if utilized) will be IAW MISP Standard 9708 - Embedded Time Reference for Motion Imagery Systems and Standard 9715 - Time Reference Synchronization. Embedded time is used for the synchronization of core AVC/H.264 data when extracted from the IRIG-106 Chapter 10 domain, i.e., an export to AVC/H.264 files.

- Mode (MD). (bit 13) indicates whether the AVC/H.264 MPEG-2 bit stream was encoded using a variable or constant bit rate parameter setting.
  - 0 = Constant Bit Rate stream
  - 1 = Variable Bit Rate stream

- Type (TP). (bit 12) indicates the type of data the packetized AVC/H.264 MPEG-2 bit stream contains.
  - 0 = Transport data bit stream
  - 1 = Program data bit stream

- Packet Count (PC). (bits 11-0) indicate the binary value of the number of AVC/H.264 packets included in the Format 2 packet.

  An integral number of complete packets will be in each Format 2 packet. If the AVC/H.264 hardware implementation is unable to determine the value of this number, the value of 0 is used by default. If TYPE=0, then this number represents the number of Transport stream packets within the Format 2 packet. If TYPE=1, then this number represents of the number of Program stream packs within the Format 2 packet.

d. Intra-Packet Header. If enabled, the Intra-Packet Header shall include a 64-bit Intra-Packet Time Stamp, which is inserted immediately before the AVC/H.264 packet (transport or program). The length of the Intra-Packet Header is fixed at 8 bytes (64 bits) positioned contiguously, in the following sequence (Figure 10-82):

msb 31	lsb 0
TIME (LSLW)	
TIME (MSLW)	

Figure 10-82. Intra-packet header.

- Intra-Packet Time Stamp. (8 Bytes) indicates the time tag of the individual AVC/H.264 packets (transport or program). First Long Word (LSLW) bits 31-0 and Second Long Word (MSLW) bits 31-0 indicate the following values:
  - The 48-bit Relative Time Counter that will correspond to the first bit of the AVC/H.264 packet. Bits 31 to 16 in the second long word (MSLW) will be zero filled; or
  - The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format indicated by bits 2 and 3 in the

Packet Flags (paragraph 10.6.1.1.g) and the first bit of the AVC/H.264 packet.

10.6.11 Image Packets.

10.6.11.1 Image Packets, Format 0 (Image Data). A Format 0 Image Packet (Figure 10-83) shall contain one or more fixed-length segments of one or more video images. The channel specific data word for an image packet identifies the number of segments in the packet and the portion of the image or images contained in the packet. If the optional Intra-Packet Header is not included with each segment, the Relative Time Counter in the packet header is the time of the first segment in the packet.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
OPTIONAL INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 15–0)	
OPTIONAL INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 31–16)	
OPTIONAL INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 47-32)	
OPTIONAL INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 63-48)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
:	
OPTIONAL INTRA-PACKET HEADER FOR SEGMENT n (BITS 15–0)	
OPTIONAL INTRA-PACKET HEADER FOR SEGMENT n (BITS 31–16)	
OPTIONAL INTRA-PACKET HEADER FOR SEGMENT n (BITS 47-32)	
OPTIONAL INTRA-PACKET HEADER FOR SEGMENT n (BITS 63-48)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
PACKET TRAILER	

Figure 10-83. Image packet, format 0.

a. Image Packet Channel Specific Data Word. The Packet Body portion of each Image Packet begins with a Channel Specific Data word. It defines the byte length of each segment and indicates if the Packet Body contains several complete images or partial images, and whether or not the Intra-Packet Data Header precedes each segment (Figure 10-84).

msb 31 30	29 28	27	26 lsb 0
PARTS	SUM	IPH	LENGTH

Figure 10-84. Image packet channel specific data word format.

- Parts. (bit 31-30) indicates which piece or pieces of the video frame are contained in the packet.

  00 = Packet does not contains first or last segment of image
  01 = Packet contains first segment of image
  10 = Packet contains last segment of image
  11 = Packet contains both first and last segment of image

- Sum. (bit 29-28) indicates if the packet contains a partial image, one complete image, multiple complete images, or pieces from multiple images.

  00 = Packet contains less than one complete image
  01 = Packet contains one complete image
  10 = Packet contains multiple complete images
  11 = Packet contains multiple incomplete images

- Intra-Packet Header (IPH). (bit 27) indicates whether the Intra-Packet Header (Time Stamp) precedes each segment of the image.

  0 = Intra-Packet Header not enabled
  1 = Intra-Packet Header enabled

- Length. (bits 26-0) indicates a binary value that represents the byte length of each segment.

b. Image Intra-Packet Header. After the Channel Specific Data, Format 0 Image Data is inserted into the packet. Each block of data is optionally preceded by an Intra-Packet Header as indicated by the IPH bit in the Channel Specific Data word. When included, the Intra-Packet Header consists of an Intra-Packet Time Stamp only. The length of the Intra-Packet Header is fixed at 8 bytes (64 bits) positioned contiguously, in the following sequence (Figure 10-85).

msb 31	lsb 0
TIME (LSLW)	
TIME (MSLW)	

Figure 10-85. Image data intra-packet header, format 0.

- Intra-Packet Time Stamp. (8 Bytes) indicate the time tag of the Format 0 Image Data. First long word bits 31-0 and second long word bits 31-0 indicate the following values:
  - o The 48-bit Relative Time Counter that corresponds to the first data bit in the first byte with bits 31 to 16 in the second long word zero filled; or
  - o The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format corresponds to the time format

indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and to the first data bit in the Message.

10.6.11.2 Image Packets, Format 1 (Still Imagery). A Format 1 Image Packet (Figure 10-86) shall contain one or more fixed-length segments of a partial Still Image, one complete Still Image, or multiple Still Images. The Still Image source can be external or internal to the recorder. The Still Image formats will be specified in the Channel Specific Data Word and in the Computer Generated Data, Format 1 Setup Record for each Still Imagery channel. Only one format can be contained within each channel ID for Still Imagery.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 15-0)	
INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 31-16)	
INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 47-32)	
INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 63-48)	
INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 79-64)	
INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 95-80)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
:	
INTRA-PACKET HEADER FOR SEGMENT n (BITS 15-0)	
INTRA-PACKET HEADER FOR SEGMENT n (BITS 31-16)	
INTRA-PACKET HEADER FOR SEGMENT n (BITS 47-32)	
INTRA-PACKET HEADER FOR SEGMENT n (BITS 63-48)	
INTRA-PACKET HEADER FOR SEGMENT 1 (BITS 79-64)	
INTRA-PACKET HEADER FOR SEGMENT n (BITS 95-80)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
PACKET TRAILER	

Figure 10-86. Still Imagery packet, format 1.

a. Still Imagery Packet Channel Specific Data Word. The Packet Body portion of each Still Image Packet begins with a Channel Specific Data word. It defines the format of the Still Imagery Format (which will coincide with the Still Imagery format with the Setup Record), and indicates if the Packet Body contains several complete images or partial images (Figure 10-87).

msb 31 30	29 28	27	26 23	22 lsb 0
PARTS	SUM	IPH	FORMAT	RESERVED

Figure 10-87. Still Imagery packet channel specific data word format.

- Parts. (bit 31-30) indicates which piece or pieces of the image are contained in the packet.

  00 = Packet does not contains first or last segment of image
  01 = Packet contains first segment of image
  10 = Packet contains last segment of image
  11 = Packet contains both first and last segment of image

- Sum. (bit 29-28) indicates if the packet contains a partial image, one complete image, multiple complete images, or pieces from multiple images.

  00 = Packet contains less than one complete image
  01 = Packet contains one complete image
  10 = Packet contains multiple complete images
  11 = Packet contains multiple incomplete images

- Intra-Packet Header (IPH). (bit 27) indicates whether the Intra-Packet Header (Time Stamp) precedes each segment of the image.

  0 = Intra-Packet Header not enabled
  1 = Intra-Packet Header enabled

- Format. (bits 26-23) indicates a binary value which represents the Still Image Format.

  0000 = MIL-STD-2500 National Imagery Transmission Format
  0001 = JPEG File Interchange Format
  0010 = JPEG 2000 (ISO/IEC 15444-1)
  0011 = Portable Network Graphics Format
  0100-1111 = Reserved

- Reserved. (bits 22-0) are reserved.

b. Still Imagery Intra-Packet Header. After the Channel Specific Data, Format 1 Still Imagery Data is inserted into the packet. Each still image or segment is preceded by an Intra-Packet Header. The Intra-Packet Header consists of an

Intra-Packet Time Stamp and Intra-Packet Data. The length of the Intra-Packet Header is fixed at 12 bytes (96 bits) positioned contiguously, in the following sequence (Figure 10-88).

msb. 31	lsb 0
TIME (LSLW)	
TIME (MSLW)	
INTRA-PACKET DATA	

Figure 10-88. Still Imagery intra-packet header.

- Intra-Packet Time Stamp. (8 Bytes) indicate the time tag of the Format 1 Still Imagery Data. First long word bits 31-0 and second long word bits 31-0 indicate the following values:
  - The 48-bit Relative Time Counter that corresponds to the first data bit in the still image or segment with bits 31 to 16 in the second long word zero filled or;
  - The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format is indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and the first data bit in the still image or segment.
- Intra-Packet Data. (4 Bytes) indicates a binary value that represents the byte length of following still image or segment.

10.6.12 UART Data Packets.

10.6.12.1 UART Data Packets, Format 0. The data from one or more separate asynchronous serial communication interface channels (RS-232, RS-422, RS-485, etc…) can be placed into a UART Data Packet as shown in Figure 10-89.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
(OPTIONAL) INTRA-PACKET TIME STAMP FOR UART 1 (BITS 15–0)	
(OPTIONAL) INTRA-PACKET TIME STAMP FOR UART 1 (BITS 31–16)	
(OPTIONAL) INTRA-PACKET TIME STAMP FOR UART 1 (BITS 47-32)	
(OPTIONAL) INTRA-PACKET TIME STAMP FOR UART 1 (BITS 63-48)	
INTRA-PACKET DATA HEADER (UART ID) for UART 1 (BITS 15-0)	
INTRA-PACKET DATA HEADER (UART ID) for UART 1 (BITS 31-16)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
:	
(OPTIONAL) INTRA-PACKET TIME STAMP FOR UART n (BITS 15–0)	
(OPTIONAL) INTRA-PACKET TIME STAMP FOR UART n (BITS 31–16)	
(OPTIONAL) INTRA-PACKET TIME STAMP FOR UART n (BITS 47-32)	
(OPTIONAL) INTRA-PACKET TIME STAMP FOR UART n (BITS 63-48)	
INTRA-PACKET DATA HEADER (UART ID) for UART n (BITS 15-0)	
INTRA-PACKET DATA HEADER (UART ID) for UART n (BITS 31-16)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
PACKET TRAILER	

Figure 10-89. UART data packet format.

a. UART Packet Channel Specific Data Word. The Packet Body portion of each UART Data Packet begins with a Channel Specific Data word as shown in Figure 10-90.

msb 31	30	lsb 0
IPH	RESERVED	

Figure 10-90. UART packet channel specific data word format.

- Intra-Packet Header (IPH). (bit 31) indicates whether the Intra-Packet Header Time Stamp is inserted before the UART ID words.

  0 = Intra-Packet Header Time Stamp not enabled
  1 = Intra-Packet Header Time Stamp enabled

- Reserved. (bits 30-0) are reserved.

b. UART Intra-Packet Header. After the Channel Specific Data, UART data is inserted into the packet. Each block of data shall be preceded by an Intra-Packet Header with optional Intra-Packet Time Stamp and a mandatory UART

ID WORD Intra-Packet Data Header. The length of the Intra-Packet Header is either 4 bytes (32 bits) or 12 bytes (96 bits) positioned contiguously, in the following sequence (Figure 10-91).

msb 31	lsb 0
TIME (LSLW)	
TIME (MSLW)	
UART ID WORD	

Figure 10-91. UART data intra-packet header.

- UART Intra-Packet Time Stamp. (8 Bytes) indicate the time tag of the UART Data. First long word bits 31-0 and second long word bits 31-0 indicate the following values:
  - The 48-bit Relative Time Counter that corresponds to the first data bit in the first byte with bits 31 to 16 in the second long word zero filled; or
  - The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format is indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and the first data bit in the Message.
- UART Intra-Packet Data Header. The Intra-Packet Data Header is an identification word (UART ID Word) that precedes the data and is inserted into the packet with the following format. Inclusion of the Intra-Packet Data Header is mandatory and is not controlled by the IPH bit in the Channel Specific Data Word. (Figure 10-92).

msb 31	30	29 16	15 lsb 0
PE	RESERVED	SUBCHANNEL	DATA LENGTH

Figure 10-92. Intra-packet data header format.

  - Parity Error (PE). (bit 31) indicates a Parity Error.
    - 0 = No Parity Error
    - 1 = Parity Error
  - Reserved. (bit 30) is reserved.
  - Subchannel. (bits 29-16) indicates a binary value defining the subchannel number belonging to the data that follows the UART ID Word when the Channel ID in the packet header defines a group of subchannels. Zero means first and/or only subchannel into which the Intra-Packet Data Header is inserted before the UART ID words.
  - Data Length. (bits 15-0) indicate a binary value representing the length of the UART data in bytes (n) that follows the UART ID word.

10.6.13 IEEE-1394 Data Packets.

10.6.13.1 IEEE-1394 Data Packets, Format 0 (IEEE-1394 Transaction). This format applies to IEEE-1394 data as described by IEEE 1394-1995, IEEE 1394a, and IEEE 1394b. IEEE-1394 data is packetized to encapsulate completed transactions between nodes. A packet may contain 0 to 65,536 transactions, but is constrained by the common packet elements size limits prescribed in paragraph 10.6.1. IEEE-1394 packets have the basic structure shown in Figure 10-93. Note that the width of the structure is not related to any number of bits. The figure merely represents relative placement of data within the packet.

PACKET HEADER
CHANNEL SPECIFIC DATA WORD
(Optional) INTRA-PACKET HEADER
(Optional) TRANSACTION DATA
(Optional) INTRA-PACKET HEADER
(Optional) TRANSACTION DATA
:
(Optional) INTRA-PACKET HEADER
(Optional) TRANSACTION DATA
PACKET TRAILER

Figure 10-93. IEEE-1394 data packet, format 0.

a. IEEE-1394 Channel Specific Data Word. The packet body portion (Figure 10-94) of each IEEE-1394 packet shall begin with a Channel Specific Data Word.

msb 31 29	28 25	24 16	15 lsb 0
PBT	SY	RESERVED	TC

Figure 10-94. IEEE-1394 channel specific data word.

- Packet Body Type (PBT). (bits 31-29) indicate the packet body type as follows:

  000 = Type 0
  001 = Type 1
  010 = Type 2
  011 = Reserved
  1xx = Reserved

- Synchronization Code (SY). (bits 28-25) indicate the IEEE-1394 Synchronization Code from the transaction. This field is mandatory for Type 1 packet bodies. Otherwise, it is reserved.

- Reserved. (bits 24-16) are reserved.

- Transaction Count (TC). (bits 15-0) indicate the binary value of the number of transactions encapsulated in the packet. An integral number of complete transactions shall be included in each packet. It is mandatory that Transaction Count be 0 for Type 0 packet bodies and 1 for Type 1 packet bodies.

b. IEEE-1394 Intra-Packet Header. Each Intra-Packet Header shall contain an 8 byte Intra-Packet Time Stamp only. The format of a IEEE-1394 Intra-Packet Header is described by Figure 10-95.

msb 31	lsb 0
INTRA-PACKET TIME STAMP	
INTRA-PACKET TIME STAMP	

Figure 10-95. IEEE-1394 intra-packet header.

- IEEE-1394 Intra-Packet Time Stamp. This frame (8 Bytes) indicates the time tag of the IEEE-1394 transaction that immediately follows it in the packet. Time is coded in accordance with all other Chapter 10 packet formats. Specifically, the first long word bits 31-0 and second long word bits 31-0 indicate the following values:
    - The 48-bit Relative Time Counter that corresponds to the first data bit of the transaction, with bits 31-16 in the second long word zero filled; or
    - The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format is indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and the first data bit of the transaction.

c. IEEE-1394 Data Packet Body Types. Three packet body types are defined for the encapsulation of IEEE-1394 data. Regardless of type, each packet body shall begin with the IEEE-1394 Packet Channel Specific Data Word as described by paragraph 10.6.13.1a above. The packet body type is indicated within the Channel Specific Data Word. Depending on the packet body type, the Channel Specific Data Word is followed by 0 or more transactions. Also, dependent on packet body type, each transaction may be preceded by an Intra-Packet Header.

- IEEE-1394 Packet Body Type 0: Bus Status. Type 0 packet bodies shall contain zero Intra-Packet Headers and zero transactions. The Channel Specific Data Word Transaction Count shall be zero. The packet body shall contain the Channel Specific Data Word immediately followed by a single 32-bit word.

Bus Status events shall be encapsulated by Type 0 packet bodies. The 32-bit word in the packet body shall contain an Event Data Word as indicated in Figure 10-96:

msb 31	 30	lsb 0
RE	RESERVED	

Figure 10-96. IEEE-1394 event data word format.

- RE. (bit 31) RESET - when set, this bit indicates that an IEEE-1394 Bus Reset has occurred.
- RESERVED. (bits 30-0) Reserved.

- IEEE-1394 Packet Body Type 1: Data streaming. Type 1 packet bodies shall encapsulate IEEE-1394 data streaming only. Type 1 packet body data is restricted to that from an IEEE-1394 packet with a Transaction Code of 0xA, be it from an Isochronous Channel or Asynchronous Stream. The packet body shall contain zero Intra-Packet Headers and one transaction. The Channel Specific Data Word Transaction Count shall be one.

  The Channel Specific Data Word is immediately followed by a non-zero number of data bytes. The data bytes shall be exactly those of a single IEEE-1394 data block, excluding the IEEE-1394 packet header and Data Block CRC. Data recorded from the stream shall be known to be valid, insofar as, both the IEEE-1394 Header CRC and Data Block CRC tests have passed. In accordance with the definition of Packet Header Data Length, and accounting for the size of the Channel Specific Data Word, the number of data bytes shall be exactly four less than the value indicated in Data Length. Conversely, the value stored in the Packet Header Data Length shall be the number of bytes in the IEEE-1394 data block plus four. The Synchronization Code (sy) from the stream packet shall be indicated in the Channel Specific Data Word, and the Channel Number shall be indicated in the Packet Header Channel ID.

- IEEE-1394 Packet Body Type 2: General Purpose. Type 2 packet bodies encapsulate complete IEEE-1394 packets, including header and data. Use of Type 2 packet bodies is unrestricted and may encapsulate streaming, non-streaming (IEEE-1394 packets with Transaction Codes other than 0xA), isochronous, and asynchronous data. Multiple IEEE-1394 packets, with differing Transaction Codes and Channel Numbers, may be carried within a single Type 2 packet body.

  The Channel Specific Data Word shall be followed by a non-zero number of completed transactions as indicated by the Channel Specific Data Word Transaction Count. Each transaction shall be preceded by an Intra-Packet Header as defined above for IEEE-1394 Data Packets. Immediately following the Intra-Packet Header, the transaction shall be recorded in its entirety and must be a properly formed IEEE-1394 packet in accordance with the specification in use. The revision of the specification used shall be declared within the accompanying Setup Record packet.

NOTE	All IEEE-1394 packets contain a 4-bit Transaction Code field (tcode). This field indicates the IEEE-1394 specific format of the transaction.

10.6.13.2 IEEE-1394 Data Packets, Format 1 (IEEE-1394 Physical Layer). This format applies to IEEE-1394 data as described by IEEE 1394-1995, IEEE 1394a, and IEEE 1394b. IEEE-1394 data is packetized in Format 1 packets as physical Layer data transfers (IAW Annex J of Standard 1394-1995 and Chapter 17 of Standard 1394b-2002). A packet may contain 0 to 65,536 transfers, but is constrained by the common packet elements size limits prescribed in paragraph 10.6.1. IEEE-1394 packets have the basic structure shown in Figure 10-97 below. Note that the width of the structure is not related to any number of bits. The drawing merely represents relative placement of data within the packet.

PACKET HEADER
CHANNEL SPECIFIC DATA WORD
INTRA-PACKET HEADER
DATA
(Optional) INTRA-PACKET HEADER
(Optional) DATA
:
(Optional) INTRA-PACKET HEADER
(Optional) DATA
PACKET TRAILER

Figure 10-97. IEEE-1394 data packet, format 1.

a. IEEE-1394 Channel Specific Data Word. The packet body portion (Figure 10-98) of each IEEE-1394 packet shall begin with a Channel Specific Data Word.

msB 31 16	15 **lsb** 0
RESERVED	IPC

Figure 10-98. IEEE-1394 channel specific data word format 1.

- Reserved. (bits 31-16) are reserved.
- Intra-Packet Count (IPC). (bits 15-0) indicate the binary value of the number of Intra-Packets encapsulated in the packet. An integral number of complete Intra-Packets shall be included in each packet.

b. IEEE-1394 Format 1 Intra-Packet Header. The Channel Specific Data Word is followed by 1 or more IEEE-1394 transfers. Each transfer starts with an Intra-Packet Header, followed by 0-32780 encapsulated data bytes. The length of the Intra-Packet Header is fixed at 12 bytes (96 bits) positioned contiguously, in the following sequence as shown in Figure 10-99:

msb 31	lsb 0
INTRA-PACKET TIME STAMP	
INTRA-PACKET TIME STAMP	
INTRA-PACKET ID WORD	

Figure 10-99. IEEE-1394 format 1 intra-packet header.

- IEEE-1394 Format 1 Intra-Packet Time Stamp. This frame (8 Bytes) indicates the time tag of the IEEE-1394 transfer that immediately follows it in the packet. Time is coded in accordance with all other Chapter 10 packet formats. Specifically, the first long word bits 31-0 and second long word bits 31-0 indicate the following values:
  - o The Relative Time Counter that corresponds to the first data byte of the transfer, with bits 15-0 in the second long word zero filled or;
  - o Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format is indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and the first data byte of the transfer.

- Message ID Word. The Message ID Word is an identification word that precedes the message and is inserted into the packet as in Figure 10-100.

msb 31 24	23 20	19 18	17	16	lsb 15 0
STATUS BYTE	SPEED	TRFOVF	LBO	RSV	DATA LENGTH

Figure 10-100. Intra-packet data header - message ID word.

  - o Status Byte. (bits 31-24) This byte is the status byte received from the PHY IAW IEEE-1394b Specification.
  - o Transmission Speed (SPEED). (bits 23-20) This field identifies the speed of transmission of the message. (Speed codes IAW IEEE-1394b)
    0000 = S100 A
    0001 = S100 B
    0010 = S200 A
    0011 = S200 B
    0100 = S400 A
    0101 = S400 B
    0111 = S800 B
    1001 = S1600 B
    1010 = S3200 B
    other values are reserved

- Transfer Overflow Error (TRFOVF). (bit 19-18) This field indicates if a transfer synchronization error occurred.

  00 = IEEE1394 flow did not exceed maximum Intra-packet size
  01 = IEEE1394 This IEEE1394 transfer started correctly but longer than the standard transfer length.
  10 = The previous IEEE1394 transfer was in "01" type overflow and this IEEE1394 transfer ended correctly (did not exceed standard transfer length).
  11 = The previous IEEE1394 transfer was in "01" type overflow and this IEEE1394 transfer did not end correctly (exceeds standard transfer length).

  Most of the time, this field shall be "00." Possible combinations are: "01"intra-packet, zero or more; '11"intra-packet; and finally "10"intra-packet.

- LBO. (bits 17) Local Buffer Overflow. If this bit is set, some messages are lost before this transfer due to local buffer overflow.
- Reserved. (bit 16) is reserved.
- Data Length. (bits 15-0) contain a binary value that represents the length of the transfer in bytes (n) that follows the ID Word. The maximum length of a captured data is 4120 for transfers corresponding to asynchronous packets, and 32780 for transfers corresponding to isochronous packets.

  If the Data Length field is not a multiple of 4 bytes, 1-3 fill character (0) is added to maintain the packet structures in 32-bit boundary.

  If the Data Length field contains 0, the Intra-Packet Data is not provided and this word contains only the status byte information.

c. IEEE-1394 Format 1 Packet Body. The packet body shall encapsulate IEEE-1394 isochronous or asynchronous message data. The data bytes shall be exactly those of a single IEEE-1394 physical transmission message, including the IEEE-1394 packet header and Data Block CRC. The Data Length field shall indicate the exact number of total bytes encapsulated in the Message Data.

10.6.14 Parallel Data Packets.

10.6.14.1 Parallel Data Packet, Format 0. Parallel data packets are designed to record data from parallel interfaces (2-128 bit wide) including the industry de facto standard 8-bit DCRsi interface. A single packet can hold data words or special data structures as currently defined for the DCRsi scan format. The exact format selection is defined in the Channel Specific Data Word. The data recorded from a parallel interface shall be placed into a Parallel Data Packet Format 0 as shown in Figure 10-101.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
DATA WORD 1	
:	
DATA WORD n	
PACKET TRAILER	

Figure 10-101. Parallel data packet, format 0.

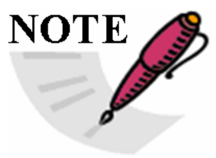
a. Parallel Packet Channel Specific Data Word. The Packet Body portion of each Parallel Data Packet begins with a Channel Specific Data word. The Channel Specific Data word is formatted as shown in Figure 10-102.

msb 31 24	23 lsb 0
TYPE	RESERVED (0) OR SCAN NUMBER

Figure 10-102. Parallel packet channel specific data word format.

- Type. (bits 31-24) indicate the data type stored
  - 0-1: Reserved
  - 2-128: Number of bits of recorded data (parallel data word width in bits)
  - 129-253: Reserved
  - 254: DCRsi scan format, contains auxiliary data, DCRsi main data
  - 255: Reserved
- Scan Number. (bits 23-0) is reserved (0) for general purpose parallel data packets or contains the scan number of the first scan stored in the packet for DCRsi data.

b. General Purpose Parallel Data. General purpose parallel data packets can contain any number of data bytes – as indicated in the Data Length field in the Packet Headers (Figure 10-103).



To get the number of data words stored in the packet, the Data Length must be divided by the number of bytes necessary to hold one Parallel Data word.

- If the number of data bits is less than 9 bits, the word shall be padded to 8-bit bytes.

msb 15			lsb 0
PAD (0)	WORD 2	PAD (0)	WORD 1
	:		:
PAD (0)	WORD N, or PAD (0) IF LENGTH IS ODD	PAD (0)	WORD N-1

Figure 10-103. Parallel data, up to 8-bit wide words.

- If the number of data bits is between 9 and 16, the words shall be padded to one 16-bit word, as in Figure 10-104.

msb 15	lsb 0
PAD (0)	DATA WORD 1
	:
PAD (0)	DATA WORD N

Figure 10-104. Parallel data, 9-16 bit wide words.

- If the number of data bits is greater than 16 the words shall be padded to multiples of 16-bit data words. Figure 10-105 shows storing of 28-bit data words.

msb 15	lsb 0
	DATA WORD1, LS BITS 15-0
PAD (0)	DATA WORD 1, MS BITS 27-16
	:
	DATA WORD N, LS BITS 15-0
PAD (0)	DATA WORD N, MS BITS 27-16

Figure 10-105. Parallel data, (Example: 28-bit wide words).

c. DCRsi Parallel Data Packets. The DCRsi data packets can contain any number of complete DCRsi Scans – containing nine auxiliary data and 4356 main data bytes. The number of the scans can be calculated from the Data Length field of the Packet Header. The structure of one DCRsi scan is in Figure 10-106.

msb 15	lsb 0
AUXILIARY DATA 2	AUXILIARY DATA 1
PAD (0)	AUXILIARY DATA 3
AUXILIARY DATA 5	AUXILIARY DATA 4
PAD (0)	AUXILIARY DATA 6
AUXILIARY DATA 8	AUXILIARY DATA 7
PAD (0)	AUXILIARY DATA 9
DATA BYTE 2	DATA BYTE 1
DATA BYTE 4	DATA BYTE 3
:	:
DATA BYTE 4356	DATA BYTE 4355

Figure 10-106. DCRsi Scan, 9-auxiliary data byte +4356 bytes.

The length of the packet can be only N * (12 + 4356) + 4 bytes, including the length of the Channel Specific data word.

DCRsi data without auxiliary data bytes can be stored also as 8-bit General Purpose Parallel Data as described in paragraph 10.6.15.1b (General Purpose Parallel Data).

10.6.15 Ethernet Data Packets.

10.6.15.1 Ethernet Data Packets, Format 0. Data from one or more Ethernet network interfaces can be placed into an Ethernet Data Packet Format 0 as shown in Figure 10-107.

msb 15	lsb 0
PACKET HEADER	
CHANNEL SPECIFIC DATA (BITS 15-0)	
CHANNEL SPECIFIC DATA (BITS 31-16)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 15–0)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 31–16)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 47-32)	
INTRA-PACKET TIME STAMP FOR MSG 1 (BITS 63-48)	
INTRA-PACKET DATA HEADER FOR MSG 1 (BITS 15-0)	
INTRA-PACKET DATA HEADER FOR MSG 1 (BITS 31-16)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
:	
INTRA-PACKET TIME STAMP FOR MSG n (BITS 15–0)	
INTRA-PACKET TIME STAMP FOR MSG n (BITS 31–16)	
INTRA-PACKET TIME STAMP FOR MSG n (BITS 47-32)	
INTRA-PACKET TIME STAMP FOR MSG n (BITS 63-48)	
INTRA-PACKET DATA HEADER FOR MSG n (BITS 15-0)	
INTRA-PACKET DATA HEADER FOR MSG n (BITS 31-16)	
BYTE 2	BYTE 1
:	:
FILLER (IF n IS ODD)	BYTE n
PACKET TRAILER	

Figure 10-107. Ethernet Data packet format 1.

a. Ethernet Data Packet Format 0, Channel Specific Data Word. The Packet Body portion of each Ethernet Data Packet begins with a Channel Specific Data word. It indicates how many Physical Ethernet messages (MAC Frame) are placed in the Packet Body. The Channel Specific Data word is formatted for the complete type of packet body as shown in Figure 10-108.

msb 31 28	27 16	15 lsb 0
FORMAT	RESERVED	NUMBER OF FRAMES

Figure 10-108. Ethernet Data packet format 1 channel specific data word.

- Format. (bits 31-28) indicate the type of Ethernet Packet.

  0000 = Ethernet Physical Layer IEEE-802.3
  0001 - 1111 = Reserved

- Reserved. (bits 27-16) are reserved.
- Number of Frames. (bits 15-0) contain a binary value that represents the number of frames included in the packet.

b. Ethernet Data Packet Format 0 Intra-Packet Header. After the Channel Specific Data, Ethernet Data is inserted into the packet. Each Frame is preceded by an Intra-Packet Header that has both an Intra-Packet Time Stamp and an Intra-Packet Data Header containing a Frame ID Word. The length of the Intra-Packet Header is fixed at 12 bytes (96 bits) positioned contiguously, in the following sequence as shown in Figure 10-109.

msb   lsb 31   0
TIME (LSLW)
TIME (MSLW)
FRAME ID WORD

Figure 10-109. Ethernet Data format 1 intra-packet header.

- Intra-Packet Time Stamp. (8 Bytes) indicate the time tag of the Frame Data. First long word bits 31-0 and second long word bits 31-0 indicate the following values:
  - The 48-bit Relative Time Counter that corresponds to the first data bit in the Frame with bits 31 to 16 in the second long word zero filled or;
  - The Absolute Time, if enabled by bit 6 in the Packet Flags (paragraph 10.6.1.1.g). Time format is indicated by bits 2 and 3 in the Packet Flags (paragraph 10.6.1.1.g) and the first data bit in the Frame.

- Frame ID Word. The Frame ID Word is an identification word that precedes the Ethernet frame and is inserted into the packet with the format shown in Figure 10-110.

msb 31	30	29 28	27 24	23 16	15 14	13 lsb 0
RESERVED	FE	CONTENT	SPEED	NET ID	RESERVED	DATA LENGTH

Figure 10-110. Intra-packet frame ID word.

- Reserved. (bit 31) is reserved.
- Frame Error (FE). (bit 30). The Frame Error bit is used to indicate any error that occurred during frame transmission.
  - 0 = No Frame Error
  - 1 = Frame Error encountered
- Captured Data Content (CONTENT). (bits 29-28). This field specifies the extent of the captured MAC Frame.
  - 00 = Full MAC Frame: starting with the 6-byte Destination MAC address and ending with the 4-byte Frame Check Sequence
  - 01-11 = Reserved for further formats
- Ethernet Speed (SPEED). (bits 27-24). This field indicates the negotiated bit rate for the identified NETID on which the frame was captured.
  - 0000 = Auto
  - 0001 = 10 Mbit/s
  - 0010 = 100 Mbit/s
  - 0011 = 1 Gbit/s
  - 0100 = 10 Gbit/s
- Network Identifier (NETID). (bits 23-16) contain a binary value which represents the physical network identification of frame origination that follows the ID Word. Zero means first and/or only physical network.
- Reserved. (bits 15-14) are reserved.
- Data Length. (bits 13-0) contain a binary value that represents the length of the frame in bytes (n) that follows the ID Word.

10.7 Recorder Control and Status

10.7.1 Recorder Control. The recorder shall be controlled by either discrete control/status lines and/or serial communication ports. The serial interface shall consist of both RS-232 and RS-422 full duplex serial communications.

10.7.1.1 Optional Recorder Control. The recorder may be controlled over the Fibre Channel, IEEE-1394B, or Ethernet recorder download interface ports from paragraph 10.4. These interfaces shall support communications using SCSI (Fibre Channel) IAW paragraph 10.4.1, SCSI over SBP-2 (IEEE-1394B) IAW 10.4.2, or iSCSI(Ethernet) IAW paragraph 10.4.3 Recorder login and IRIG-106 Chapter 6 Command and Control Mnemonics shall be transmitted and received using the SCSI ORB structures IAW paragraphs 10.9.3 (as required for IEEE-1394B), 10.9.4, and 10.9.14.

10.7.1.2 Optional TELNET Control. The recorder may be controlled over ETHERNET/ TELNET utilizing recorder control mnemonics as defined in IRIG 106 Chapter 6.

10.7.2 Communication Ports. The RS-232 and RS-422 serial communication ports shall be functional simultaneously without requiring selection of either port. Status requested by either port shall be returned on both ports. Note that unexpected results may occur if commands are issued on both ports simultaneously.

10.7.3 RS-232/422 Port. An RS-232/422 port shall be available at the Download Port.

10.7.4 Commands. Commands received through the serial communication ports shall not override hardwire discrete controls.

10.7.5 Status Requests. Status requests received through the serial communication ports shall not interfere with hardwire controls.

10.7.6 Serial Status. Serial status shall be provided on either serial status request or discrete activation.

10.7.7 Default Interface. Default Interface with user equipment shall utilize the following ASCII serial communication protocol:

a. 38400 baud.
b. One start bit.
c. 8 bit data.
d. No parity.
e. One stop bit.

10.7.8 Serial Commands. The following commands are a subset of the Recorder Command and Control Mnemonics defined in IRIG Standard 106 Chapter 6, paragraph 8, where additional rules regarding command syntax and recorder operation are also specified, along with examples showing the use of each command. The commands are simple ASCII command strings delimited by spaces. All commands begin with an ASCII period (.””) and, with the single exception of the .TMATS command, end with a carriage return and line-feed terminator sequence.

- Case Sensitivity. Commands will not be case sensitive except for location parameter in .PLAY and file name in .RECORD.

10.7.9 Serial Commands. Table 10-9 summarizes the required commands.

TABLE 10-9. COMMAND SUMMARY		
Command	Parameters*	Description
.ASSIGN	[destination-ID] [source-ID]	Assign replay (output) channels to source (input) channels
.BIT		Runs all of the built-in-tests
.CRITICAL	[*n* [*mask*]]	Specify and view masks that determine which of the .HEALTH status bits are critical warnings
.DATE	[start-date]	Specify setting or displaying date from recording device
.DECLASSIFY		Secure erases the recording media
.DISMOUNT		Unloads the recording media
.DUB	[*location*]	Same as .PLAY but with internal clock
.ERASE		Erases the recording media
.EVENT	[*text string*]	Display event table or add event to event table
.FILES		Displays information about each recorded file
.FIND	[*value* [*mode*]]	Display current locations or find new play point
.HEALTH	[*feature*]	Display detailed status of the recorder system
.HELP		Displays table of "dot" commands
.IRIG-106		Returns supported version number of IRIG-106 Recorder Command and Control Mnemonics
.LOOP		Starts record and play in read-after-write mode
.MEDIA		Displays media usage summary
.MOUNT		Powers and enables the recording media
.PAUSE		Pause current replay
.PLAY	[*location*][speed]	Reproduce recorded data of assigned output channels starting at [location] and at [speed]
.PUBLISH	*[keyword] [parameter]*	Configure, start and stop live data over the recorder Ethernet interface
.RECORD	[*filename*]	Starts a recording at the current end of data
.REPLAY	[*endpoint* [*mode*]]	Same as .SHUTTLE but with internal clock
.RESET		Perform software initiated system reset
.RESUME		Resume replay from pause condition
.SETUP	[*n*]	Displays or selects 1 of 16 (0…15) pre-programmed data recording formats
.SHUTTLE	[*endpoint* [*mode*]]	Play data repeatedly from current location to the specified endpoint location using external clock
.STATUS		Displays the current system status
.STOP	[*mode*]	Stops the current recording, playback, or both
.TIME	[*start-time*]	Displays or sets the internal system time
.TMATS	{mode} [*n*]	Write, Read, Save, or Get TMATS file

* Parameters in braces "{}" are required. Parameters in brackets "[]" are optional. When optional parameters are nested ("[xxx [yy]]"), the outer parameter (xxx) must be specified in order to also specify the inner parameter (yy).

10.7.10 Required Discrete Control Functions. Required discrete control functions are noted in Figure 10-111.

Description
RECORD
ERASE
DECLASSIFY
ENABLE
BIT

Figure 10-111. Required discrete control functions.

a. Control and Status Lines. Five contacts for discrete control and five lines for indicating status shall be provided. Grounding a control line (or causing the indicator line to go to ground) referenced to the recorders ground completes the circuit to activate a function as shown in Figure 10-112.

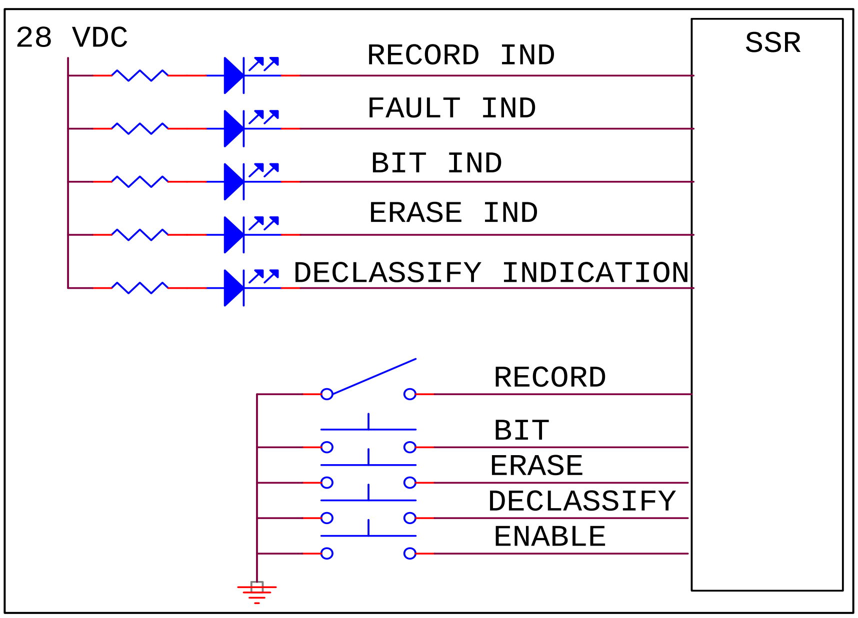


Figure 10-112. Discrete control and indicator functional diagram.

b. Record Command. Activated by toggle switch (Normally closed position 0.55 Volts or less), this discrete commands the recorder to start recording. Recorder will remain in this mode until such time as the switch is set to normally open position.

c. Erase Command. Activated by momentary switch (0.55 Volts or less, minimum duration of 100 ms), this discrete commands the recorder to erase its user data and file directory memory provided the enable switch is also activated.

d. Declassify Command. Activated by momentary switch (0.55 Volts or less, minimum duration of 100 ms), this discrete causes the recorder to start the declassify procedure provided the enable switch is also activated.

e. Command Enable. Activated by momentary switch (0.55 volts or less) for either ERASE or DECLASSIFY discrete to operate.

f. Bit Command. Activated by momentary switch (0.55 Volts or less), this discrete commands the recorder to start the BIT procedure.

g. Record Status. A Record indication (ON) shall be active low 0.55 volts or less. A Non-Record indication (OFF) will be an open circuit. Current limit of 60 milliamps required.

h. BIT Status. A BIT indication (ON) shall be 0.55 volts or less. A Non-BIT indication (OFF) will be an open circuit. Current limit of 60 milliamps required.

i. Fault Status. A Fault indication (ON) shall be 0.55 volts or less. A Non-Fault indication (OFF) will be an open circuit. Current limit of 60 milliamps required.

j. Erase Status. An Erase indication (ON) shall be 0.55 volts or less. A Non-Erase indication (OFF) will be an open circuit. Current limit of 60 milliamps required.

k. Declassify Status. A Declassify indication (ON) shall be 0.55 volts or less. A Non-Declassify indication (OFF) will be an open circuit. No discrete control line shall be available at the Download port. Current limit of 60 milliamps required.

10.7.11 Voltage. 28-VDC auxiliary voltage output shall be provided from the discrete/control port (250 mA max, short circuit protection).

10.7.12 Status Query. Status querying shall be limited to intervals not to exceed 2 seconds and not less than one second.

10.7.13 Erase Command. Activated by momentary switch (.55 Volts or less, minimum duration for 100 milliseconds, if ENABLE discrete is also activated for 100 milliseconds), this discrete commands the recorder to erase its user data and file directory memory provided the enable switch is also activated.

10.7.14 Declassify Command. Activated by momentary switch (0.55 Volts or less, minimum duration of 100 milliseconds, if ENABLE discrete is also activated for 100 milliseconds) this discrete commands the recorder to start the declassify procedure provided the enable switch is also activated.

10.7.15 Command Enable. Activated by a momentary switch (0.55 Volts or less for minimum duration of 100 milliseconds) for either ERASE or DECLASSIFY discrete to operate. In order to activate the ERASE OR DECLASSIFY, the ENABLE and ERASE OR

DECLASSIFY must simultaneously be active for a minimum duration of 100 milliseconds. Table 10-10 presents the Recorder LED states.

TABLE 10-10. RECORDER LED STATES

LED	On	Flash	Off
ERASE	Media erased.	Media erasing is in progress.	Not erased media.
RECORD	In recording	-	Not in recording
FAULT	Recorder is not ready, or any of the critical warning exists.	-	Recording is running properly. No critical warning.
BIT	Built-in test running.	-	Built-in test is not running.
DECLASSIFY	Media declassified.	Media declassification is in progress.	Not declassified media.

Note: Flashing is defined as On: 500 ms, Off: 500 ms

10.8 Declassification

Associated Documents such as NSA-130-2, DOD 5200.28 (1972), and DCI-116 historically covered declassification guidelines/requirements. These documents focused on declassification of standard disk and other conventional memory technologies. Declassification is the determination by an authorized official that classified information no longer requires, in the interest of national security, any degree of protection against unauthorized disclosure. This standard provides for the minimum set of commands that may be utilized to allow for user declassification of Solid State Media residing in an RMM. The Solid State Media may consist of COTS Solid State Disks (SSD) or a memory configuration unique to the manufacturer. There are several approaches for declassification. The responsibility for ensuring that a proper declassification process has been effectively implemented will reside ultimately with the user/customer/program manager.

10.8.1 <u>Approach</u>. The following approaches for declassification are currently recommended. The risk that proper declassification has been effectively implemented will reside ultimately with the user/customer/program manager. It is believed that the user is the most qualified to determine the declassification procedures for any program situation. It is the user’s responsibility to correctly apply the guidelines to the program in each location to optimize the cost/effect while providing appropriate protection for the data. The guidelines are planned to be available on the Internet at Defense Link.

10.8.2 <u>Algorithm</u>. The algorithm to erase secure data is described below. During the secure erase procedure, all blocks of memory shall be processed. No block in memory shall be excluded from secure erase processing for any reason.

a. <u>First Erase</u>. Every memory block on the board is erased. Any erase failures reported by memory chips will result in the corresponding chip/block being declared a bad block. In the event this bad block is not already in the corresponding board's bad block table, a new bad block entry will be appended onto the board's bad block table. Note that this new entry will not have the Secure Erase flag set.

b. First Write (0x55). Every memory chip location is recorded with the pattern 0x55. As each location is written, the data is read back to guarantee that all bits were written to the expected pattern. Any write failures reported by the chips, or any data errors will result in the corresponding chip/block being declared a bad block. In the event this bad block is not already in the corresponding board's bad block table, a new bad block entry will be appended onto the board's bad block table. Note that this new entry will not have the Secure Erase flag set.

c. Second Erase. Every memory chip shall be erased. Any erase failures reported by the memory chips will result in the corresponding chip/block being declared a bad block. In the event this bad block is not already in the corresponding board's bad block table, a new bad block entry will be appended onto the board's bad block table. Note that this new entry will not have the Secure Erase flag set.

d. Second Write (0xAA). Every memory chip location is recorded with the pattern 0xAA. As each location is written, the data is read back to guarantee that all bits were written to the expected pattern. Any write failures reported by the memory chips, or any data errors will result in the corresponding chip/block being declared a bad block. In the event this bad block is not already in the corresponding board's bad block table, a new bad block entry will be appended onto the board's bad block table. Note that this new entry will not have the Secure Erase flag set.

e. Third Erase. Every memory location is erased. Any erase failures reported by the memory chips will result in the corresponding chip/block being declared a bad block. In the event this bad block is not already in the corresponding board's bad block table, a new bad block entry will be appended onto the board's bad block table. Note that this new entry will not have the Secure Erase flag set.

f. Usable Secure Erased Blocks. All blocks that do not have an entry in the bad block table are now considered to be Secure Erased.

g. Unusable Secure Erased Blocks. If a bad block entry contains the flag indicating it has already been Secure Erased, this block has already been secure erased and requires no further processing, since it is known that this block was skipped during the previous recording.

h. Unsecure Bad Block Processing. A board's bad block table may contain bad block entries that have not previously been Secure Erased. If any such entries exist, the following steps are performed on each block.

- Write Zeros Loop. For each page in the block, a pattern of all zeros is written to the page, and the page is checked to determine if any unexpected ones (UOs) are found. If any UOs are found, the page is re-written to all zeros. This process is repeated up to 16 times. After all allowed re-writes, the board, chip, and block numbers of the block containing any remaining UOs are written to a Failed Erase Table.

- Write Ones Loop. For each page in the block, the page is erased (to all ones) and checked to determine if any unexpected zeros (UZs) are found. If any UZs are found, another erase command is issued to the block. This process is repeated up to 16 times. After all allowed erase operations, the board, chip, and block numbers of the block containing any remaining UZs are written to the Failed Erase Table.

i. Failed Erase Table Processing. Any remaining entries in the Failed Erase Table correspond to blocks that cannot be erased. These blocks may still contain user data and therefore are declared to have failed the secure erase.

A count of the number of bad blocks in the Failed Erase Table that have not been Secure Erased is returned as part of the secure erase results. A non-zero count indicates a secure erase failure of at least one block. A command will allow the user to retrieve the Failed Erase Table. A command will also allow a user to retrieve the data from such blocks and manually determine if these blocks can be designated as “Secure Erased.” In most cases, a single stuck bit will not compromise any user data and the offending block can be manually declared to be Secure Erased. If the results of manual inspection are indeterminate, the chip containing the failed block must be removed and destroyed, and the Secure Erase procedure must be repeated.

j. Secure Erase Completion. When all blocks are secure erased (no entries in the Failed Erase Table), the content of the file is the ASCII string “Secure Erase” repeated over and over.

10.9 IEEE 1394B Interface to Recorder Media

This interface definition specifies the interface between the removable media and the host platform as IEEE 1394B. The selection of this protocol was adopted to facilitate a common interface between the media and the computing platform.

NOTE	This definition does not mandate the interface between the recorder and media.

10.9.1 Media Time Synchronization. In order to allow recorders to be synchronized to the same time without requiring platform modification or external time source provided to recorder, the Removable Media Cartridges shall maintain time allowing for time initialization of recorder. Removable media cartridges shall allow for a battery back-up real time clock device. Initialization of time shall be accomplished via IEEE-1394B interface.

10.9.2 Physical and Signaling. The interface shall allow control of Vendor Specific Solid State devices and Commercial Off The Shelf (COTS) Media as per Figure 10-113.

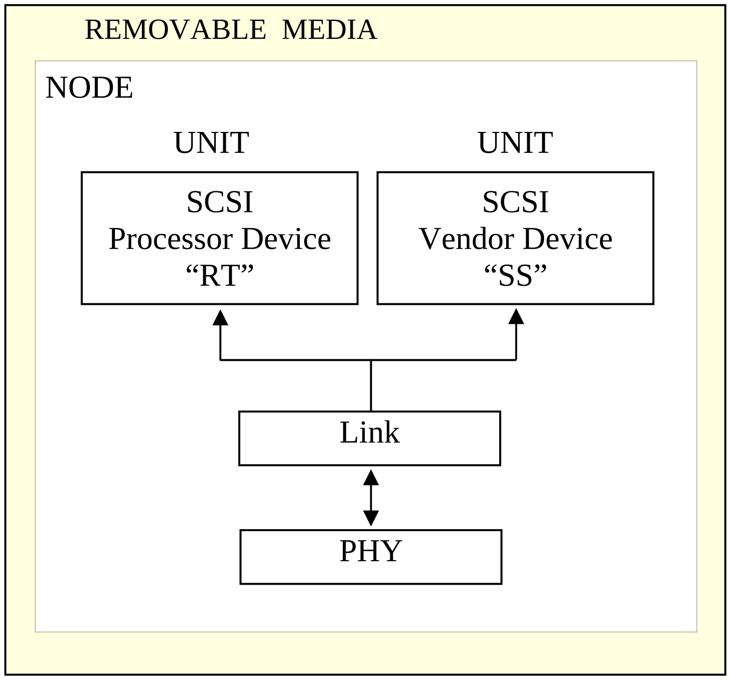
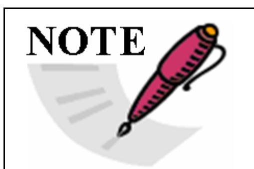


Figure 10-113. Removable media.

10.9.3 Removable Media Communication. The fundamental method of communicating shall be in accordance to the IEEE 1394b protocol. Packets sent and received shall be asynchronous transmissions. IEEE-1394b packets shall encapsulate Serial Bus Protocol (SBP-2) formatted packets for the transport of commands and data. Removable media devices are to use SCSI command set(s) and therefore SCSI commands and status shall be encapsulated in SBP-2 Operation Request Blocks (ORB).



SBP-2 provides for the transport of 6-10- and 12-byte SCSI Common Descriptor Blocks within a command ORB.

10.9.4 Transport of Serial Commands. Removable devices shall implement the SEND and RECEIVE Processor Device SCSI-2 Commands. The IRIG Standard 106 Chapter 6 commands and data will be transported using these SCSI commands and the data buffers.

10.9.5 Mandated IEEE-1394b Interface Connector. The connector type for the removable media shall be a "IEEE 1394b Bilingual Socket" connector. Power for the Removable Media shall be derived from the Bilingual interface connector.

10.9.6 Real Time Clock. Removable media configured with a real time clock shall allow for time to be preset in the media allowing for the transfer to the recorder. SCSI command set shall be utilized to set time on the cartridge.

a. Real Time Clock Time Format. Time format shall be in accordance with paragraph IRIG 106 Chapter 6 paragraph 6.8.4.23. The date format shall be in accordance with ISO 8601.

b. Real Time Clock Logic Unit Number. The standard SCSI Media devices are using Logical Unit Number (LUN) = 0. The Real Time Clock shall be assigned LUN =1.

10.9.7 Mandatory Commands for Processor Devices. The mandatory commands for processor devices are listed in Table 10-11.

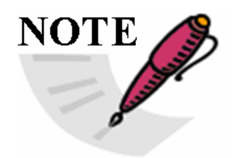
TABLE 10-11. MANDATORY PROCESSOR COMMANDS

Command	Parameters*	Description
.BBLIST		Directs the RMM to retrieve the bad block list
.BBLIST R		Retrieves the bad block list from the RMM
.BBREAD	{block identifier}	Returns contents of specified block in ASCII hexadecimal byte format
.BBREAD P	{block identifier}	Directs the RMM to initiate a physical block read of the specified physical block identifier
.BBREAD D		Retrieves the data from the physical block. See the .MEDIA P command for information. Data is returned in binary format.
.BBSECURE	{block identifier}	Marks an unsecured bad block as secure
.BIT		Runs all of the RMM built-in-tests
.CRITICAL	[*n* [*mask*]]	Specify and view masks that determine which of the .HEALTH status bits are critical warnings
.DATE	[start-date]	Specify setting or displaying date from RMM
.DECLASSIFY		Initiates command as specified by user specification or user CONOP overwrite procedures
.ERASE		Erases the RMM media
.HEALTH	[*feature*]	Display detailed status of the RMM
.IDENTIFY		Queries the RMM for Solid State Memory identification and firmware version
.INITIALIZE		Initializes RMM internal components
.MEDIA P		Queries the RMM for information about the physical media of the RMM and the transfer limits for the required physical I/O commands
.PBWRITE P	{block identifier}	Directs the RMM to initiate a physical block write of the specified physical block identifier
.PBWRITE D		Write the data to the physical block in binary format. See the .MEDIA P command for information.
.SANITIZE		Initiates a memory clear and identification of bad memory blocks
.STATUS		Displays the current RMM status
.TIME	[*start-time*]	Displays or sets the internal system time

10.9.7.1 RMM .HEALTH Command Response. The RMM .HEALTH command response is presented in Figure 10-114.

	BIT	MASK	DESCRIPTION
RMM	0	01	BIT Failure
	1	02	Setup Failure (unable to set the time or date properly)
	2	04	Operational Failure (I/O error, media error, etc)
	3	08	Low or dead battery warning
	4	10	RMM Busy
	5	20	Reserved for future Chapter 10 status bit
	6	40	Reserved for future Chapter 10 status bit
	7	80	Reserved for future Chapter 10 status bit
	8-31		Vendor-Specific Health Status Bits

Figure 10-114. RMM .HEALTH command response.



The operation of these commands is described in Chapter 6, paragraph 8, Command and Control Mnemonics.

10.9.8 Time Setting Requirements. To set time, the .TIME commands should be used according to Chapter 6, paragraph 8, Recorder Command and Control Mnemonics, paragraph 6.8.4.32. To guarantee and avoid uncontrolled delay, the following algorithm shall be used:

a. The Host device puts a .TIME command with time parameter to be set in its SEND buffer and sends it at least 100 ms prior to the correct time to the Real Time Clock device. The delay is necessary to allow the Processor Device to be prepared for the exact time setting and to hold off enough in the Host to force a doorbell with the next SCSI command. Without enough delay the Host will not be able to chain the next SCSI command together with the previous command. If the operating system demands it a delay greater than 100 ms can be used.
b. The Processor Device shall process this time and be prepared to set it at receipt of the doorbell.
c. A SEND command shall be sent to the Real Time Clock with the message .TIME without parameters to ask back the time set.

10.9.9 Set Time. To set time the .TIME commands should be used according to IRIG-106 Chapter 6, paragraph 8 Recorder Command and Control Mnemonics, paragraph 6.8.4.23.

10.9.10 Date Setting Requirements. A .DATE [start-date] command shall be utilized for setting or displaying date of the removable memory real time clock. The date shall be set in year-month-day format according to ISO 8601.

- Date Example.
  .DATE
  DATE 2002-12-31
  *

10.9.11 Checking Battery Status. Verification of health of battery shall be accomplished with .CRITICAL and .HEALTH commands IAW IRIG 106 Chapter 6, paragraph 8. Recorder Command and Control Mnemonics, paragraph 6.8.4.2 and paragraph 6.8.4.10.

10.9.12 Declassification Supporting Commands.

10.9.12.1 .IDENTIFY: A .IDENTIFY command queries the RMM for Solid State Disk (SSD) identification and firmware version.

Description
This command queries the RMM for SSD identification information and SSD firmware version.

Parameters
None

Response
The RMM responds with one line containing five comma separated fields. Characters and spaces are allowed within the comma separated fields. Response time shall be within 100 milliseconds. A .STATUS command request prior to 100 milliseconds shall elicit a BUSY response.

*.IDENTIFY
A, B, C, D, E
*

Where
A … SSD Manufacturer
B … SSD Model
C … SSD Serial Number
D … RMM Firmware Version
E … SSD Firmware Version

10.9.12.2 .MEDIA P: The .MEDIA P command is utilized to query the RMM for information regarding the physical block architecture of the SSD and the SCSI RECEIVE transfer limits in effect when reading physical blocks.

Parameters
The parameter "P" distinguishes this command from the standard .MEDIA command.

Response
The RMM responds with one line containing the tag "PHYSICAL" and five space separated integer numbers. Response time shall be within 100 milliseconds. A .STATUS command prior to 100 msec shall return a BUSY state.

Example

*.MEDIA P
PHYSICAL A B C D E
*

Where …
A … Physical block size in bytes. This value must be a multiple of item D below.
B ... Total number of physical blocks in SSD.
C … Maximum ORB transfer size that can be used when reading the binary data from the physical block with the .BBREAD D and .PBWRITE D commands.
D … Number of valid data bytes in a physical page. Item A above must be an integer multiple of this value.
E … This field specifies the number of filler bytes appended onto each physical page read from the RMM. Filler bytes are typically inserted to pad the transfer to the next ATA sector boundary. If no padding is required, this field may be 0.

10.9.12.3 _.SANITIZE_: A .SANITIZE command shall initiate a write/verify of all RMM user data physical blocks. The pattern may consist of either all "FF's" or all "00's". The .SANITIZE command shall identify any blocks that cannot be written or verified. Blocks that cannot be written to or contain at least one bit that is stuck in either the '0' state or the '1' are termed "bad blocks". The user shall review the block contents and map out the bad blocks such that they are no longer addressable. Once the address has been mapped out the blocks are no longer addressable and are no longer identified in the *bad block table* (Figure 10-115).

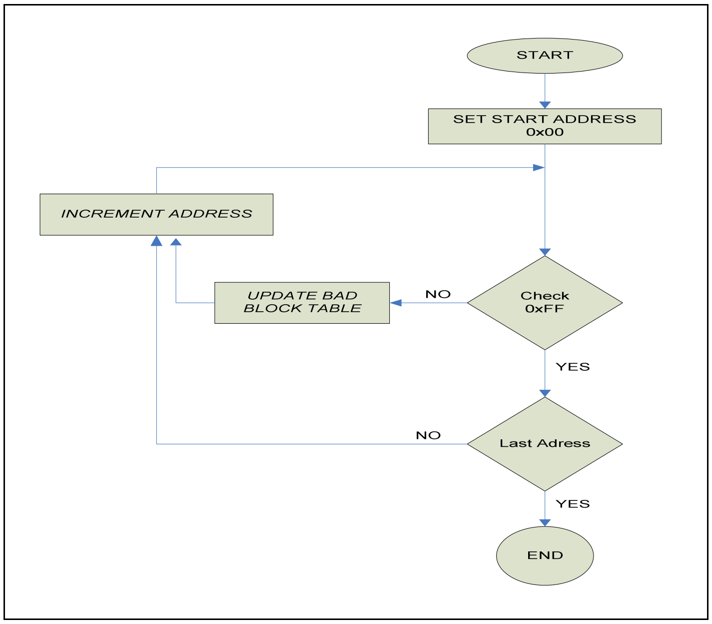


Figure 10-115. Updating the Bad Block Table

Parameters
None

Response
The RMM responds with an asterisk. Response time shall be within 100 milliseconds. A .STATUS command prior to 100 milliseconds shall elicit a BUSY response. During Sanitization the RMM shall respond with “S 04 xx yy zz”; where zz indicates percentage complete, reference the .STATUS command. Upon completion a status response of “S 11 xx yy” shall indicate that bad blocks were found. A status response upon completion of “S 12 xx yy” shall indicate that no bad blocks were found.

Example
*.SANITIZE
*

10.9.12.4 .BBLIST: A .BBLIST command shall be utilized to instruct the RMM to retrieve the list of UNSECURED bad block identifiers from Solid State Media residing in the RMM. A BBLIST command is only valid following a SANITIZE command.

Parameters
None

Response
The RMM responds with an asterisk. Response time shall be within 100 milliseconds. A .STATUS command prior to 100 msec shall return a BUSY state.

Example
*.BBLIST
*

10.9.12.5 .BBLIST R: A .BBLIST R shall be used to retrieve bad block identifiers from the RMM. This command may only be issued immediately following a successful .BBLIST command.

Parameters
The parameter “R” distinguishes this command from the standard .BBLIST command.

Response
The RMM must respond with a list of hexadecimal bad block identifiers. Each identifier must be terminated with a <CR><LF> sequence. Each identifier must be a legal hexadecimal number from 1 to 16 digits. No embedded spaces or other special characters are allowed. Response time shall be within 100 milliseconds. A .STATUS command prior to 100 milliseconds shall return a BUSY state.

Example
*.BBLIST R
000000E3
0000034f
FE0184C9
*

10.9.12.6 .BBREAD P {block_identifier}: A .BBREAD P {block_identifier} shall direct the RMM to initiate a physical block read of the specified physical block identifier.

Parameters
The parameter “P” distinguishes this as a binary physical block read command.
The parameter *block_identifier* is the physical block identifier [from the BBLIST R response] of the block to be read.

Response
The RMM responds with an asterisk. Response time shall be within 100 milliseconds. A .STATUS command prior to 100 msec shall return a BUSY state.

Example
.BBREAD P FE0184C9
*

10.9.12.7 .BBREAD D: A .BBREAD D command shall read one binary physical block from the RMM. This command may only be issued immediately after a successful .BBREAD P command. The size of physical block, page size, page filler size, and maximum SCSI RECEIVE transfer size that are required to perform the transfer are all specified in the RMM’s response to the .MEDIA P command.

Parameters
None.

Response
The RMM responds by returning the requested binary physical block data. Multiple SCSI RECEIVE commands may be required to retrieve the entire physical data block.

*.BBREAD D
Response is in binary.

10.9.12.8 .BBSECURE {*block identifier*}: A .BBSECURE command shall be utilized to mark an unsecured bad block as being secured. A block that has been identified as secured shall never be used for any subsequent data recording. Secured bad blocks shall be removed from the unsecured bad block identifier list. The *block identifier* shall be provided for the block to be secured.

Parameters
None.

Response
The RMM responds with an asterisk.

Example
.BBSECURE 5678
*

10.9.12.9 .PBWRITE P {block_identifier}: A .PBWRITE P {block_identifier} shall direct the RMM to initiate a physical block write of the specified physical block identifier.

Parameters
The parameter *block_identifier* is the physical block identifier [from the BBLIST R response] of the block to be written.

Response
The RMM responds with an asterisk. Response time shall be within 100 milliseconds. A .STATUS command prior to 100 milliseconds shall return a BUSY state.

Example
.PBWRITE P FE0184C9
*

10.9.12.10 .PBWRITE D: A PBWRITE D command shall write one binary physical block to the RMM. This command may only be issued immediately after a successful .PBWRITE P command. The size of physical block transfer size and the maximum SCSI SEND page size that are required to perform the transfer are all specified in the RMM's response to the .MEDIA P command.

Parameters
Binary data block. Multiple SCSI SEND commands may be required to transfer the entire physical data block.

Response
The RMM responds with an asterisk after all data is successfully received.

*.PBWRITE D
<binary data> total length = physical block size.

10.9.12.11 .INITIALIZE: A .INITIALIZE command shall be utilized to configure the RMM memory and reset of the firmware.

Parameters
None

Response
The RMM responds with an asterisk. Response time shall be within 100 milliseconds. A .STATUS command prior to 100 milliseconds shall return a BUSY state. A response of "S13 xx yy zz"; where zz indicates percentage complete shall be provided. Upon completion, a response of "S 14 xx yy" shall be provided; where yy indicates number of seconds required after initialization.

```
*.INITIALIZE
*
.STATUS
S 13 00 00 01%
.STATUS
S 13 00 00 02%
  •
  •
  •
.STATUS
S 13 00 00 100%
.STATUS
S 14 00 03
.STATUS
S 01 00 00
```

10.9.12.12 .DECLASSIFY: A .DECLASSIFY command shall be utilized to initiate user procedures.

Parameters
None

Response
The RMM responds with an asterisk. Response time shall be within 100 milliseconds. A .STATUS command prior to 100 milliseconds shall return a BUSY state. During Sanitization the RMM shall respond with "S 04 xx yy zz"; where zz indicates percentage complete, reference status command. Upon completion a status response of "S 11 xx yy" shall indicate that bad blocks were found. A status response upon completion of "S 12 xx yy" shall indicate that no bad blocks were found.

Example

```
*.DECLASSIFY
*
```

10.9.12.13 .IRIG106: A .IRIG106 shall be utilized to retrieve the IRIG-106 supported version number.

Parameters
None

Response
The RMM responds with a version number which shall be a two integer value. Response time shall be within 100 milliseconds. A .STATUS command prior to 100 milliseconds shall return a BUSY state.

Example
*.IRIG106
9
*

10.9.12.14 .STATUS: A .STATUS shall be utilized to query the RMM for status information (see Table 10-12).

Description
This command queries the RMM for status information.

Parameters
None

Response
The RMM response to a .STATUS command with a response of the form…

*.STATUS
S A B C [D%]
*

TABLE 10-12. RMM STATES				
STATE	Description			
	State CODE (A)	State Value (B)	State Value (C)	Progress Percentage (D)
FAIL	00			
IDLE	01	00	00	
BIT	02	00	00	Percent Complete
ERASE	03	00	00	Percent Complete
DECLASSIFY SANITIZE	04	00	00	Percent Complete
BUSY	09	00	00	
SANITIZE COMPLETED BAD BLOCKS FOUND	11	00	Number of bad blocks found (Integer)	
SANITIZE COMPLETED NO BAD BLOCKS FOUND	12	00	00	
INITIALIZE IN PROGRESS	13	00	00	Percent Complete
INITIALIZE COMPLETE	14	00	Number of seconds required for initialization (Integer)	

10.9.12.15 RMM Command Error Codes. Issuing invalid commands (bad syntax) or illegal commands (not accepted in the current system state) result in error code responses (with an ASCII “E” identifier) prior to the asterisk response terminator when a command cannot be completed. Table 10-13 shows possible error codes and the conditions under which they occur.

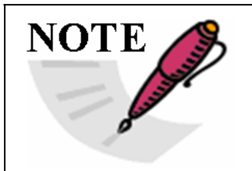
Example
.CLEAR
E 00
*

TABLE 10-13. COMMAND ERROR CODES

Error	Description	Conditions
00	INVALID COMMAND	Command does not exist
01	INVALID PARAMETER	Parameter is out of range, or wrong alpha-numeric type
02	INVALID MODE	Command cannot be executed in the current state
05	COMMAND FAILED	Command failed to execute for any reason other than those listed above

10.9.13 Vendor Specific Devices. The Mandatory SCSI Command Set for Vendor Specific Devices is as follows:

a. For random access devices:
   INQUIRY
   READ
   READ CAPACITY
   TEST UNIT READY
   REQUEST SENSE
b. For sequential access devices:
   INQUIRY
   READ
   REWIND
   TEST UNIT READY
   REQUEST SENSE



COTS media shall support as a minimum the SCSI command set to support data download in accordance with paragraph 10.4 of this standard.

10.9.14 Mandatory ORB Formats for the Processor Device.

10.9.14.1 Minimum Operational Requirements. The time setting accuracy of the Real Time Clock device shall be better than 1 millisecond. The short time accuracy of the Real Time Clock device must be at least 10 ppm in the temperature range 0-40C, and at least 50 ppm in the temperature range –40C - +85C.

10.9.14.2 ORB Format.

a. Login ORB format. The login ORB format is illustrated in Figure 10-116.

<table>
<tr><td colspan="7">most significant</td></tr>
<tr><td colspan="7">password</td></tr>
<tr><td colspan="7">Login_response</td></tr>
<tr><td>n</td><td>Rq_fmt</td><td>x</td><td>Reserved</td><td>reconnect</td><td>function</td><td>LUN</td></tr>
<tr><td colspan="6">password_length</td><td>login_response_length</td></tr>
<tr><td colspan="7">Status_FIFO</td></tr>
<tr><td colspan="7">least significant</td></tr>
</table>

Field and Bit Descriptions		
Password	32 bits	The password shall be "RTC." The password field shall contain the immediate data and the password_length shall be zero.
Login_response login_response_length	64 bits 16 bits	The login_response (32 bits) field and login_response_length (16 bits) field shall specify the address and size of a buffer (minimum of 12 bytes) allocated by the host for the return of the login response.
N	1 bit	The notify bit "n" shall be one.
Rq_fmt	2 bits	The rq_fmt shall be zero.
X	1 bit	The exclusive bit "x" shall be one.
Reserved	4 bits	Reserved shall be zero (0).
Reconnect	4 bits	The reconnect field shall specify the reconnect time as a power of 2 seconds. A value of zero shall mean one second.
Function	4 bits	The Function shall be zero.
LUN	16 bits	The logical unit number shall be one (1).
Status_FIFO	64 bits	The Status_FIFO shall contain the address allocated by the host for the return of status for the LOGIN request and for the return of subsequent write and read buffer response(s) indicating success/failure of the operation

Figure 10-116. Login ORB format.

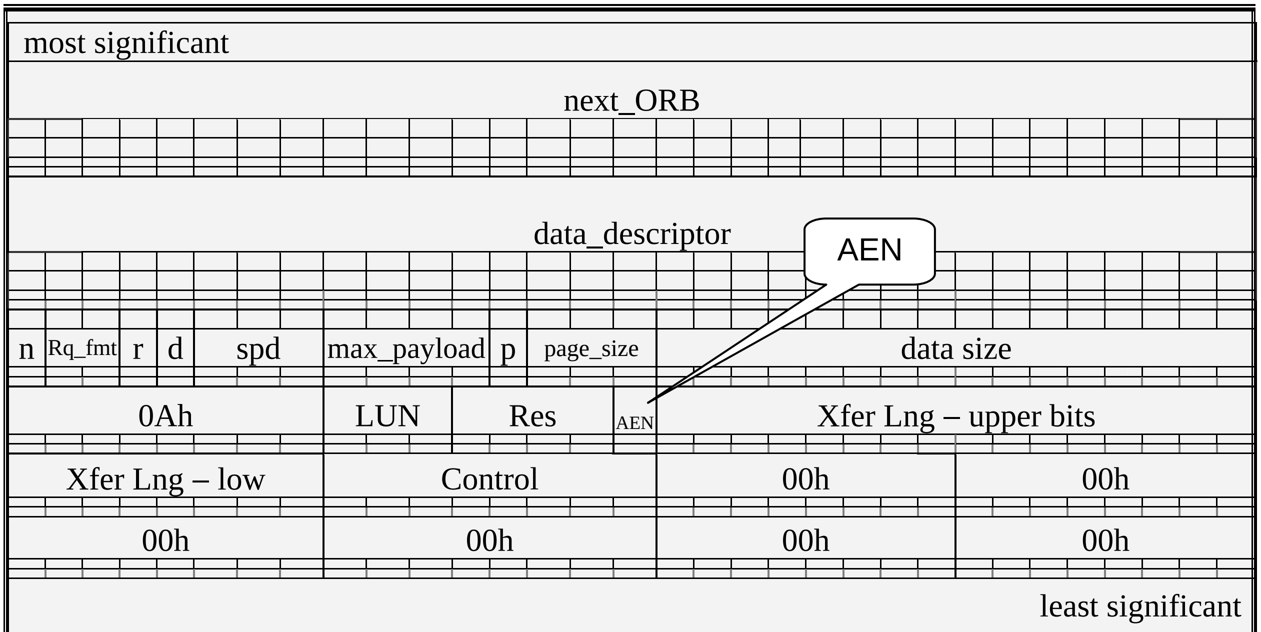
b. Login Response. The Login Response format is illustrated in Figure 10-117.

most significant	
Length	login_ID
command_block_agent	
reserved	reconnect_hold
	least significant

Field and Bit Descriptions		
Length	16 bits	The length, in bytes, of the login response data
login_ID	16 bits	Used in all subsequent requests to the SSMC's Management Agent.
command_block_agent	64 bits	Base address of agent's CSR
Reserved	16 bits	Reserved shall be zero (0).
Reconnect_hold	16 bits	TBD

Figure 10-117. Login response format.

c. Send. The Send command ORB format is illustrated in Figure 10-118, and the Send data buffer format is illustrated in Figure 10-119. The Send data buffer contains the send command (according to Chapter 6) with the carriage return, line feed and binary 0 character terminated. Alternatively, a .PBWRITE D command will send data in binary format.



Field and Bit Descriptions		
next_ORB	64 bits	ORB pointer format shall be in accordance with SBP-2 specifications.
data_descriptor	32 bits	Data_descriptor shall contain the address of the data buffer.
n	1 bit	Completion notification “n” shall be one. The target shall store a status block at the status_FIFO address at the address supplied in the login request.
Rq_fmt	2 bits	Required format shall be zero.
r	1 bit	Reserved shall be zero (0);
d	1 bit	Direction bit “d” shall be zero.
spd	3 bits	Speed “spd” shall have a value of two.
max_payload	4 bits	The maximum data transfer length “max_payload” shall be nine.
p	1 bit	The removable media device must be prepared to handle the Page table bit p=0 and p=1 cases as the standard operating systems set this bit without influence of the application programs.
page_size	3 bits	Page size shall be zero if the p field is set to 0, otherwise this field shall be set to the valid page size.
data size	16 bits	The data size field should be set according to the allocated send buffer size in bytes (N). The length must be at least 80 (50h).
LUN	3 bits	The logical unit number shall be one (1).
Res	4 bits	Reserved shall be zero (0).
AEN	1 bit	AEN shall be zero.
Xfer Lng	24 bits	The length must be at least 80 (50h).
Control	8 bits	Control shall be 80h.

Figure 10-118. Send command ORB format.

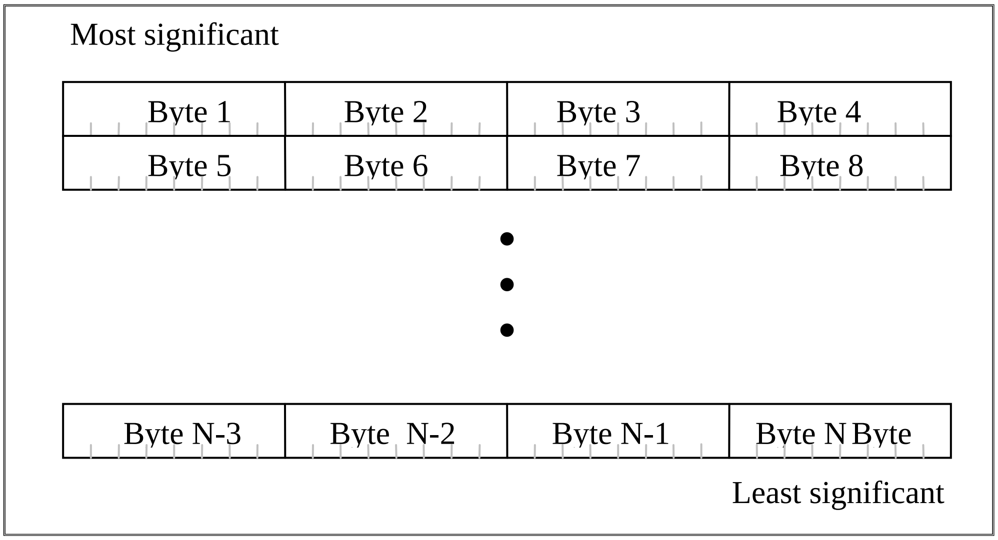


Figure 10-119. Send data buffer format.

d. Receive. The Receive command block ORB format is illustrated in Figure 10-120; the Receive data buffer can be returned in ASCII format (see Figure 10-121) or in binary format (see Figure 10-122), if the retrieved data contains binary information. Multiple ORBs may be used to retrieve the data required.

- The returned remote answer is an ASCII text terminated by the '*' (asterisk) character IAW IRIG-106 Chapter 6-8. If the '*' terminator is missing, multiple Receive commands must be used to retrieve the data until the '*' terminator is received.

- If the returned remote answer can contain mixed ASCII text or binary information until the specified length in the first 32-bit word. The first byte is a hexadecimal 10 code to identify the binary format (codes hexadecimal 11-1F are reserved for future extensions). The answer must be terminated by the '*' (asterisk) character IAW IRIG-106 Chapter 6-8. If the '*' terminator is missing, multiple Receive commands must be used to retrieve the data until the '*' terminator is received.

most significant								
next_ORB								
data_descriptor								
n	Rq_fmt	r	d	spd	max_payload	p	page_size	data size
08h					LUN	Res	AEN	Allocation Lng – upper bits
Allo Lng – low					Control			00h / 00h
00h					00h			00h / 00h
least significant								

Field and Bit Descriptions

next ORB	6 bits	ORB pointer format shall be in accordance with SBP-2 specifications.
data Descriptor	64 bits	Data_descriptor shall contain the address of the data buffer.
n	1 bit	Completion notification “n” shall be one. The target shall store a status block at the status_FIFO address at the address supplied in the login request.
Rq_fmt	2 bits	Rq_fmt shall be zero.
r	1 bit	Reserved: shall be zero (0).
d	1 bit	Direction bit “d” shall be one.
spd	3 bits	Speed “spd” shall have a value of two.
max_payload	4 bits	The maximum data transfer length “max_payload” shall be nine.
p	1 bit	The removable media device must be prepared to handle the Page table bit p=0 and p=1 cases as the standard operating systems set this bit without influence of the application programs.
page_size	3 bits	Page size shall be zero if the p field is set to 0, otherwise this field shall be set to the valid page size.
data size	16 bits	The data size field should be set according to the allocated receive buffer size in bytes (N). The length must be at least 80 (50h).
LUN	3 bits	The logical unit number shall be one (1).
Res	4 bits	Reserved shall be zero (0).
AEN	1 bit	AEN shall be zero.
Allocation Lng	16 bits	Allocation_Lng = length of the Chapter 6 response string.
Control	8 bits	Control shall be 80h.

Figure 10-120. Receive command block ORB format (8 quadlets).

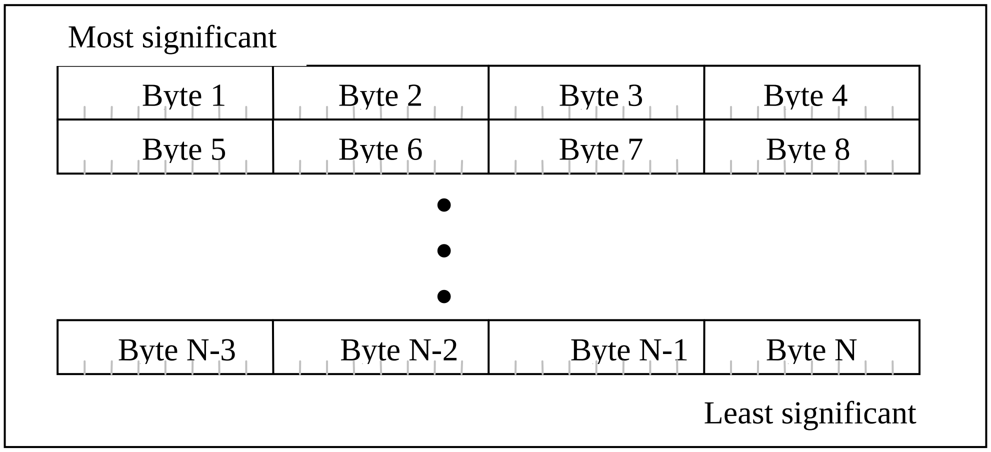


Figure 10-121. Receive data buffer format ASCII format.

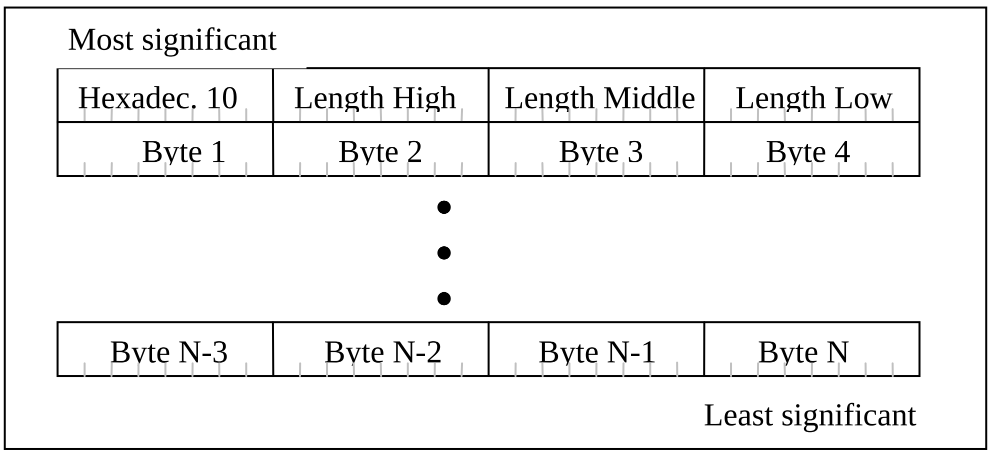


Figure 10-122. Receive data buffer binary format.

10.10 Ground Based Recorders

This section of the standard specifies the basic requirements of Ground Based Recorders hereinafter referred to as Ground Recorders. The main functional requirements of Ground Recorders are:

a. Recorder Interface.
b. Recorder Data Format.
c. Recorder Media.
d. Recorder Command and Control (if the Ground Recorder is to be controlled remotely).

Optionally, Ground Recorders may support replay, reproduction, and display of IRIG-106 Chapter 10 data recordings. Basic replay and reproduction interoperability requirements will be defined in this section. Data display requirements are outside the scope of this standard and will not be defined.

10.10.1 Interface.

a. At a minimum, the required Ground Recorders interface shall be Ethernet for remote command and control IAW paragraph 10.4 and 10.7 of this standard.

b. Optionally, ground recorders can implement additional interfaces for remote command and control, remote data access, and/or data streaming. If a ground recorder uses iSCSI or contains an RS-232/422, IEEE-1394 and/or Fibre Channel for these interfaces, it shall be IAW paragraph 10.4 and 10.7 of this standard.

c. Data streaming.

- The recorder can optionally have the capability to stream IRIG-106 Chapter 10 format data (paragraph 10.10.2) out of its required Ethernet Interface IAW paragraph 10.3.9.1.
- Stream commit time as defined in paragraph 10.6.1.h of this standard shall apply to Ethernet Interface Data streaming.

10.10.2 Data Format. Ground Recorders shall format, multiplex, and record all data IAW paragraph 10.6 of this standard.

10.10.3 Recording Media. Ground Recorders shall record data IAW paragraph 10.10.2 to the following Recording Media:

a. COTS Media. Commercial Off-The-Shelf (COTS) is defined as any recording media (such as hard disks, solid state drives, tape, RAID, and JBOD) that is ready-made and available for sale to the general public.
COTS Media shall have a ready-made and available for sale to the general public electrical interface (such as PATA, SATA, IEEE-1394, USB, SCSI, Ethernet) to the Ground Recorders.

NOTE	If Ground Recorders use COTS Media for recording of the paragraph 10.10.2 data format, the recorded data Remote Data Access at a minimum shall be across the required Ground Recorder Ethernet interface using iSCSI IAW paragraph 10.4.3 (Ethernet Recorder Interface) and paragraph 10.5 (Interface File Structure) of this standard.

NOTE	If Ground Recorders provide Remote Data Access across the Ground Recorder Ethernet interface, the paragraph 10.5 (Interface File Structure) at a minimum shall be presented at the interface. This does not dictate which COTS media format or data organization is implemented, but does require that the paragraph 10.5 (Interface File Structure) is presented at the recorder Ethernet interface.

b. COTS Media Requirements. COTS Media used by Ground Recorders shall provide the capability of recording valid IRIG-106 Chapter 10 Original Recording File(s) IAW paragraph 10.11 (Data Interoperability). All paragraph 10.11 Data Transfer and File Management requirements of this standard shall apply to Ground Recorders.

10.10.4 Remote Command and Control.

a. Optionally, if a Ground Recorder is controlled remotely, it shall provide command and control IAW paragraph 10.7.8 and 10.7.9 of this standard across the Ethernet Interface port as defined in paragraph 10.10.1 of this standard.

b. Ground Recorders at a minimum are required to use iSCSI or TELNET as the command and control Ethernet transport mechanism as defined in paragraph 10.4 and paragraph 10.7.

c. Ground Recorders providing remote command and control capability shall provide the functionality for all commands defined in paragraph 10.7.9 of this standard.

d. Optionally, if a Ground Recorder contains a RS-232/422, IEEE-1394B and/or Fibre Channel interface as defined in paragraph 10.10.1 the recorder will provide command and control IAW paragraph 10.7 and IRIG-106 Chapter 6.

10.10.5 Data Replay and Reproduction.

10.10.5.1 Channel Mapping.

a. Optionally, if a Ground Recorder provides data playback capability, it shall provide for the logical assignment of recorded channels to physical channels on the Ground Recorders.

b. Playback will not require movement of cards between slots to make assignments for playback.

10.10.5.2 Recording/Reproduction Data Rates. Optionally, if a Ground Recorder provides a data playback capability, it shall provide information using the IRIG-106 Chapter 6 .HEALTH and .CRITICAL commands if the bandwidth of data to be played back exceeds the aggregate bandwidth of the Ground Recorder.

10.10.5.3 Network Recording Playback.

a. Optionally, if a Ground Recorder provides a data playback capability, it shall provide replay from COTS Media (paragraph 10.10.3) to the Ethernet Interface. The Ethernet format of the network recording playback will be IAW paragraph 10.3.9.1.

b. If the network recording playback capability is commanded remotely, Ground Recorders shall support the functionality of the IRIG-106 Chapter 6.

10.11 Data Interoperability

10.11.1 Original Recording Files. All files contained within a recorder, RMM, COTS Media, or that are a byte-for-byte single file downloaded to a host computing platform in unaltered form shall be considered Original Recording Files and be in full compliance with the Data Organization in paragraph 10.5.1 and Data Format in paragraph 10.6.

- Original Recording File Annotation. In order to provide a standardized method of annotation for Original Recording Files, the following procedures shall be used to ensure IRIG-106 Chapter 10 compliancy:
  - The Computer Generated Data, Format 1 Setup Record shall always contain the required attributes IAW paragraph 10.11.
  - The Original Recording File Setup Record R-x\RI3 "Original Tape/Storage" attribute value shall be **R-x\RI3:Y;**
  -

10.11.2 Modified Recording Files. Modified Recording files are created from Original Recording Files directly from a recorder, RMM, COTS Media or from Original Recording Files that have been downloaded to a host computing platform. There are several instances of Modified Recording Files - filtered or sanitized data, a subset of channels, a superset of channels, a subset of time, a subset of both channels and time, or a superset of channels and subset of time.

10.11.2.1 Modified Recording File Annotation. In order to provide a standardized method of annotation for Modified Recording Files, the following procedures shall be used to ensure IRIG-106 Chapter 10 compliancy.

a. The Computer Generated Data, Format 1 Setup Record shall always contain the required attributes IAW paragraph 10.11.

b. Any time a modification is made to an original recording the R-x\RI3 "Original Tape/Storage" attribute value shall be changed:

   From: R-x\RI3:Y;
   To: R-x\RI3:N;

   Also, the R-x\RI6 "Date of Modification" attribute will be added if not already present, in which case if R-x\RI3 contains a "Y" R-x\RI6 shall be empty. The R-x\RI8 attribute value shall contain the last date and time the Modified Recording File was created.

c. If the Modified Recording File is not a Time Subset but either a Channel Subset or both a Time and Channel Subset, then the step "b" attributes shall be changed as defined. Also the original channels which are not included in the Recording Subset File shall have the R-x\CHE-n "Channel Enable" attribute changed:

   From: R-x\CHE-n:T;
   To: R-x\CHE-n:F;

   A comment attribute R-x\COM will be inserted directly after the changed R-x\CHE-n attribute and shall contain the following:

   "original recording change - removed channel-n" (where n represents the Channel ID of the channel that was removed).

d. If the Modified Recording File is not a Time Subset but either a Channel Superset or both a Time Subset and Channel Superset, then the step "b" attributes shall be changed as defined. Also the channels added in the Modified Recording File shall contain the required attribute IAW paragraph 10.11:

A comment attribute R-x\COM will be inserted directly after the added channel R-x\CHE-n attribute and shall contain the following:

"original recording change - additional channel-n" (where n represents the Channel ID of the channel that was added).

If the Modified Recording File contains filtered (removed packets or data) or sanitized data (overwrite of data), then the step "b" attributes shall be changed as defined. Also the channels which contain filtered or sanitized data in the Modified Recording File shall also contain a comment attribute R-x\COM inserted directly after the channel R-x\CHE-n attribute and shall contain the following:

"original recording change - filtered channel-n" (where n represents the Channel ID of the channel that was filtered).

10.11.2.2 Modified Recording File Restructuring. When a Modified Recording File is created there will be alterations to original packets or possibly structure. Therefore:

a. All files shall reflect any sequence number, packet length, or checksum changes in the appropriate Packet Header fields.
b. If enabled in the original recording Computer Generated Data, Format 3 Recording Index packets shall be recalculated to ensure correct information is contained within the entries as they relate to the newly created Modified Recording File.

10.11.3 Original Recording and Modified Recording File Extension. Upon data download to a host computing platform, all Original Recording Files and/or Modified Recording Files shall use the file extension ***.ch10 (or *.c10** extension for use on systems with a 3 character extension limit). The use of this standard extension will indicate that any Original Recording File and/or Modified Recording File on a ground computing or storage platform shall be in compliance with this paragraph 10.11 of this standard.

10.11.4 File Naming. Upon data download from the recorder or RMM to a host computing platform, all Original Recording Files or Modified Recording Files shall use the following structure and naming conventions unless host computing platform operating system imposes naming length limits. In this case the directory and file names are to be truncated after the last component that completely fits within the name length limit:

10.11.4.1 <u>On-Board Recorder</u>.

a. Data Recording Directory Name. Each directory block from a RMM to be downloaded to a ground computing or storage platform shall use paragraph 10.5, Table 10-2 VolName as the directory name where the Data Files will be placed. The directory name shall use lower case alpha characters. If the VolName is empty (0x00), a default name or user-defined name shall be used. If used the default name shall be ch10dirnnn, where nnn is the sequential directory block count.

b. Data File Name. Each Data File contained within a Directory Block on the RMM to be downloaded will be placed in the Directory identified in paragraph 10.11.4.1.a and shall use the following naming convention. The data file name shall use lower case alpha characters:

“filennnn”; where nnnn is the sequential RMM file count from each Directory Block File Entry (must be 8 alpha-numeric characters).

<u>Example</u>: “file0001,” “file0002,” …:“file9999.”

If available, “File Create Date,” “File Create Time” and “File Close Time” from paragraph 10.5, Table 10-3, DDMMYYYY_HHMMSSss_HHMMSSss (8 numeric characters for File Create Date, 8 numeric characters for File Create Time separated by an underscore ASCII character code 0x5F, and 8 numeric characters for File Close Time). No spaces or other non-numeric characters allowed). Example 02092004_21302731_21451505.

If the “File Create Date,” “File Create Time” and “File Close Time” from paragraph 10.5, Table 10-3 values are not available and are filled with 0x2D, then the system time from the host download platform will be used for “File Create Date” and “File Create Time” (DDMMYYYY_HHMMSS). “File Close Time” will not be used. “File Close Time” shall be replaced with “sys_time.”

A structure example follows:

…\VolName\FileName_FileCreateDate_FileCreateTime_FileCloseTime
When VolName not empty example:
…\<VolName>\file0001_02092004_21302731_21451505.ch10
When VolName empty default example:
…\ch10dir001\file0001_02092004_21302731_21451505.ch10
When VolName empty user defined example:
…\<User Defined>\file0001_02092004_21302731_21451505.ch10
When Date/Time not available (0x2D fill) example:
…\file0001_02092004_213027_sys_time.ch10

The use of this standard recording and file naming convention will indicate that any file on a ground computing or storage platform is in compliance with this standard.

10.11.4.2 Ground Based Recorder.

a. Recording Directory Name. Each directory where the Data Files will be placed shall use the naming convention \ch10dir_DDMMYYYY_nnn; where n is the sequential number of Chapter 10 recording directories created on the DDMMYYYY date. The directory name shall use lower case alpha characters.

b. Recording File Name. Each Data File contained within a Directory shall use the following naming convention. The data file name shall use lower case alpha characters:

“filennnn”; where nnnn is the sequential file count from each recording (must be 8 alpha-numeric characters). Example “file0001,” “file0002,” …:“file9999.”

File Create Date, File Create Time and File Close Time shall use the following naming convention DDMMYYYY_HHMMSSss_HHMMSSss (8 numeric characters for File Create Date, 8 numeric characters for File Create Time separated by an underscore ASCII character code 0x5F, and 8 numeric characters for File Close Time). No spaces or other non-numeric characters allowed). Example 02092004_21302731_21451505.

A structure example follows:

…\ch10dir_02092005_001\file0001_02092005_21302731_21451505.ch10

The use of this standard recording and file naming convention will indicate that any file on a ground computing or storage platform is in compliance with this standard.

10.11.5 Data Transfer File. In order to ensure the highest degree of interoperability for transfer of IRIG-106 Chapter 10 Recorder or RMM contents, Original Recording Files or Modified Recording Files between organizations the Data Transfer File Structure shall be used. Essentially, a Data Transfer File contains all the same information and data that a recorder or RMM would present at the interface albeit within a single binary structure on either tape or random access devices. The Data Transfer File could also contain Original Recording Files or Modified Recording Files from multiple recordings or dates.

NOTE	Original Recording Files or Modified Recording Files downloaded to a host computing platform and transferred as a single file shall follow 10.11.1 and 10.11.2.

10.11.5.1 Data Transfer File Structure Definition. The following describes Data Transfer File Structure and media environments:

a. Tape Devices. A Data Transfer File on tape devices is treated essentially the same as a recorder or RMM in that the directory structure and data contents are as defined and organized in this standard. The Data Transfer File is a single binary file containing a directory structure IAW 10.5 and a single or multiple IRIG-106 Chapter 10 Original Recording Files or Modified Recording Files. Only one Data Transfer File will be contained on a tape device media. The tape block size shall be 32K (32,768) bytes.

- Logical Address 1 will contain a directory and file structure IAW paragraph 10.5.2.
- The corresponding IRIG-106 Chapter 10 Original Recording Files or Modified Recording Files will follow the directory structure in contiguous bytes until the end of the Data Transfer File. The beginning of each IRIG-106 Chapter 10 Original Recording File or Modified Recording File in the Data Transfer File will begin at the byte offset contained in each File Entry table File Start Address value.

b. Random Access Devices. A Data Transfer File on a random access device is treated essentially the same as a recorder RMM in that the directory structure and data contents are as defined and organized in this standard. The Data Transfer File is a single binary file containing a directory structure IAW 10.5.2 and a single or multiple IRIG-106 Chapter 10 Original Recording Files or Modified Recording Files. Multiple Data Transfer Files can be contained on a random access device.

- The paragraph 10.5.2 directory structure within the Data Transfer File begins at byte 0 and runs contiguously until the last file entry paragraph. The next byte after the last file entry block shall be the first byte in the first data file.

- The corresponding IRIG-106 Chapter 10 Original Recording Files or Modified Recording Files will follow the directory structure in contiguous bytes until the end of the Data Transfer File. The beginning of each IRIG-106 Chapter 10 Original Recording File or Modified Recording File in the Data Transfer File will begin at the byte offset contained in each File Entry table File Start Address value.

-

10.11.5.2 Data Transfer File Extension. Upon creation, all IRIG 106 Chapter 10 compliant Data Transfer Files not on Tape Devices shall use the file extension ***.tf10 (or *.t10** extension for use on systems with a 3 character extension limit). The use of this standard extension will indicate that any Data Transfer File on a host computing or storage platform shall be in compliance with paragraph 10.11.5 of this standard.

10.11.6 Recording Directory File. A binary recording directory file which is a byte-for-byte copy of the RMM or Recorder directory structure presented at the interface. This file should represent the contents of a RMM or recorder directory at the time of IRIG-106 Chapter 10 data download. The bytes in this file contain the byte-for-byte contents of the RMM's directory blocks in the order the directory blocks are linked, using each block's "forward Link" field.

10.11.6.1 Recording Directory File Extension. Upon creation, all IRIG 106 Chapter 10 compliant Recording Directory Files shall use the file extension ***.df10 (or *.d10** extension for use on systems with a 3 character extension limit). The use of this standard extension will indicate that any Recording Directory file on a host computing or storage platform shall be in compliance with paragraph 10.11.6 of this standard.