CHAPTER 10

Digital Recording Standard

Changes to This Edition of Chapter 10

Numerous changes 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.

Paragraph	Description
10.6.17	CR21- Added CAN Bus Data Type.
Table 10-10, 10.6.15.2	CR22-Added AFDX Data Type
10.6.2.2, Table 10-13, Table 10-15, Table 10-17	CR49-Clarification of 32-bit alignment
Table 10-10, 10.6.11.3	CR57 - Added Dynamic Image Packets Sub-Type
10.9	CR58-Added Ethernet download port capability to RMM
10.6.1.1.g, 10.6.1.2.a, Figure 10-14	CR59-Added Real-Time extension for RTC to support other than 10 MHz.
10.6.7.3.d, 10.6.7.3.e, 10.6.7.3.f	CR60-Added support for Mode 7 Events
10.6.15.1.b, Figure 10-81	CR61-Added CRC error detection in Ethernet Packets
10.3.9.2	CR64-Added support for TCP/IP

List of Figures

List of Tables

This page intentionally left blank.

Acronyms

μs	microsecond
ACCTS	Air Combat Test and Training System
ARINC	Aeronautical Radio, Incorporated
BC	bus controller
BCD	binary-coded decimal
BCS	basic character set
BIT	built-in test
CAN	controller area network
CBR	constant bit rate
CCM	command and control mnemonics
CDB	command descriptor block
CIU	communication interface unit
COTS	Commercial Off-the-Shelf
CSDW	channel-specific data word
CTS	Combat Training System
DC	direct current
DCRsi	Digital Cartridge Recording System
DHCP	Dynamic Host Control Protocol
EUI	enterprise-unique identifier
ERTC	extended relative time counter
FC-PLDA	Fibre Channel Private Loop SCSI Direct Attach
FTP	file transfer protocol
Gbps	gigabit per second
GHz	gigahertz
GPS	Global Positioning System
I/O	input/output
IAW	in accordance with
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
IPDH	intra-packet data header
IPH	intra-packet header
IPMH	intra-packet message header
IPTS	intra-packet time stamp
IPv4	Internet Protocol version 4
IQN	iSCSI qualified name
IRIG	Inter-Range Instrumentation Group
iSCSI	Internet Small Computer Systems Interface
ISO	International Organization for Standards
IT	Index Type
ITU-T	International Telecommunications Union/Telecommunication Standardization Sector
KB	kilobyte

KITS	Kadena Interim Training System
KLV	Key-Length-Value
LSB	least significant bit
LSLW	least significant long word
LUN	logical unit number
MAC	media access control
Mbps	megabit per second
MHz	megahertz
MIL-STD	Military Standard
MISP	Motion Imagery Standards Profile
mm	millimeter
MPEG	Moving Picture Experts Group
ms	millisecond
MSB	most significant bit
MSLW	most significant long word
MTU	maximum transmission unit
NADSI	NATO Advanced Data Storage Interface
NATO	North Atlantic Treaty Organization
ORB	operation request block
PAT	program association table
PCM	pulse code modulation
PCR	program clock reference
PES	program elementary stream
PID	program ID
PMT	program map table
PoE	power over Ethernet
ppm	parts per million
PS	program stream
PTP	precision time protocol
RCC	Range Commanders Council
RFC	Request For Comment
RIU	remote interface unit
RMM	removable memory module
RS	Recommended Standard
RSCF	recorder setup configuration file
RT	remote terminal
RTC	relative time counter
SBP	Serial Bus Protocol
SCSI	Small Computer Systems Interface
SLP	service location protocol
SSD	solid-state disk
STANAG	Standardization Agreement
TCP	Transmission Control Protocol
TMATS	Telemetry Attributes Transfer Standard
TS	transport stream
TSPI	Time Space Position Information

UART	Universal Asynchronous Receiver and Transmitter
UCS	Universal Multiple-Octet Coded Character Set
UDP	User Datagram Protocol
UO	unexpected one
UTC	Universal Coordinated Time
UZ	unexpected zero
VDC	volts direct current

This page intentionally left blank.

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 that 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 predominant 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 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

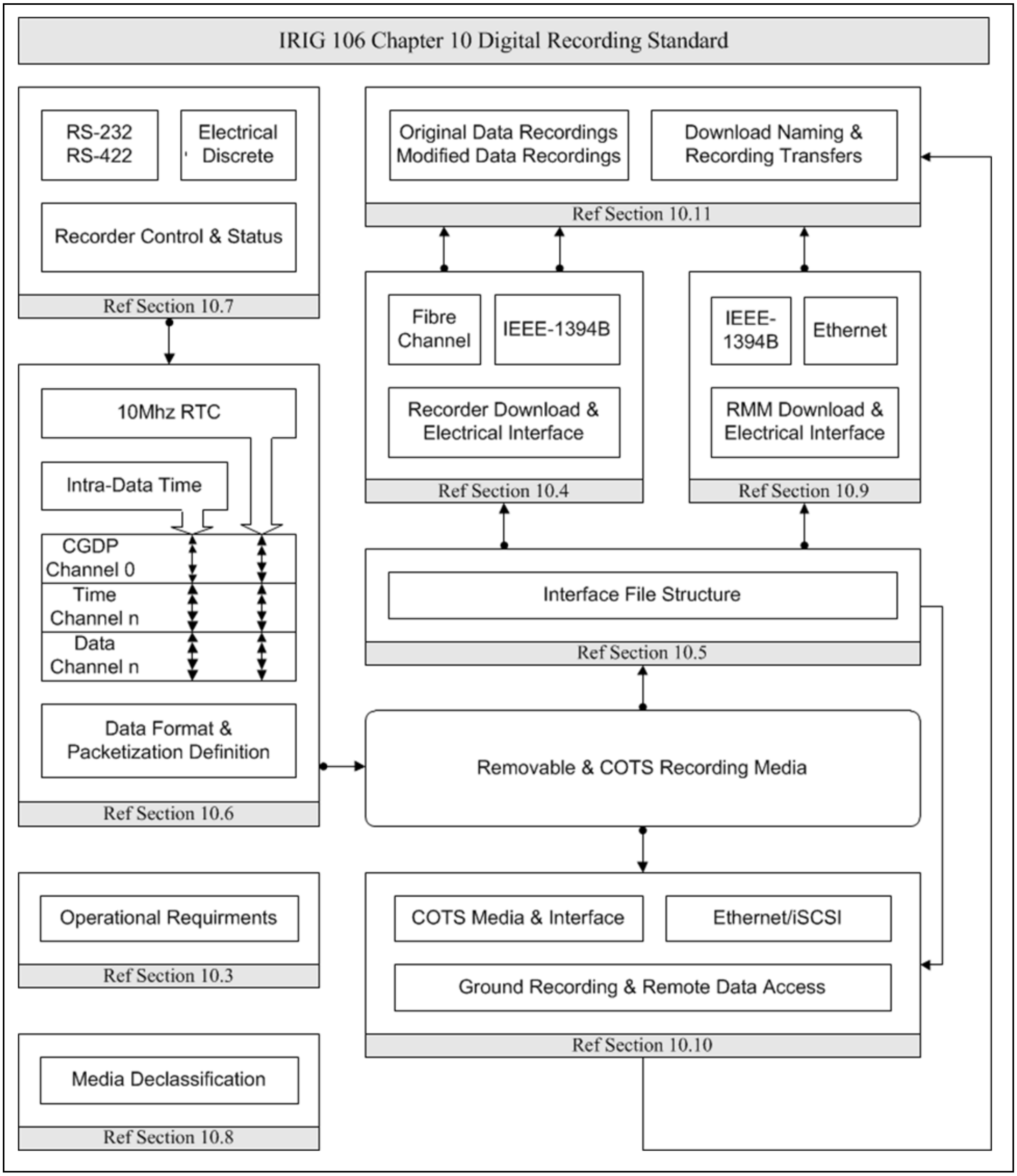


Figure 10-1. Functional Layout of Digital Recorder Standard

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 Military Standard (MIL-STD) 1553 data bus, time, analog, video, Aeronautical Radio, Inc.(ARINC) 429, discrete, and Universal Asynchronous Receiver and Transmitter (UART) containing Recommended Standard (RS)-232/422/485 communication data. This digital recording standard will allow use of a common set of playback/data reduction hardware/software to take advantage of emerging random access recording media.

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

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. In addition, declassification requirements are discussed in Section 10.8, ground-based recording is discussed in Section 10.10, and data interoperability requirements are discussed in Section 10.11.

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

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

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

d. Recorder Control and Status, which defines command and control mnemonics (CCM), status, and their interfaces, is described in Section 10.7.

e. Host Platform Interface to Recorder Removable Media is defined in Section 10.9.

f. Ground-Based Recorder Interface, which defines unique interoperability requirements of a ground-based recorder, is described in Section 10.10.

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

10.2 Definitions

As of RCC 106-13 published June 2013, the definitions that in previous versions comprised this section are now located in Appendix 10-A, one of two appendixes new to this publication.

10.3 Operational Requirements

On-board recorders are the basis and original justification for this standard. This section defines the requirements for on-board recorders to be in 100 percent compliance.

10.3.1 Recorder Compliancy Requirements.

Table 10-1 and Table 10-2 represent the mandatory recorder requirements to meet 100 percent 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 (IAW) the definitions in this standard in order to meet 100 percent compliancy of this standard.

Table 10-1. On-Board Recorder Mandatory Compliancy Requirements

Applicable Compliancy Section	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
Removable Memory Module (RMM) Electrical Interface and Power	
10.3, 10.9	IEEE 1394b Bilingual Socket or Ethernet 8P8c/RJ45
Commercial Off-the-Shelf (COTS) Media Electrical Interfaces	
10.3	COTS Media Interface
RMM Interface Protocols	
10.9	IEEE 1394b SCSI/SBP-2 or IEEE 802.3 IPv4
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 Interoperability	
10.11	Original Recording Files

Table 10-2. Ground-Based Recorder Mandatory Compliancy Requirements

Applicable Compliancy Section	Function/Capability
Recorder Electrical Interfaces	
10.10	Ethernet
Recorder Remote Interface Protocols	
10.10, 10.4	Internet Small Computer Systems Interface (iSCSI) and/or Telnet
COTS Media Electrical Interfaces	
10.10	COTS Media Interface
COTS Media Interface Protocols	

Table 10-2. Ground-Based Recorder Mandatory Compliancy Requirements

Applicable Compliancy Section	Function/Capability
10.10	COTS Media Interface
Remote Data Access 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 Interoperability	
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 Section 10.4. This combination will allow data extraction and transfer from any recorder to any Section 10.4-compliant intermediate storage unit. The required control port interface shall be IAW Section 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, 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 Subsection 10.4.1 for recorder download and electrical interface, Section 10.9 for RMM interface, and Section 10.11 for data transfer and file management.

10.3.6 Host Platform Interface to Recorder Media.

Interface to on-board recorder media shall be accomplished utilizing IEEE 1394b or Ethernet interfaces. Interface connectors IAW Section 10.9.5 shall be provided on the media to allow direct download of data to the host computer or storage device.

10.3.7 Required File Tables Entries.

Within Section 10.5, Table 10-5 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 10-5 has been adopted from Standardization Agreement (STANAG) 4575[1] but in the case of Chapter 10 unless 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 10-4 Shutdown value is 0xFF or 0x00 and 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 Setup Configuration File.

A recorder setup configuration file (RSCF) can reside on the recorder or optionally reside in the RMM. Recorder setup configuration must be IAW Chapter 9. Recorder setup configurations shall be programmed IAW Section 10.7 of this standard. Optionally the recorder can be configured from a Chapter 10 configuration file residing in the RMM. The RMM RSCF 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 Subsection 10.5.2.

a. All directory block format fields shall be IAW Table 10-4. The field n File Entries value shall be 1.

b. All directory entry format fields shall be IAW Table 10-5. 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.

[1] North Atlantic Treaty Organization. “NATO Advanced Data Storage Interface (NADSI).” STANAG 4575 (Edition 3). 8 May 2009. May be superseded by update. Available at http://www.nato.int/structur/AC/224/standard/4575/ag4_4575_E_ed3_nu.pdf.

10.3.8.2 Recorder Configuration File Structure.

The RSCF 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 Subsection 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 Subsection 10.5.2.

a. All directory block format fields shall be IAW Table 10-4. The field "n File Entries" value shall be 1.

b. All directory entry format fields shall be IAW Table 10-5. 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 RSCF shall NOT be able to be erased by the recorder .ERASE or DISCRETE command.

10.3.9 Recorder Data Streaming Transport.

Data streaming transport may be accomplished across the Section 10.4 recorder download and electrical interfaces using the definitions in Section 10.2 and commands in 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 (Section 10.10) IAW Section 10.4. This Ethernet interface can optionally be used for data streaming using User Datagram Protocol (UDP) or optionally Transmission Control Protocol/Internet Protocol (TCP/IP). This will be accomplished with the Chapter 6 PUBLISH command.

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 Chapter 10 data.

10.3.9.1.2 UDP Transfer Header.

Network broadcasting limitations of Section 10.6 packets (up to 128 megabytes or 512 kilobytes [KB]) 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 structure shown at Figure 10-2 shall be used for UDP transfer headers in UDP packets containing one or more full Chapter 10 data packets.

Most Significant Word			Least Significant Word
Word 1		Word 0	
Most Significant Bit (MSB)			Least Significant Bit (LSB)
31 8		7 4	3 0
UDP Message Sequence Number		Type of message	Version

Figure 10-2. User Datagram Protocol Transfer Header for Non-Segmented Data

The structure at Figure 10-3 shall be used for UDP transfer headers in UDP packets containing a segmented Chapter 10 data packet.

Most Significant Word			Least Significant Word
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. User Datagram Protocol 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 Chapter 10 packet.

- Channel ID (16 bits). Segmented packets only, channel ID of the data in the Chapter 10 packet.

- Channel Sequence Number (8 bits). Segmented packets only, channel sequence number of the data in the Chapter 10 packet.
- Reserved (8 bits). Reserved.
- Segment Offset (32 bits). Segmented packets only, position of the data in the Chapter 10 packet.

10.3.9.1.3 UDP 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 the recording segment of a packet and shall be ordered (segment offset incrementing). Figure 10-4 and Figure 10-5 present the sequence of the general UDP network broadcast of full or segmented packets.

UDP Header
UDP Transfer Header
Chapter 10 Packet 1
:
Chapter 10 Packet N

Figure 10-4. General User Datagram Protocol Network Broadcast (Full Packet)

UDP Header
UDP Transfer Header
Chapter 10 Packet Segment

Figure 10-5. General User Datagram Protocol Network Broadcast (Segmented Packet)

- When using Internet Protocol version 4 (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 Chapter 10 packets, so a jumbogram shall always contain an integer number of Chapter 10 packets.
- Stream Commit time applies to recorders using Ethernet data streaming transport.

10.3.9.2 Transmission Control Protocol Data Transfer.

NEW

The recorder may optionally also offer access to the acquired data on a dedicated TCP/IP connection on port # 10620.

The data on this port will include only the live acquisition data. Especially Computer-Generated Data Packet, Format 1 setup record will typically not be included to conserve bandwidth. Configuration information can be obtained using other protocols (i.e., .TMATS READ command via Telnet, see Subsection 10.4.3).

The data availability can be controlled with the remote control command: .PUBLISH_TCP (see Chapter 6).

If the bandwidth of the TCP channel is not sufficient to convey all acquisition data, the recorder will have priority for recording on the media and may drop some complete as well as partial packets on the TCP port.

10.3.10 Commercial Off-the-Shelf Media.

In conjunction with an on-board recorder and/or a multiplexer when an 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 Section 10.4 and optionally by at least one COTS media interface. When accessing COTS media the interface file structure definition defined in Section 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 Subsection 10.3.2) shall be Fibre Channel or IEEE 1394b and optionally Ethernet (Subsection 10.4.3). The physical, signaling, and command protocols contained in subsections 10.4.1 and 10.4.2 are a subset of, and adapted from STANAG 4575.

10.4.1 Fibre Channel Recorder Download Interface.

10.4.1.1 Physical and Signaling.

The interface shall comply with Fibre Channel-Physical Interfaces and Fibre Channel-Framing and Signaling in Section 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 gigabaud.

10.4.1.2 Command Protocol.

The interface shall conform to the requirements of the Fibre Channel Private Loop SCSI Direct Attach (FC-PLDA) (American National Standards Institute/International Committee for Information Technology Standards TR19-1998)[2] 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. 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.

[2] International Committee for Information Technology Standards. “Fibre Channel - Private Loop SCSI Direct Attach (FC-PLDA).” INCITS TR-19-1998. January 1998. Available for purchase at http://www.techstreet.com/incits/searches/385689. Replaced by “INCITS Technical Report - for Information Technology - Fibre Channel - Device Attach (FC-DA).” INCITS TR-36-2004. February 2005. Available for purchase at http://www.techstreet.com/incits/searches/385707.

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." The 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 RMM as well. Commands required to format and/or write to an RMM are defined as "Recommended." These commands are not required for any 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 10-3 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.

Table 10-3. Required and Recommended SCSI Commands, Features, and Parameters

Feature (Command)	Initiator	Target*	Notes
Inquiry	I	R	
Standard INQUIRY data (bytes 0-35)	I	R	
Enable Vital Product Data = 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 (Basic Character Set [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	

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.

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 NADSI interface, the RMM shall automatically initialize into a mode where the NADSI port is active and is the priority data and control interface.

10.4.2 IEEE 1394b Recorder Interface.

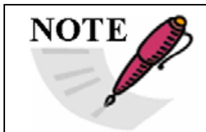
The IEEE 1394b recorder download interface shall use the same mechanisms as Section 10.9 where applicable.

10.4.2.1 Physical and Signaling.

The interface shall allow control of vendor-specific recorder devices. The command protocol shall be IAW Subsection 10.4.1.2 and Table 10-3.

10.4.2.2 Recorder Communication.

The fundamental method of communicating shall be IAW the IEEE 1394b protocol. Packets sent and received shall be asynchronous transmissions. The 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 (ORBs).



The SBP-2 provides for the transport of 6-, 10-, and 12-byte SCSI command descriptor blocks (CDBs) 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) Request for Comment (RFC) 854,[3] RFC 855,[4] and RFC 1184.[5] The protocol will support Chapter 6 CCM (reference Paragraph 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 Logical Unit Number Assignments.

The following iSCSI target logical unit number (LUN) assignments shall be used.

a. LUN 0 or 32 shall be used for recorder data download via Section 10.5 interface.

b. LUN 1 or 33 shall be used for recorder Chapter 6 CCM (Reference Section 10.7).

[3] Internet Engineering Task Force. "Telnet Protocol Specification." RFC 854. May 1983. May be superseded or amended by update. Available at http://tools.ietf.org/html/rfc854.

[4] Internet Engineering Task Force. "Telnet Option Specifications." RFC 855. May 1983. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc855/.

[5] Internet Engineering Task Force. "Telnet Linemode Option." D. Borman, ed. RFC 1184. October 1990. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc1184/.

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 Subsection 10.4.1.2 and Table 10-3.

10.4.3.4 Recorder Communication.

The fundamental method of communicating shall be IAW 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. 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.

NOTE	This section duplicates text from STANAG 4575. Any definition in this standard that varies from the STANAG 4575 text is noted in a NOTE box. The text in a NOTE box takes precedence over the text from STANAG 4575.

NOTE	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 Section 10.5 is maintained or seen at the interface (Section 10.4).

10.5.1 Data Organization.

A data recording can contain a single file, which is composed of one or more types of packetized data, or multiple files, in which one 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), Format 1 setup record IAW Subsection 10.6.7.2 as the first packets in the recording

b. Time data packet(s) IAW Subsection 10.6.3 as the first dynamic packet after the computer-generated packet, setup record

c. One or more data format packets IAW Subsection 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). See Figure 10-6.

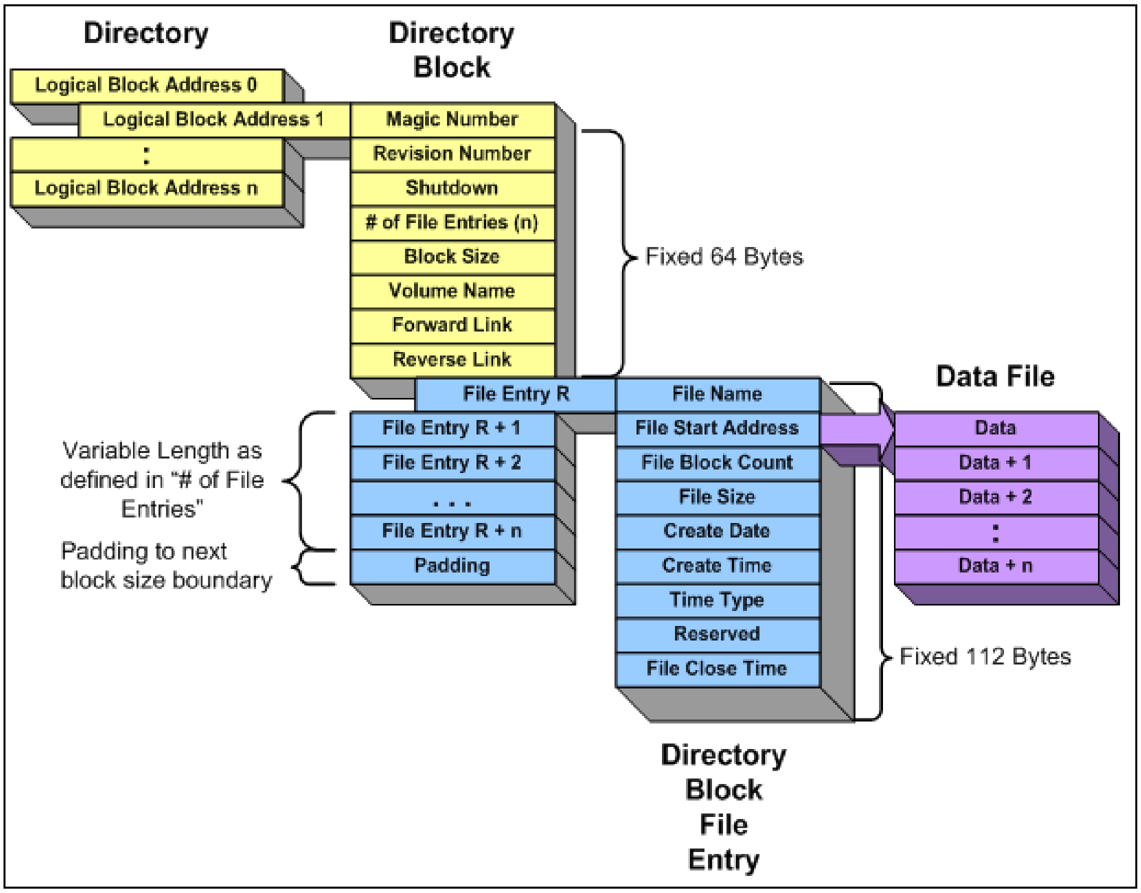


Figure 10-6. Directory Structure

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.

10.5.2 Directory Definition.

The name and location information for all files recorded in a directory 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.

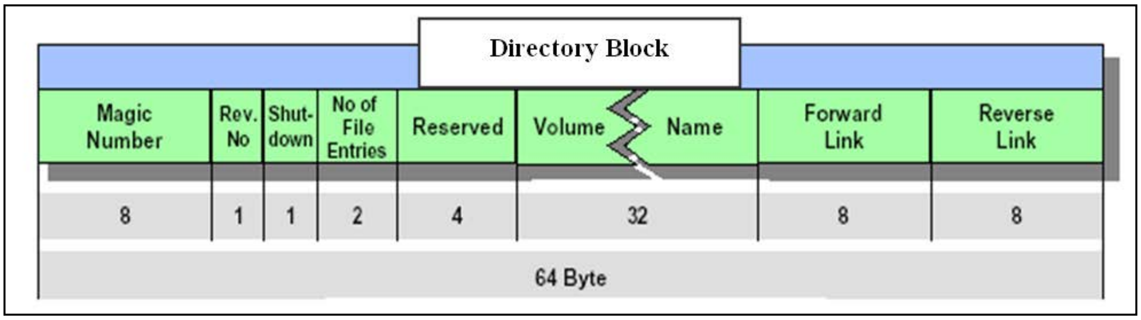


Figure 10-7. Directory Block

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 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 64 KB 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 10-4. The directory entry for a data file, as shown in Table 10-5, 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 10-4. Directory Block Format

Field Name	Bytes	Description	Data Type
Magic Number	8	An identifier for a directory block. This identifier supports discovery of lost directory entries and directory reconstruction after a fault. The value is BCS “FORTYtwo” (0x464F52545974776F)	BCS

Table 10-4. Directory Block Format

Field Name	Bytes	Description	Data Type
Revision Number	1	Revision number of the standard compiled by the recording system. 0x01 = RCC 106-03 through RCC 106-05 0x0F = RCC 106-07 or later	Unsigned Binary
Shutdown	1	Flag, if cleared to a 0x00, indicates that the volume was not properly dismounted, and if seen on power-up is an indication that the directory chain may be faulty. If set = 0xFF, then the file system properly shut down. 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 10-5.	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 this block address 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 10-5
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 as unused and should be filled with 0xFF.	Unsigned Binary
Note: 64 bytes in fixed fields.			

Table 10-5. Data 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

Table 10-5. Data File Entry Format

Field Name	Bytes	Description	Data Type
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 hours in a 24-hour-based day, 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 = Universal Coordinated Time (UTC) (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

Table 10-5. Data File Entry Format

Field Name	Bytes	Description	Data Type
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 hours in a 24-hour-based day, 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 Subsection 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.

Deleted files are not applicable to 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 Chapter 10 implementations and shall be filled with 0xFF. Reserved fields are intended for future 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 Section 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 MSB 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 [CSDW], Intra-Packet Data Header [IPDH], and Packet Trailer) described in Section 10.6 of this standard are defined to have the following bit and byte orientation. The least significant byte shall be transmitted first, the LSB 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 10646:2012.[6] The NATO Imagery Interoperability Architecture limits characters to a subset rather than allowing all characters. The subset will be single octets, known as the 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 10-6.

[6] International Organization for Standardization/International Electrotechnical Commission. *Information Technology --Universal Coded Character Set (UCS)*. ISO/IEC 10646:2012. May 2012. May be superseded by update. Available at http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html.

Table 10-6. 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 Figure 10-8.

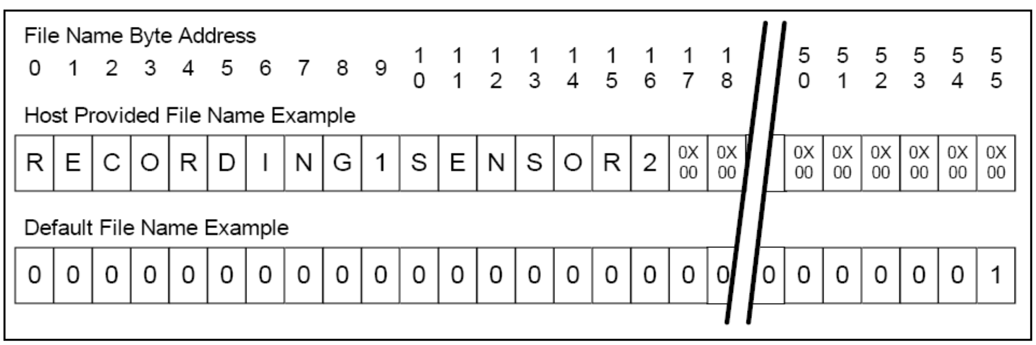


Figure 10-8. File Name Examples

10.6 Data Format Definition

10.6.1 Common Packet Elements.

Data shall have three required parts: a packet header, a packet body, and 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 10-9.

a. A packet has the basic structure shown in Table 10-7. 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 Table 10-8 for a diagram of the generic packet format. This table 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, Table 10-8 shows the generic packet in a 32-bit, Little-endian format, and assumes 16-bit data words and data checksum.

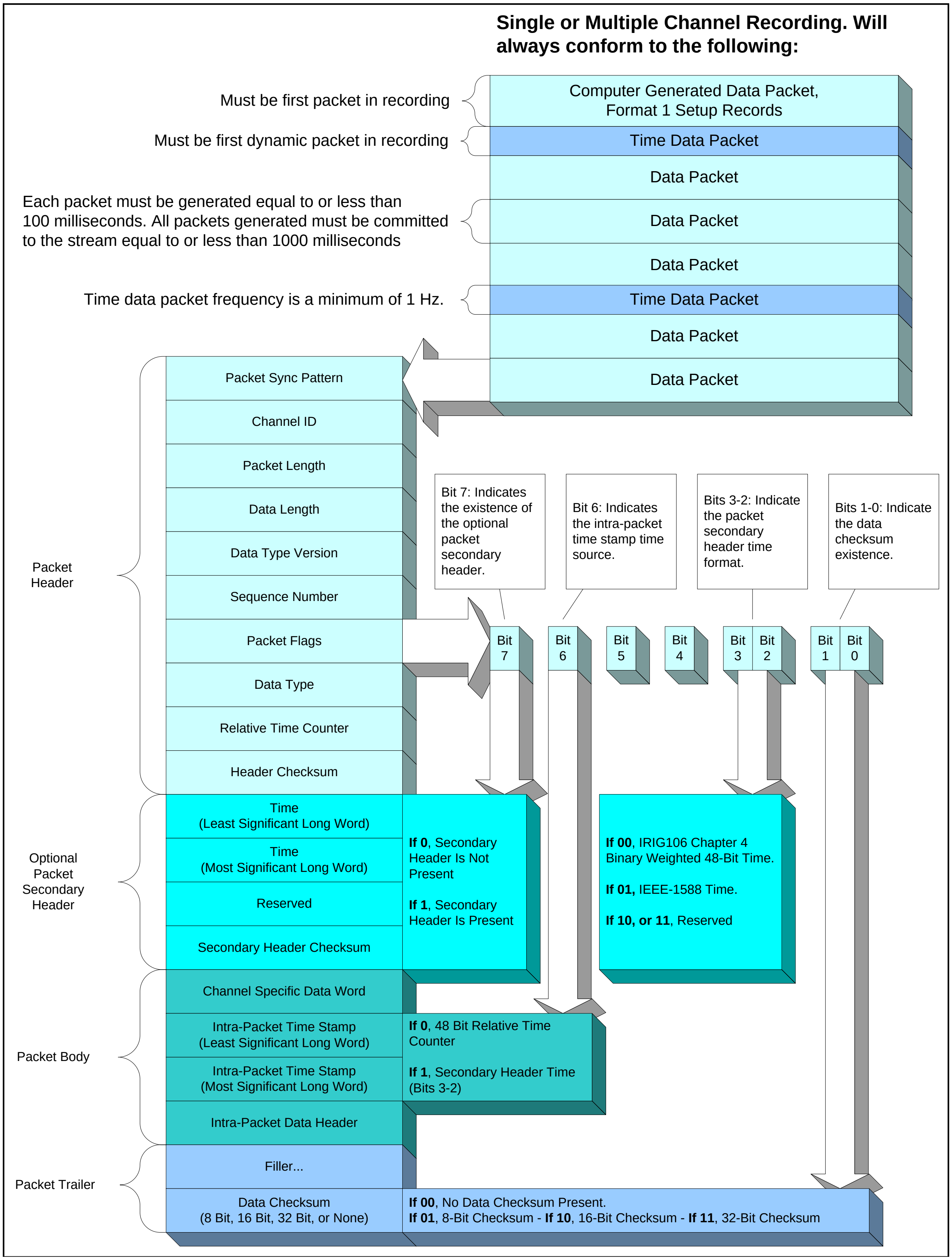


Figure 10-9. Data Recording Structure

Table 10-7. General Packet Format

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

Table 10-8. 32-Bit Packet Format Layout

MSB			LSB	
31	16	15	0	
CHANNEL ID		PACKET SYNC PATTERN		
PACKET LENGTH				
DATA LENGTH				
DATA TYPE	PACKET FLAGS	SEQUENCE NUMBER	DATA TYPE VERSION	Packet Header
RELATIVE TIME COUNTER				
HEADER CHECKSUM		RELATIVE TIME COUNTER		
TIME (LEAST SIGNIFICANT LONG WORD [LSLW])				(Optional) Packet Secondary Header
TIME (MOST SIGNIFICANT LONG WORD [MSLW])				
SECONDARY HEADER CHECKSUM		RESERVED		
CHANNEL-SPECIFIC DATA				
INTRA-PACKET TIME STAMP 1				
INTRA-PACKET TIME STAMP 1				
INTRA-PACKET DATA HEADER 1				
DATA 1 WORD 2		DATA 1 WORD 1		

DATA 1 WORD N	:	
INTRA-PACKET TIME STAMP 2		
INTRA-PACKET TIME STAMP 2		
INTRA-PACKET DATA HEADER 2		
DATA 2 WORD 2	DATA 2 WORD 1	Packet
DATA 2 WORD N	:	Body
:		
INTRA-PACKET TIME STAMP N		
INTRA-PACKET TIME STAMP N		
INTRA-PACKET DATA HEADER N		
DATA N WORD 2	DATA N WORD 1	
DATA N WORD N	:	
[FILLER]		Packet Trailer
DATA CHECKSUM		

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 10-10.

MSB 7	LSB 0	
[Filler]		Packet Trailer
Data Checksum (LSB)		
Data Checksum		
Data Checksum		
Data Checksum (MSB)		

Figure 10-10. Packet Trailer for 32-Bit Data Checksum

b. For an 8-bit data checksum, the packet trailer would be as shown in Figure 10-11.

MSB 7	LSB 0	
[Filler]		Packet Trailer
Data Checksum		

Figure 10-11. Packet Trailer for 8-Bit Data Checksum

c. The size of a single packet may be a maximum of 524,288 (2^{19}) bytes as shown in Table 10-9. 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 (2^{27}) 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.

Table 10-9. 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 Packet, Format 1 setup records.
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.

d. With the exception of computer-generated packets, all other packet generation times shall be equal to or less than 100 milliseconds (ms) as measured by the 10-megahertz (MHz) relative time counter (RTC) whenever data is available. This requirement ensures that a packet shall contain equal to or less than 100 ms worth of data, and that a packet containing any data must be generated equal to or less than 100 ms 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. All packets that are generated shall contain data. Filler only, idle (as defined by medium or interface) only, or empty packets shall not be allowed.

f. All reserved bit fields in packet headers or CSDWs 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 ms as measured by the 10-MHz RTC 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.

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 as shown in Table 10-8 and defined below.

a. Packet Sync Pattern. These 2 bytes contain a static sync value for the every packet. The packet sync pattern value shall be 0xEB25.

b. Channel ID. These 2 bytes contain 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 MSBs (from the channel ID) that distinguish the multiplexer source ID. The remaining LSBs of the channel ID field shall be the channel ID for each data source acquired by the multiplexer.

   CHANGE (2) Reserved Channel ID. Channel ID 0x0000 is reserved, and as of 106-13 is used to insert only the Computer-Generated Data Packet, Format 1 setup record(s) into the composite data stream.

   CHANGE (3) Available Channel IDs. All values not comprising the reserved channel ID are available. As of 106-13, when Computer-Generated Data Packet, Formats 0 and 2-7 reside in channel ID 0x0001-0xFFFF, only one packet type shall exist per channel ID.

c. Packet Length. These 4 bytes contain 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 (IPHs), data, filler, and data checksum.

d. Data Length. These 4 bytes contain a value representing the valid data length within the packet. This value shall be represented in bytes. Valid data length includes channel-specific data, IPDHs, intra-packet time stamp(s) (IPTS), and data but does not include packet trailer filler and data checksum.

e. Data Type Version. This byte contains a value at or below the release version of the standard applied to the data types in Table 10-10. The value shall be represented by the following bit patterns:

   0x00 = Reserved
   0x01 = Initial Release (RCC 106-04)
   0x02 = RCC 106-05
   0x03 = RCC 106-07
   0x04 = RCC 106-09
   0x05 = RCC 106-11
   0x06 = RCC 106-13

0x07 through 0xFF = Reserved

Table 10-10. 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	0x06
0x01	Computer-Generated Data, Format 1	Setup Record	0x06
0x02	Computer-Generated Data, Format 2	Recording Events	0x06
0x03	Computer-Generated Data, Format 3	Recording Index	0x06
0x04 - 0x07	Computer-Generated Data, Format 4-Format 7	Reserved for future use	0x06
0x08	PCM Data, Format 0	Reserved for future use	0x06
0x09	PCM Data, Format 1	Chapter 4 or 8	0x06
0x0A - 0x0F	PCM Data, Format 2 - Format 7	Reserved for future use	0x06
0x10	Time Data, Format 0	Reserved for future use	0x06
0x11	Time Data, Format 1	RCC/Global Positioning System [GPS]/RTC	0x06
0x12-0x17	Time Data, Format 2-Format 7	Reserved for future use	0x06
0x18	MIL-STD-1553 Data, Format 0	Reserved for future use	0x06
0x19	MIL-STD-1553 Data, Format 1	MIL-STD-1553B Data	0x06
0x1A	MIL-STD-1553 Data, Format 2	16PP194 Bus	0x06
0x1B-0x1F	MIL-STD-1553 Data, Format 3-Format 7	Reserved for future use	0x06
0x20	Analog Data, Format 0	Reserved for future use	0x06
0x21	Analog Data, Format 1	Analog Data	0x06
0x22-0x27	Analog Data, Format 2-Format 7	Reserved for future use	0x06
0x28	Discrete Data, Format 0	Reserved for future use	0x06
0x29	Discrete Data, Format 1	Discrete Data	0x06
0x2A-0x2F	Discrete Data, Format 2-Format 7	Reserved for future use	0x06
0x30	Message Data, Format 0	Generic Message Data	0x06
0x31-0x37	Message Data, Format 1-Format 7	Reserved for future use	0x06
0x38	ARINC-429 Data, Format 0	ARINC-429 Data	0x06
0x39- 0x3F	ARINC-429 Data, Format 1-Format 7	Reserved for future use	0x06
0x40	Video Data, Format 0	MPEG-2/H.264 Video	0x06
0x41	Video Data, Format 1	ISO 13818-1 MPEG-2	0x06
0x42	Video Data, Format 2	ISO 14496 MPEG-4 Part 10 110 AVC/H.264	0x06
0x43-0x47	Video Data, Format 3-Format 7	Reserved for future use	0x06
0x48	Image Data, Format 0	Image Data	0x06
0x49	Image Data, Format 1	Still Imagery	0x06
0x4A-	Image Data, Format 2-	Dynamic Imagery	0x06
0x4B-0x4F	Image Data, Format 3-Format 7	Reserved for future use	0x06
0x50	UART Data, Format 0	UART Data	0x06

Table 10-10. Data Type Names and Descriptions

Packet Header Value	Data Type Name	Data Type Description	Current Data Type Version
0x51-0x57	UART Data, Format 1-Format 7	Reserved for future use	0x06
0x58	IEEE 1394 Data, Format 0	IEEE 1394 Transaction	0x06
0x59	IEEE 1394 Data, Format 1	IEEE 1394 Physical Layer	0x06
0x5A-0x5F	IEEE 1394 Data, Format 2-Format 7	Reserved for future use	0x06
0x60	Parallel Data, Format 0	Parallel Data	0x06
0x61-0x67	Parallel Data, Format 1-Format 7	Reserved for future use	0x06
0x68	Ethernet Data, Format 0	Ethernet Data	0x06
0x69-	Ethernet Data, Format 1-	Ethernet UDP Payload	0x06
0x6A-0x6F	Ethernet Data, Format 2-Format 7	Reserved for future use	0x06
0x70	TSPI/CTS Data, Format 0	GPS NMEA-RTCM	0x06
0x71	TSPI/CTS Data, Format 1	EAG ACMI	0x06
0x72	TSPI/CTS Data, Format 2	ACTTS	0x06
0x73- 0x77	TSPI/CTS Data, Format 3-Format 7	Reserved for future use	0x06
0X78	Controller Area Network Bus	CAN Bus	0x06

f. Sequence Number. This 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. This 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 IPTS time source.

0 = Packet header 48-bit RTC.
1 = Packet secondary header time (bit 7 must be 1).

Bit 5: RTC sync error.

0 = No RTC sync error.
1 = RTC 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 = Chapter 4 binary weighted 48-bit time format. The two LSBs of the 64-bit packet secondary header time and IPTS shall be zero-filled.
01 = IEEE 1588 time format. The packet secondary header time and each IPTS shall contain a 64-bit timestamp represented IAW the time representation type as specified by IEEE STD 1588-2008.[7] The 32 bits indicating seconds shall be placed into the MSLW portion of the secondary header and the 32 bits indicating nanoseconds shall be placed into the LSLW portion.
10 = 64-bit binary extended relative time counter (ERTC) with 1-nanosecond resolution. The counter shall be derived from a free-running 1-gigahertz (GHz) clock - similar to the RTC described below - just with higher resolution. When this option is used, the 10-MHz RTC shall be synchronized with the ERTC (RTC = ERTC/100).
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. This 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 10-10 lists the data types and their descriptions.

i. Relative Time Counter. These 6 bytes contain a value representing the 10-MHz RTC. This is a free-running 10-MHz binary counter represented by 48 bits that 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	If enabled, the applicable data bit of the 48-bit value of the packet secondary time value shall correspond to the first bit of the data in the packet body (unless it is defined in each data type section).

[7] Institute of Electrical and Electronics Engineers. *IEEE standard for a precision clock synchronization protocol for networked measurement and control systems.* IEEE 1588-2008. Geneva: International Electrotechnical Commision, 2008.

j. Header Checksum. These 2 bytes contain 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. These 8 bytes contain the value representing time in the format indicated by bits 2 and 3 of the packet flags in Subsection 10.6.1.1 item g. The secondary header can be enabled on a channel-by-channel basis but all channels that have a secondary header must use the same time source in bits 2-3 of the packet flags.

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

When Chapter 4 binary weighted time is used, time shall be stored as shown in Figure 10-12.

MSB 31 16	LSB 15 0
Micro-Seconds Word	Reserved
High Order Time Word	Low Order Time Word

Figure 10-12. Secondary Header Chapter 4 Time

When IEEE 1588 time is used time shall be stored as shown in Figure 10-13.

MSB LSB 31 0
Nanoseconds Word
Seconds Word

Figure 10-13. Secondary Header IEEE 1588 Time

When ERTC time is used time shall be stored as shown in Figure 10-14.

NEW

MSB LSB 31 0
LSLW
MSLW

Figure 10-14. Secondary Header ERTC Time

b. Reserved. These 2 bytes are reserved and shall be zero filled.

c. Secondary Header Checksum. These 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 in size, this 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. These 8 bytes contain time in either 48-bit RTC format (plus 16 high-order zero bits) or 64-bit format as specified in the packet flags in the packet header. The IPTSs are only mandatory where defined by the data formats.

c. Intra-Packet Data Header. Variable in size, this contains additional time, status, data, and/or format information pertaining to the data items that follow. The IPDHs are only mandatory where defined by the data formats.

d. Data. With n bytes, this contains valid data from a particular channel as defined within the data formats contained within this standard.

NOTE	The IPTS and the IPDH are collectively called the IPH. In some cases, an IPH may only have a time stamp (zero-length data header), while in other cases, the IPH only has a data header (zero-length time stamp). Some data types have no IPH. The IPH requirements are specified separately for each data type.

NOTE	The 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 in size, all filler shall be set to 0x00 or 0xFF.

b. 8-Bit Data Checksum. This 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 Subsection 10.6.1.1 item g).

c. 16-Bit Data Checksum. These 2 bytes contain a value representing a 16-bit arithmetic sum of the words in the packet. Only inserted if packet flag bits are set (Subsection 10.6.1.1 item g).

d. 32-Bit Data Checksum. These 4 bytes contain 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 (Subsection 10.6.1.1 item 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 (Chapter 4 and Chapter 8).

A packet with Chapter 4 or Chapter 8 PCM data has the basic structure as shown in Table 10-11. 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.

Table 10-11. General PCM Data Packet, Format 1

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

The user may separately enable or disable word unpacking on each active PCM channel. Word unpacking will force the LSB 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 10-15.

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

Figure 10-15. Pulse Code Modulation Packet Channel-Specific Data Format

- Reserved. Bit 31 is reserved.
- Intra-Packet Header. Bit 30 indicates if IPHs (IPTS and IPDH) are inserted before each minor frame. The IPHs are only optional because of the mode selection. This determines whether IPHs are included or omitted.

  0 = The IPHs are omitted for throughput mode.
  1 = The IPHs 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 indicate 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 Chapter 4, Subsection 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 indicate 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 contain 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 IPHs 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 IPHs 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 exists 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 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 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 16-bit alignment mode PCM packets are as shown in Table 10-12. Given PCM frames with a 24-bit minor sync pattern and n data words where the bit-lengths of data words 1, 2, 3, and 4 are 12, 16, 8, and 10 respectively, the resultant 32-bit alignment mode PCM packets are as shown in Table 10-13.

Table 10-12. PCM Data-Unpacked (16-Bit Alignment 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)	
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	
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

Table 10-13. PCM Data-Unpacked (32-Bit Alignment 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)		
Intra-Packet Data Header (Bits 31-16)		
4 Bits Pad	12 Bits Sync (Bits 11-0)	
4 Bits Pad	12 Bits Sync (Bits 23-12)	
16 Bits Word 2 Data		
4 Bits Pad	12 Bits Word 1 Data	
6 Bits Pad	10 Bits Word 4 Data	
8 Bits Pad		8 Bits Word 3 Data
:		
Word N Data Bits + Pad If Needed		
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		

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 16-bit alignment mode, then the resultant PCM packets are as shown in Table 10-14. Table 10-15 shows the resultant PCM packets for 32-bit alignment mode.

Table 10-14. PCM Data-Packed (16-Bit Alignment 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	

Table 10-15. PCM Data-Packed (32-Bit Alignment 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)	
Intra-Packet Data Header (Bits 31-16)	
Data Word 2	
Data Word 1	
Data Word 4	
Data Word 3	
:	
Filler Bits	X Data Bits
16 Filler Bits (If Required to Maintain 32-Bit Alignment)	
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	

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 Table 10-16 and Table 10-17.

Table 10-16. PCM Data-Throughput (16-Bit Alignment Mode) Sample Packet

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	

Table 10-17. PCM Data-Throughput(32-Bit Alignment Mode) Sample Packet

MSB 15	LSB 0
Packet Header	
Channel-Specific Data (Bits 15-0)	
Channel-Specific Data (Bits 31-16)	
PCM Stream Bits 17-32	
PCM Stream Bits 1-16	
PCM Stream Bits 49-64	
PCM Stream Bits 33-48	
:	
Packet Trailer	

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 10-16 and Figure 10-17 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 10-16. 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 10-17. 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 IPH containing a 64-bit IPTS and a 16- or 32-bit IPDH, 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 IPH is either 10 or 12 bytes (80 or 96 bits) positioned contiguously, as depicted in Figure 10-18. In 16-bit alignment mode, the IPDH length is fixed at 2 bytes. A 32-bit alignment mode requires a 4-byte IPDH, 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 10-18. Pulse Code Modulation Intra-Packet Header

- Intra-Packet Time Stamp. These 8 bytes indicate 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 RTC 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item 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 indicate 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 10-19), the time packet will be generated at a minimum frequency of 1 hertz.

MSB 31 12	11 8	7 4	LSB 3 0
Reserved	DATE	FMT	SRC

Figure 10-19. Time Packet Channel-Specific Data Format

- Inter-Range Instrumentation Group (IRIG) Time Type Formats. The 10-MHz RTC shall be captured for insertion into the time packet data header IAW IRIG 200.
- 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 ms 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 Table 10-18. 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 IPHs.

Table 10-18. General Time Data Packet, Format 1

Packet Header
Channel-Specific Data
Time Data
Packet Trailer

a. Time Packet Channel-Specific Data. The packet body portion of each time data packet begins with a CSDW formatted as shown in Figure 10-19.

- Reserved. Bits 31-12 are reserved.
- Date Format (DATE). Bits 11-8 indicate 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 10-20)
1 = Month and year available (Figure 10-21)

<table>
<tr><td colspan="8">MSB</td><td colspan="8" align="right">LSB</td></tr>
<tr><td>15</td><td colspan="3">14 12</td><td colspan="4">11 8</td><td colspan="4">7 4</td><td colspan="4">3 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 10-20. Time Data-Packet Format, Day Format

<table>
<tr><td colspan="8">MSB</td><td colspan="8" align="right">LSB</td></tr>
<tr><td>15</td><td colspan="3">14 12</td><td colspan="4">11 8</td><td colspan="4">7 4</td><td colspan="4">3 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 10-21. Time Data-Packet Format, Day, Month, and Year Format

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 indicate 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 through 0xE = Reserved

  0xF = None (time packet payload invalid)

- Time Source (SRC). Bits 3-0 indicate 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

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

b. Time Packet Body. After the CSDW, the time data words are inserted in the packet in binary-coded decimal (BCD) format as shown in Figure 10-20 and Figure 10-21 (units of measure presented in Table 10-19).

Table 10-19. Units of Measure

Tmn	Tens of ms	TDn	Tens of days
Hmn	Hundreds of ms	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).

Data in the MIL-STD-1553 bus format is packetized as messages, with each 1553 bus transaction recorded as a message. A transaction is a bus controller (BC)-to-remote terminal (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 CSDW formatted as shown in Figure 10-22.

MSB 31 30	29 24	23 LSB 0
TTB	RESERVED	MSGCOUNT

Figure 10-22. 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 IPH 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 indicate 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 within MIL-STD-1553 messages has the basic structure shown in Table 10-20. 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.

Table 10-20. Military Standard 1553 Data Packet, Format 1 Basic Layout

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

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 IPH consisting of an IPTS and an IPDH.

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

- The 48-bit RTC that corresponds to the data bit indicated in the MIL-STD-1553 channel-specific data, time tag bits (Subsection 10.6.4.2 item a) 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and to the data bit indicated in the MIL-STD-1553 channel-specific data, time tag bits (Subsection 10.6.4.2 item a).

(2) MIL-STD-1553 Intra-Packet Data Header. The length of the IPDH is fixed at 6 bytes (48 bits) positioned contiguously, in the following sequence (Figure 10-23).

MSB 15	LSB 0
Block Status Word	
Gap Times Word	
Length Word	

Figure 10-23. 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 10-24.

MSB 15 14	13	12	11	10	9	8 6	5	4	3	2 LSB 0
R	BID	ME	RR	FE	TM	RESERVED	LE	SE	WE	RESERVED

Figure 10-24. 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 14 microseconds (μs) defined by MIL-STD-1553B.[8]

  0 = No response time out
  1 = Response time out

[8] Department of Defense. Aircraft Internal Time Division Command/Response Multiplex Data Bus. MIL-STD-1553B. 30 April 1975. May be superseded by update. Available at http://quicksearch.dla.mil/basic_profile.cfm?ident_number=36973&method=basic.

- 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 RT response times as defined by MIL-STD-1553. The resolution of the response time shall be in tenths of µs. A maximum of two response time words can exist. Messages of RT-to-RT type 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 RT response.

- Gap Times Word (bits 15-0). The gap times word indicates the number of tenths of µs 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 10-25.

MSB 15 8	LSB 7 0
GAP2	GAP1

Figure 10-25. 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 Table 10-21.

Table 10-21. Military Standard 1553 Data Packet, Format 1

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

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 16PP362B.[9] A 16PP194 transaction consists of six each 32-bit words consisting of a 16PP194 COMMAND (1), COMMAND (1) ECHO, COMMAND (2), COMMAND (3) GO/NO GO, COMMAND (4) GO/NO GO, and STATUS as illustrated in Figure 10-26. Multiple transactions may be encoded into the data portion of a single packet.

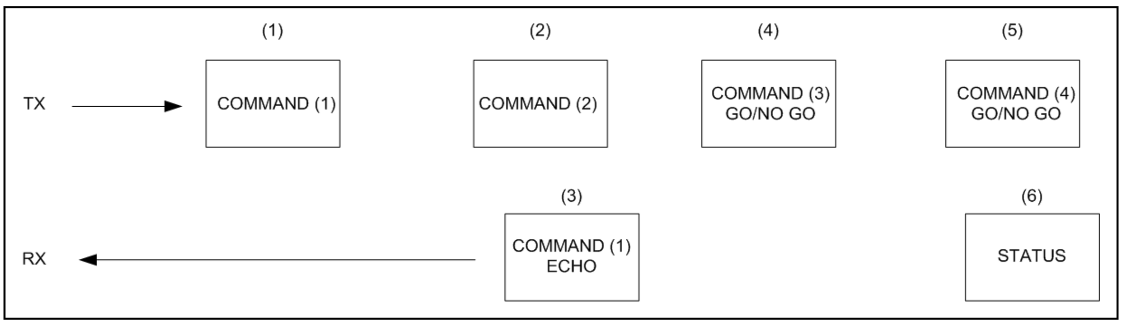


Figure 10-26. 19PP194 Message Transaction

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 CSDW formatted as shown in Figure 10-27.

MSB 31	LSB 0
MSGCOUNT	

Figure 10-27. Military Standard 1553 16PP194 Packet Channel-Specific Data Format

- Message Count (MSGCOUNT). Bits 31-0 indicate 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 Table 10-22 below. This drawing is merely to represent relative placement of data in the packet.

[9] Lockheed Martin Corporation. "Advanced Weapons Multiplex Data Bus." 8 June 2010. May be superseded by update. Available to RCC members with Private Portal access at https://wsdm.wsmr.army.mil/site/rccpri/Publications/106-13_Part%20I%20-%20Telemetry%20Standards/16PP362B.pdf.

Table 10-22. Military Standard 1553 16PP194 Data Packet Basic Layout

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	Intra-Packet Data Header Status Word
Data N	
.	
.	
PACKET TRAILER	

c. MIL-STD-1553 16PP194 Intra-Packet Header. The IPH consists of the IPDH (LENGTH and STATUS) and the IPTS.

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

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

- MIL-STD-1553 16PP194 Intra-Packet Data Header STATUS. 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 10-28.

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

Figure 10-28. Military Standard 1553 16PP194 Intra-Packet Data Header Format

- Transaction Error (TE). Bit 15 indicates an 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 IPTS (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 IPDH, the first word following the length should always be the first transaction command word. The resultant packets have the format shown in Table 10-23.

Table 10-23. Military Standard 1553 16PP194 Data Packet

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 No-Go (Bits 31-16)	

Command(3) Go No-Go (Bits 15-0)
Command(4) Go No-Go Echo (Bits 31-16)
Command(4) Go No-Go 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 No-Go (Bits 31-16)
Command(3) Go No-Go (Bits 15-0)
Command(4) Go No-Go Echo (Bits 31-16)
Command(4) Go No-Go Echo (Bits 15-0)
Status (Bits 31-16)
Status (Bits 15-0)
Packet Trailer

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 10-29). The corresponding 16PP194 sync and parity bits will not be formatted into the 16PP194 words.

MSB 15 13	12 10	9	8	7 LSB 0
BUS ID	GAP	W	P	16PP194 Data Word (bits 24-17)
16PP194 Data Word (bits 16-1)				

Figure 10-29. Military Standard 1553 26-Bit 16PP194 Word Format

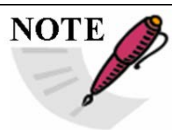
- MIL-STD-1553 16PP194 Bus ID (BUS ID). A three-bit field shall be used to indicate bus identification as follows.

111	Communication interface unit (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> 9.15 µs
110	7.55 µs <GAP <=9.15 µs
101	5.95 µs<GAP <=7.55 µs
100	4.35 µs<GAP <=5.95 µs
011	2.75 µs<GAP <=4.35 µs
010	2.75 µs<GAP <=4.35 µs
001	1.15 µs<GAP <=2.75 µs
000	Not Applicable

	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 1-µs 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 10-30.

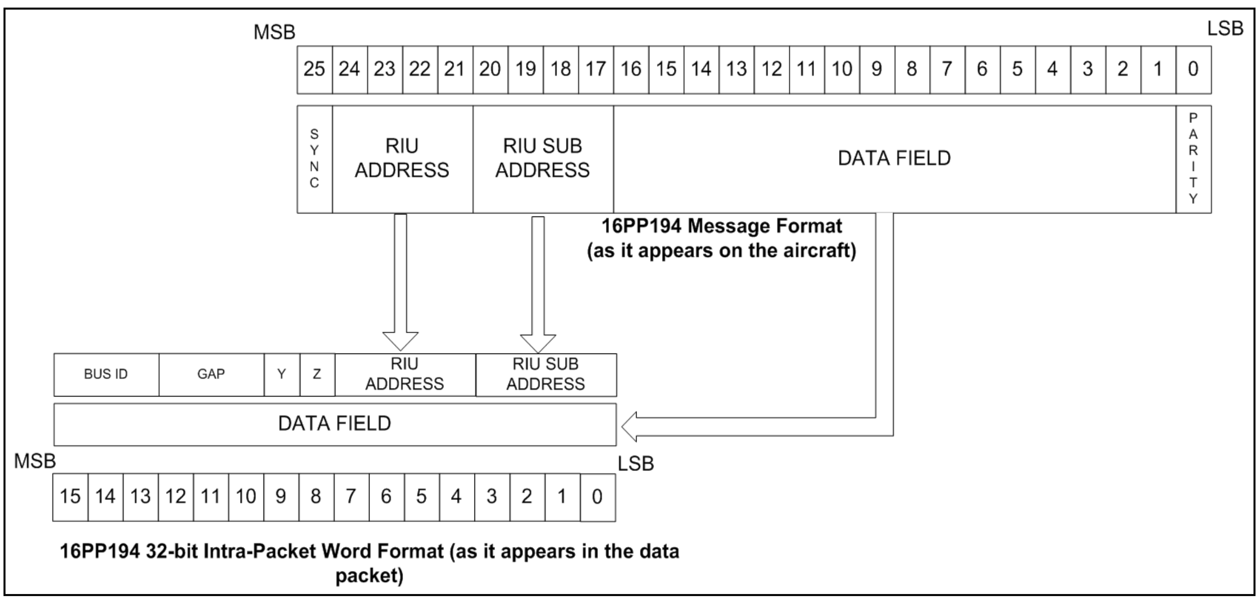


Figure 10-30. 16PP194 Word Format

- 16PP194 Data Word (bits 24-17): 16PP194 remote interface unit (RIU) address and RIU subaddress as in Figure 10-30.

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 Table 10-24.

Table 10-24. Generic Analog Data Packet, Format 1

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

An analog data packet will contain a CSDW for each subchannel of analog data sampled within that packet if the SAME bit is set to 0, or it will contain a single CSDW 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 CSDW, 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 CSDWs will usually be less than the number of samples.

Table 10-24 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.

NOTE	The packet header time in an analog data packet shall correspond to the first data sample in the packet. There are no IPHs in analog data packets.

a. Analog Packet Channel-Specific Data. The packet body portion of each analog packet begins with the CSDW(s). Each subchannel that is sampled with the packet sampling schedule must have a CSDW within the packet. Only one CSDW 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 CSDW shall be used to indicate same sampling data rate for subchannels.

The CSDWs for analog data packets are formatted as shown in Figure 10-31.

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

Figure 10-31. Analog Packet Channel-Specific Data Word

- Reserved. Bits 31-29 are reserved.
- Same. Bit 28 specifies if this CSDW applies for all the channels included in the packet or if each channel has its own CSDW.

  0 = Each analog channel has its own CSDW.
  1 = The CSDW is valid for all analog channels stored in this packet.

- Factor. Bits 27-24 are 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 20 = 1 => factor = 1/1
  0x1 = Sampling rate factor denominator 21 = 2 => factor = ½
  0x2 = Sampling rate factor denominator 22 = 4 => factor = ¼
  :
  0xF = Sampling rate factor denominator 215 = 32768 => factor = 1/32768

- Totchan. Bits 23-16 indicate the total number of analog subchannels in the packet (and the number of CSDWs in the packet).

  This Totchan field must be the same value in all CSDWs 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 Telemetry Attributes Transfer Standard (TMATS) (Computer-Generated Data, Format 1) packet.

  0x00 = 256 subchannels
  0x01 = 1 subchannel
  0x02 = 2 subchannels

- Subchan. Bits 15-8 indicate a binary value representing the number or subchannel ID of the analog subchannel.

When an analog packet contains data from more than one subchannel and the CSDWs are not the same for all channels (see field Same, bit 28), the CSDWs must be inserted into the packet in ascending subchannel number as identified by this Subchan field. The Subchan values in these CSDWs 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 indicate a binary value representing the number of bits in the analog-to-digital converter.

  000000 = 64-bit samples
  000001 = 1-bit samples
  :
  001000 = 8-bit samples
  :
  001100 = 12-bit samples
  :

- Mode. Bits 1-0 indicate alignment and packing modes of the analog data. When TOTCHAN is more than 1, 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

NOTE

For the special case of defining a single channel (Totchan = 1), there are two options: a) one channel with no sub-channels or b) one channel as its own sub-channel. In such cases the bits are to be defined as follows.

	One channel with no sub-channel	One channel with one sub-channel
Totchan (bits 23-16)	1	1
Same (bit 28)	1	0
Subchan (bits 15-8)	0	1

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 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 CSDW(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 Subsection 10.6.5.2 items b(1) and 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 CSDWs). 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 CSDWs. 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:

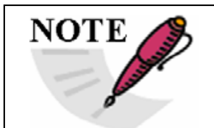
Simultaneous Sample 1:	Subchannel 3
Simultaneous Sample 1:	Subchannel 5
Simultaneous Sample 2:	Subchannel 1
Simultaneous Sample 2:	Subchannel 3
Simultaneous Sample 2:	Subchannel 4
Simultaneous Sample 2:	Subchannel 5
Simultaneous Sample 3:	Subchannel 3
Simultaneous Sample 3:	Subchannel 5
Simultaneous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

(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 pad bits are added to 12-bit words, eight 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 Table 10-25.

Table 10-25. Analog Data Packet-Unpakced Mode, MSB Padding	
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	

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 Table 10-26.

Table 10-26. Analog Data Packet-Unpacked Mode, LSB Padding

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 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	4 Pad Bits
Packet Trailer	

(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 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-(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 Table 10-27.

Table 10-27. Analog Data Packet-Packed Mode Packet

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		

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 Table 10-28. 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.

Table 10-28. General Discrete Data Packet, Format 1

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

a. Discrete Packet Channel-Specific Data Word. The packet body portion of each discrete packet begins with the CSDW, which is formatted as shown in Figure 10-32.

MSB 31 8	7 3	LSB 2 0
RESERVED	LENGTH	MODE

Figure 10-32. Discrete Packet Channel Data Word Format

- Reserved. Bits 31-8 are reserved.
- Length. Bits 7-3 indicate a binary value representing the number of bits in the event. The value of zero indicates 32 bits.
- Mode. Bits 2-0 indicate 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 10-33) is inserted in the packet. Discrete data are described as events. Each event includes the eventState for each discrete input and the corresponding intra-packet time. The eventState 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 10-33. 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 IPH consisting of an IPTS only, which is inserted immediately before the discrete event. The length of the IPH is fixed at 8 bytes (64 bits) positioned contiguously, arranged in the sequence shown in Figure 10-34.

MSB 31	LSB 0
Time (LSLW)	
Time (MSLW)	

Figure 10-34. Discrete Event Intra-Packet Header

- Intra-Packet Time Stamp. These 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 RTC 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and to the first data bit of the discrete event. The discrete data packet format is presented in Table 10-29.

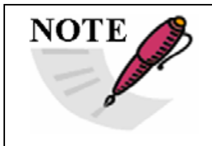
Table 10-29. Discrete Data Packet Format

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

10.6.7 Computer-Generated Data Packets.

Packets with computer-generated data have the basic structure shown in Table 10-30. 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 CSDW, 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.

Table 10-30. General Computer-Generated Data Packet Format

Packet Header
Channel-Specific Data
Computer Generated Data
Packet Trailer

10.6.7.1 Computer-Generated Data Packets Format 0, User Defined.

Format 0 enables the insertion of user-defined computer-generated data. Data cannot 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 CSDW, which is formatted as shown in Figure 10-35.

MSB 31	LSB 0
RESERVED	

Figure 10-35. 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 CSDW 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 CSDW, which is formatted as shown in Figure 10-36.

MSB 31 10	9	8	7 LSB 0
RESERVED	FRMT	SRCC	CH10VER

Figure 10-36. 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 Chapter 9 ASCII Format
  1 = Setup record IAW Chapter 9 XML Format

NOTE	It is not permissible to have both ASCII and XML 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.

- RCC 106 Chapter 10 Version (CH10VER). A 1-byte indicator of which 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 through 0x06 = Reserved
  0x07 = RCC-106-07
  0x08 = RCC-106-09
  0x09= RCC-106-11
  0x0A = RCC-106-13
  0x0B through 0xFF = Reserved

  Individual Section 10.6 data types and their format/content compliancy and applicability with the Chapter 10 release version are defined in Subsection 10.6.1.1 item e.

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 512 KB, 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 10-55 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 10-55.

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 Subsection 10.6.7.3 item c) 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 CSDW, which is formatted as shown in Figure 10-37.

MSB 31	 30	 12	 11	LSB 0
IPDH	RESERVED		REEC	

Figure 10-37. Computer-Generated Format 2 Channel-Specific Data Word

- Recording Event Intra-Packet Data Header. Bit 31 indicates the presence of the IPDH.

  0 = Recording event IPDH not present
  1 = Recording event IPDH 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 10-38), and is defined to encompass the events occurring from the time a .RECORD command (Subsection 10.7.8) is issued (if it is the first recording) until the time a .STOP command (Subsection 10.7.8) 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.

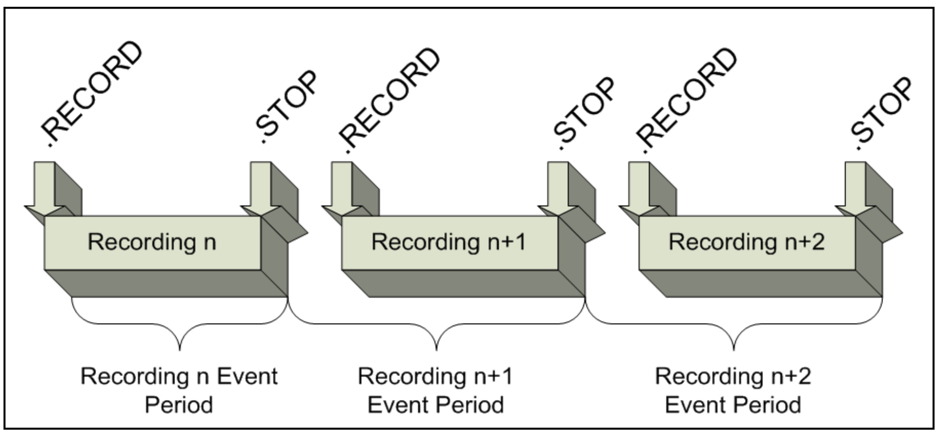


Figure 10-38. Events Recording Period

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.

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, MEASUREMENT LIMIT, or MEASUREMENT CHANGE 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.
Mode 7:	Capture MEASUREMENT CHANGE event on each change of value from the previous value.

NOTE	If Event Type is MEASUREMENT DISCRETE, MEASUREMENT LIMIT, or MEASUREMENT CHANGE, 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.

For Mode 7 capture of MEASUREMENT CHANGE event on each change of value from the previous value there shall be an option for an initial value in the setup record that does not generate an event but each successive value change from this initial value shall generate an event. Event limit counts for mode 7 shall be valid on the number of events generated based on successive value changes from each previous value.

f. Event Trigger Measurement Description. If Event Type is MEASUREMENT DISCRETE, MEASUREMENT LIMIT, or MEASUREMENT CHANGE, the event 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) (Optional) Initial measurement value (if MEASUREMENT CHANGE is used to trigger the event)

(7) Applicable measurement name engineering unit conversion

g. Recording Event Intra-Packet Time Stamp. These 8 bytes indicate the time tag of the recording event entry as follows.

(1) The 48-bit RTC 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 RTC value or, if enabled, the IPTS - whichever most accurately provides the time the event occurred; or

(2) The absolute time, if enabled by bit 6 in the packet flags (Subsection 10.6.1.1 item g). The time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item 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 or, if enabled and using an absolute time value, the IPTS - whichever most accurately provides the time the event occurred.

h. (Optional) Recording Event Intra-Packet Data Header. These 8 bytes contain 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. The format of the recording event IPH is presented in Figure 10-39.

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 10-39. Recording Event Intra-Packet Header

i. Event Packet Entry Format. Table 10-31 and Figure 10-40 present the general recording event packet format and recording event entry layout.

Table 10-31. General Recording Event Packet Format
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

MSB 31 29	28	27 12	11 LSB 0
RESERVED	EO	EVENT COUNT	NUMBER

Figure 10-40. Recording Event Entry Layout

- Reserved. Bits 31-29 are reserved for future growth and shall be zero-filled.
- Event Occurrence (EO). Bit 28 indicates EventOccurrenceState.

  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 represent 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 Subsection 10.6.7.3 item c. 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 represent 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.

- Root Index Packets. These 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 in its chain if there is more than one root index packet, 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.

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 root index type packet.

- Node Index Packets. These packets contain node item 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 10-41) will be the last packet in any recording file. More than one 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 10-MHz 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 Subsection 10.6.1.

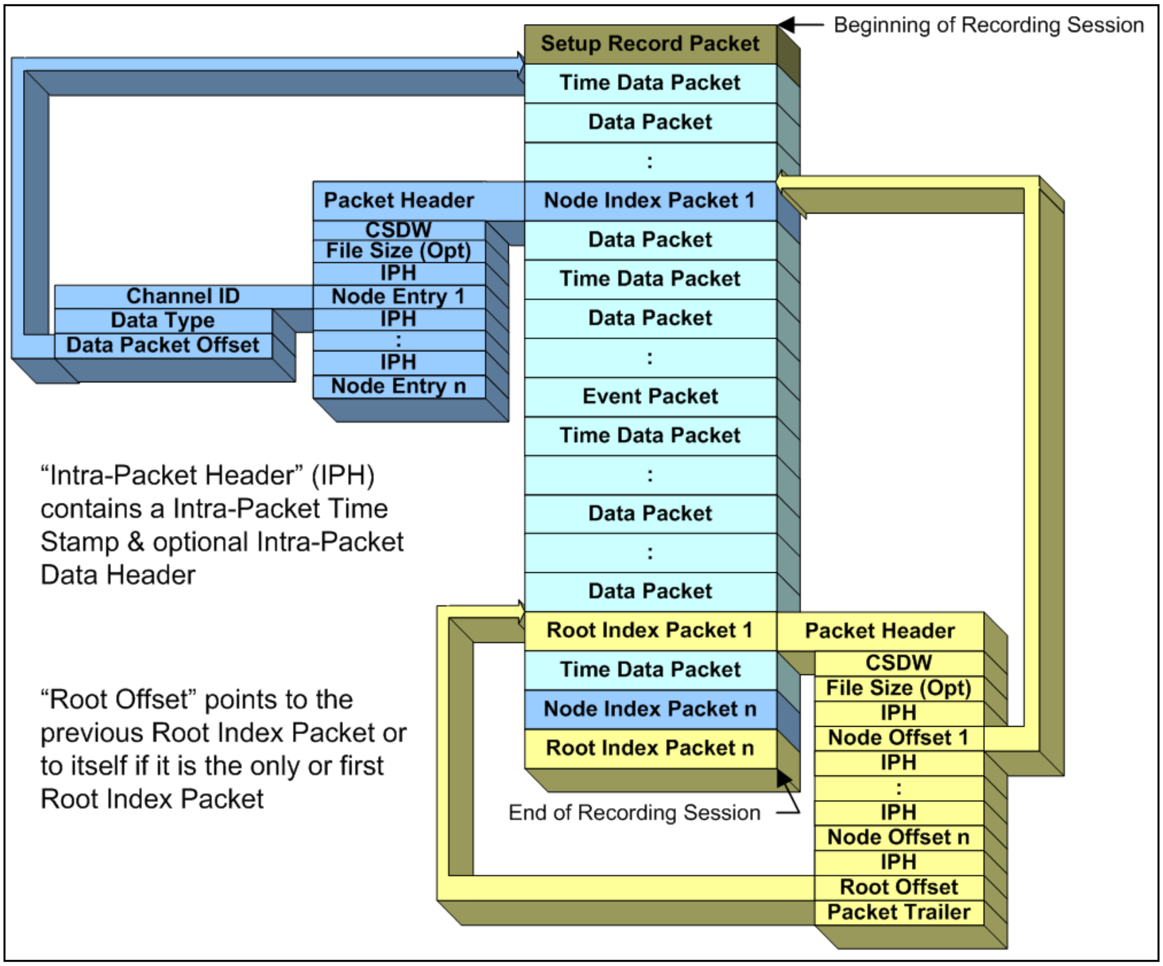


Figure 10-41. Format Showing Root Index Packet

b. Channel-Specific Data Word. The packet body portion of each Format 3 packet begins with the CSDW, which is formatted as shown in Figure 10-42.

MSB 31	30	29	28 16	15 LSB 0
IT	FSP	IPDH	RESERVED	INDEX ENTRY COUNT

Figure 10-42. 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. Bit 29 indicates the presence of the IPDH.

  0 = Index IPDH not present
  1 = Index IPDH present

- Reserved. Bits 28-16 are reserved.

- Index Entry Count. Bits 15-0 indicate 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.

	The IPDH presence once set by bit 29 shall be the same state for the entire recording.

c. Recording Index Intra-Packet Time Stamp. These 8 bytes indicate the time tag of the recording index entry as follows.

- The 48-bit RTC 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item 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. These 8 bytes contain 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 10-43 presents the format of the recording index IPH.

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 10-43. 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 Table 10-32 and Table 10-33.

Table 10-32. General Recording Root Index Packet

Table 10-32. General Recording 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

Table 10-33. Recording Root Index Entry Layout

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)	

- (Optional) Root Index File Size. These 8 bytes are 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. These 8 bytes are 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. These 8 bytes are 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 in its chain begins, if there is more than one root index packet or to itself, if it is the first or only root index packet.

f. Node Index Packet Entry Format. A node index packet contains an index entry or entries using the format shown in Table 10-34 and Figure 10-44.

Table 10-34. General Recording Node Index Packet

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

MSB 31 24	23 16	LSB 15 0
Reserved	Data Type	Channel ID
Data Packet Offset (LSLW)		
Data Packet Offset (MSLW)		

Figure 10-44. Recording Node Index Entry Layout

- (Optional) Node Index File Size. These 8 bytes are 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. These 2 bytes are an unsigned binary that identifies a value representing the packet channel ID for the data packet associated with this node index item.
- Data Type. This byte is 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. These 8 bytes are 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 IPH containing an IPDH only with an identifier (ID word) that provides type and status information. The IPH does not contain an IPTS. 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 Table 10-35.

Table 10-35. ARINC-429 Data Packet Format

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	

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 CSDW formatted as shown in Figure 10-45.

MSB 31 16	15 LSB 0
RESERVED	MSGCOUNT

Figure 10-45. ARINC-429 Packet Channel-Specific Data Word Format

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

b. Intra-Packet Data Header. Bits 31-0 contain the ARINC-429 ID word. Each ARINC-429 bus data word is preceded by an ID word and the bit definitions are as shown in Figure 10-46.

MSB 31 24	23	22	21	20	19 LSB 0
BUS	FE	PE	BS	R	GAP TIME

Figure 10-46. Intera-Packet Data Header Format

- Bus. Bits 31-24 indicate a binary value identifying the ARINC-429 bus number associated with the following data word. The first bus is indicated by 0. 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 contain 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-µs increments. The gap time of the first word in the packet is GAP TIME=0. When the gap time is longer than 100 ms, a new packet must be started.

c. ARINC-429 Packet Data Words. 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 (Table 10-36).

Table 10-36. Message Data Packet Format

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	

a. Message Packet Channel-Specific Data Word. The packet body portion of each message data packet begins with a CSDW. 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 CSDW is formatted for the complete type of packet body as shown in Figure 10-47.

MSB 31 18	17 16	LSB 15 0
RESERVED	TYPE	COUNTER

Figure 10-47. Complete Message Channel-Specific Data Word Format

- Reserved. Bits 31-18 are reserved.
- Type. Bits 17-16 indicate the type of serial packet.

  00 = One or more complete messages
  01 = Reserved
  10 = Reserved
  11 = Reserved

- Counter. Bits 15-0 contain a binary value indicating the number of messages included in the packet.

c. Segmented Message Channel-Specific Data Word. The CSDW is formatted for the segmented type of packet body as shown in Figure 10-48.

MSB 31 18	17 16	LSB 15 0
RESERVED	TYPE	COUNTER

Figure 10-48. Segmented Message Channel-Specific Data Word Format

- Reserved. Bits 31-18 are reserved.
- Type. Bits 17-16 indicate 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 contain 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 gigabytes (combined with the 16-bit Message Length field; see description in item d 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 IPH that has both an IPTS and an IPDH containing a message ID word. The length of the IPH is fixed at 12 bytes (96 bits) positioned contiguously, in the sequence shown in Figure 10-49.

MSB 31	LSB 0
Time (LSLW)	
Time (MSLW)	
Message ID Word	

Figure 10-49. Message Data Intra-Packet Header

- Intra-Packet Time Stamp. These 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 RTC 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and to the first data bit in the message.

- Intra-Packet Data Header. The IPDH is an identification word (message ID word) that precedes the message and is inserted into the packet with the format shown in Figure 10-50.

MSB 31	30	29	16	15	LSB 0
DE	FE	SUBCHANNEL		MESSAGE LENGTH	

Figure 10-50. 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. Bits 29-16 contain 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 contain 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 64 KB.

10.6.10 Video Packets.

10.6.10.1 Video Packets, Format 0 (Moving Picture Experts Group-2/H.264).

Format 0 Moving Picture Experts Group (MPEG)-2/H.264 encoding will be IAW Department of Defense Motion Imagery Standards Profile (MISP) Standard 9601.[10] The MPEG-

[10] Motion Imagery Standards Board. "Standard Definition Digital Motion Imagery, Compression Systems." STD 9601 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

2/H.264 format will be transport streams (TS) per MISP Recommended Practice (RP) 0101.1.[11] 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 Recommended Practice 9720d[12] 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. The H.264 can be carried over the MPEG-2 TSs using International Telecommunications Union/Telecommunication Standardization Sector (ITU-T) Recommendation H.222.0[13] for MPEG2 TS containment for MPEG4 AVC. The MISP has adapted this with 9720d and 9701.

The TSs are limited to a single program stream (PS) using program elementary stream (PES) packets that share the same common time base. A TS 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. The PSs 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 Table 10-37. 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.

Table 10-37. General MPEG-2/H.264 Video Packet, Format 0
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

[11] Motion Imagery Standards Board. "Use of MPEG-2 System Streams in Digital Motion Imagery Systems." RP 0101.1. 27 January 2011. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/rp/RP010101.pdf.

[12] Motion Imagery Standards Board. "Motion Imagery Systems Matrix, Standard Definition Motion Imagery." RP 9720d in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

[13] International Telecommunications Union Telecommunication Standardization Sector. Information technology - Generic coding of moving pictures and associated audio information: Systems. ITU-T Rec.H.222.0 (06/12). June 2012. May be superseded by update. Available to ITU-T members and other subscribers at http://www.itu.int/rec/T-REC-H.222.0/en.

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[14] for MPEG-2 and ISO/IEC 14496-3[15] 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 CSDW, formatted as shown in Figure 10-51.

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

Figure 10-51. 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[16] and Standard 9715[17]. Embedded time is used for the synchronization of core MPEG-2 data when extracted from the Chapter 10 domain (i.e., an export to MPEG-2 files).

- Intra-Packet Header. Bit 30 indicates if IPTSs 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

  The TSs contain their own embedded time base used to facilitate the decoding and presentation of video and/or audio data at the decoder. Within a PS, all streams are synchronized to a single time source referred to as the system clock reference (SCR). Within a TS, each embedded program contains its own PCR, requiring that each

[14] International Organization for Standardization/International Electrotechnical Commission. *Information technology--Generic coding of moving pictures and associated audio information -- Part 3, Audio.* ISO/IEC 13818-3:1998. Geneva: International Organization for Standardization, 1998.
[15] International Organization for Standardization/International Electrotechnical Commission. *Information Technology - Coding of Audio-Visual Objects - Part 3: Audio*. ISO/IEC 14496-3 ed4.0. May be superseded by update. Available for purchase at http://webstore.iec.ch/Webstore/webstore.nsf/ArtNum_PK/43306!opendocument&preview=1.
[16] Motion Imagery Standards Board. "Imbedded Time Reference for Motion Imagery Systems." STD 9708 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.
[17] Motion Imagery Standards Board. "Time Reference Synchronization." STD 9715 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

Format 0-encoded MPEG-2/H.264 TS 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 27-MHz 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:

$$\text{SCR} = \text{SCR\_base} * 300 + \text{SCR\_ext}$$

where:

$$\text{SCR\_base} = [(\text{system\_clock\_frequency} * t)\ /300]\ \text{MOD}\ 233$$

$$\text{SCR\_ext} = [(\text{system\_clock\_frequency} * t)\ /1]\ \text{MOD}\ 300$$

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

$$\text{10-MHz RTC time base} = \text{SCR} * 10/27 \text{ (rounded to nearest integer)}$$

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.

- Key-Length-Value. Bit 28 indicates if key-length-value (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,[18] Standard 9712,[19] Standard 9713[20], Recommended Practice 9717,[21] and Standard 0107.1.[22]

---

[18] Motion Imagery Standards Board. "Intelligence Motion Imagery Index, Geospatial Metadata." STD 9711 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

[19] Motion Imagery Standards Board. "Intelligence Motion Imagery Index, Content Description…". STD 9712 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

[20] Motion Imagery Standards Board. "Data Encoding Using Key-Length-Value." STD 9713 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

- Payload (PL). Bits 27-24 indicate the payload type within the MPEG-2 stream per MISP Xon2.[23]

  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.

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

  0 = Little-endian as referenced in Figure 10-52.
  1 = Big-endian as referenced in Figure 10-53.

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 10-52. 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 10-53. Format 0 MPEG-2/H.264 Video Frame Sync and Word Format, 16-Bit Big-Endian (Native) Aligned

- Reserved. Bits 22-0 are reserved.

c. Intra-Packet Header. If enabled, the IPH shall include a 64-bit IPTS, which is inserted immediately before the TS sync pattern. The length of the IPH is fixed at 8 bytes (64 bits) positioned contiguously, in Figure 10-54.

---

[21] Motion Imagery Standards Board. "Packing KLV Packets into MPEG-2 Systems Streams." RP 9717 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.
[22] Motion Imagery Standards Board. *Bit and Byte Order for Metadata in Motion Imagery Files and Streams*. STD 107.1. June 2011. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/standards/Standard010701.pdf.
[23] Motion Imagery Standards Board. "Xon2". Subsection D-1.2 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

MSB 31	LSB 0
Time (LSLW)	
Time (MSLW)	

Figure 10-54. Intra-Packet Header

- Intra-Packet Time Stamp. (8 bytes) indicate the time tag of the individual TS packets. First long word (LSLW) bits 31-0 and second long word (MSLW) bits 31-0 indicate the following values.

(1) The 48-bit RTC 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and the first bit of the TS.

d. Video Packet Data. A Format 0 packet shall contain an integral number of 188-byte (1504 bits) TS packets as illustrated in Figure 10-52 and Figure 10-53 depending on the byte alignment bit. The IPHs can be inserted in Format 0 video data packets. The 10-MHz 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 13818-1:2007[24] the TS format will be fixed-length 188-byte (1504 bits) frames containing an 8-bit sync pattern or "sync byte" (starting at bit 0 and ending at bit 7 of the TS format). The sync bytes value is 01000111 (0x47). The rest of the TS 187 data bytes will follow (Table 10-38).

Table 10-38. Format 0 MPEG-2/H.264 Video Data Packet (Example is 16-Bit 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	

[24] International Organization for Standardization/International Electrotechnical Commission. *Information technology -- Generic coding of moving pictures and associated audio information: Systems*. ISO/IEC 13818-1:2007. October 2007. May be superseded by update. Available for purchase at http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=44169.

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

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 ISO/IEC 13818-1:2007 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 ISO/IEC 13818-1:2007 may be utilized in the encoding process. The TSs are limited to a single PS using PES packets that share the same common time base. A TS must contain the PAT and PMT that define the PID for the PCR stream. The PSs also must contain at least one pack header.

a. MPEG-2 Stream Packet Body. The Format 1 packet within MPEG-2 packets has the basic structure shown in Table 10-39. 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.

Table 10-39. General MPEG-2 Video Packet, Format 1

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

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 CSDW formatted as shown in Figure 10-55.

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

Figure 10-55. 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, Standard 9712, Standard 9713, Recommended Practice 9717, and Standard 0107.1.

- 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.

  The TSs contain their own embedded time base used to facilitate the decoding and presentation of video and/or audio data at the decoder. Within a PS, all streams are synchronized to a single SCR. Within a TS, each embedded program contains its own PCR, requiring that each Format 1 encoded MPEG-2 TS 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 indicates whether IPTSs 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 and Standard 9715. Embedded time is used for the synchronization of core MPEG-2 data when extracted from the 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 CBR parameter setting.

  0 = CBR 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 TS packets within the Format 1 packet. If TYPE=1, then this number represents the number of PS packs within the Format 1 packet.

d. Intra-Packet Header. If enabled, the IPH shall include a 64-bit IPTS, which is inserted immediately before the MPEG-2 packet (transport or program). The length of the IPH is fixed at 64 bits (8 bytes) positioned contiguously, in the following sequence (Figure 10-56).

MSB 31	LSB 0
Time (LSLW)	
Time (MSLW)	

Figure 10-56. Intra-Packet Header

- Intra-Packet Time Stamp. These 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.
    - The 48-bit RTC 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
    - The absolute time, if enabled by bit 6 in the packet flags (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item 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 Part 10.[25] The carrier format for Format 2 AVC/H.264 will be ISO/IEC 13818-1:2007 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.222.0.

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 TSs/PSs, 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 TS using PES packets that share the same common time base. The TS or PS must contain the PAT and PMT that define the PID for the PCR stream. The PSs also must contain at least one pack header.

[25] International Organization for Standardization/International Electrotechnical Commission. *Information Technology - Coding of Audio-Visual Objects - Part 10: Advanced Video Coding*. ISO/IEC 14496-10:2012. April 2012. May be superseded by update. Available at http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html.

a. AVC/H.264 Stream Packet Body. The Format 2 packet within AVC/H.264 packets has the basic structure shown in Table 10-40. 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.

Table 10-40. General AVC/H.264 Video Packet, Format 2

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

b. Video Packet Audio. When recording video using Format 2 AVC/H.264, if audio is present it will be inserted into the TS per ISO/IEC 13818-3 or 13818-7.[26] A separate analog channel to specifically record audio will not be required as AVC/H.264 supports audio insertion into the AVC/H.264 TS. 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 CSDW formatted as shown in Figure 10-57.

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

Figure 10-57. AVC/H.264 Channel-Specific Data Word Format

- Reserved (R). Bits 31-27 are reserved for future use.
- AVC/H.264 Audio Encoding Type (AET). Bit 26 indicates the AVC/H.264 audio encoding type.

  0 = ISO/IEC 13818-3
  1 = ISO/IEC 13818-7

- 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

[26] International Organization for Standardization/International Electrotechnical Commission. *Information technology -- Generic coding of moving pictures and associated audio information -- Part 7: Advanced Audio Coding (AAC).* ISO/IEC 13818-7:2006(E). Geneva: International Organization for Standardization, 2006.

1111 = 5.1

- <u>KLV.</u> 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, Standard 9712, Standard 9713 Recommended Practice 9717, and Standard 0107.1.

- <u>SCR/RTC Sync (SRS).</u> 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.

  The TSs contain their own embedded time base used to facilitate the decoding and presentation of video and/or audio data at the decoder. Within a PS, all streams are synchronized to a single SCR. Within a TS, each embedded program contains its own PCR, requiring that each Format 0-encoded MPEG-2 TS 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:

  $$\text{SCR} = \text{SCR\_base} * 300 + \text{SCR\_ext}$$

  where:

  $$\text{SCR\_base} = [(\text{system\_clock\_frequency} * t)/300] \text{ MOD } 233$$

  $$\text{SCR\_ext} = [(\text{system\_clock\_frequency} * t)/1] \text{ MOD } 300$$

  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 indicates whether IPTSs 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 and Standard 9715. Embedded time is used for the synchronization of core AVC/H.264 data when extracted from the 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 CBR parameter setting.

  0 = CBR 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 TS packets within the Format 2 packet. If TYPE=1, then this number represents of the number of PS packets within the Format 2 packet.

d. Intra-Packet Header. If enabled, the IPH shall include a 64-bit IPTS, which is inserted immediately before the AVC/H.264 packet (transport or program). The length of the IPH is fixed at 8 bytes (64 bits) positioned contiguously, in the following sequence (Figure 10-58).

MSB 31	LSB 0
Time (LSLW)	
Time (MSLW)	

Figure 10-58. Intra-Packet Header

- Intra-Packet Time Stamp. These 8 bytes indicate 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.
    - o The 48-bit RTC 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
    - o The absolute time, if enabled by bit 6 in the packet flags (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item 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 (Table 10-41) shall contain one or more fixed-length segments of one or more video images. The CSDW 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 IPH is not included with each segment, the RTC in the packet header is the time of the first segment in the packet.

Table 10-41. Image Packet, Format 0

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	

a. Image Packet Channel-Specific Data Word. The packet body portion of each image packet begins with a CSDW. 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 IPDH precedes each segment (Figure 10-59).

MSB 31 30	29 28	27	26 LSB 0
PARTS	SUM	IPH	LENGTH

Figure 10-59. Image Packet Channel-Specific Data Word Format

- Parts. Bits 31-30 indicate 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. Bits 29-28 indicate 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 IPH (time stamp) precedes each segment of the image.

  0 = The IPH not enabled
  1 = The IPH enabled

- Length. Bits 26-0 indicate 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 IPH as indicated by the IPH bit in the CSDW. When included, the IPH consists of an IPTS only. The length of the IPH is fixed at 8 bytes (64 bits) positioned contiguously, in the following sequence (Figure 10-60).

MSB 31 LSB 0
Time (LSLW)
Time (MSLW)

Figure 10-60. Image Data Intra-Packet Header, Format 0

- Intra-Packet Time Stamp. These 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.

  - The 48-bit RTC 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item 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 (Table 10-42) 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 CSDW 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.

Table 10-42. Still Imagery Packet, Format 1

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	

a. Still Imagery Packet Channel-Specific Data Word. The packet body portion of each still image packet begins with a CSDW. 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 10-61).

MSB 31 30	29 28	27	26 23	22 LSB 0
PARTS	SUM	IPH	FORMAT	RESERVED

Figure 10-61. Still Imagery Packet Channel-Specific Data Word Format

- Parts. Bits 31-30 indicate 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. Bits 29-28 indicate 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 IPH (time stamp) precedes each segment of the image.

  0 = The IPH not enabled
  1 = The IPH enabled

- Format. Bits 26-23 indicate a binary value that represents the still image format.

  0000 = MIL-STD-2500[27] National Imagery Transmission Format
  0001 = JPEG File Interchange Format
  0010 = JPEG 2000 (ISO/IEC 15444-1)[28]
  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 IPH. The IPH consists of an IPTS and intra-packet data. The length of the IPH is fixed at 12 bytes (96 bits) positioned contiguously, in the following sequence (Figure 10-62).

[27] Department of Defense. "National Imagery Transmission Format Version 2.1." MIL-STD-2500C. May 2006. May be superseded by update. Available at http://quicksearch.dla.mil/basic_profile.cfm?ident_number=112606&method=basic.

[28] International Organization for Standardization/International Electrotechnical Commission. *Information Technology -- JPEG 2000 Image Coding System: Core Coding System.* ISO/IEC 15444-1:2004. September 2004. May be superseded by update. Available for purchase at http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=37674.

MSB 31	LSB 0
Time (LSLW)	
Time (MSLW)	
Intra-Packet Data	

Figure 10-62. Still Imagery Intra-Packet Header

- Intra-Packet Time Stamp. These 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.
  - o The 48-bit RTC 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
  - o The absolute time, if enabled by bit 6 in the packet flags (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and the first data bit in the still image or segment.
- Intra-Packet Data. These 4 bytes indicate a binary value that represents the byte length of following still image or segment.

10.6.11.3 Image Packets, Format 2 (Dynamic Imagery).

A Format 2 image packet (Table 10-43) shall contain one or more fixed-length segments of a partial dynamic image, one complete dynamic image, or multiple complete dynamic images. Typically dynamic image packets will be created from cameras attached to a recorder or cameras that include a recording capability.

Table 10-43. Dynamic Imagery Packet, Format 1

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)	
Image Byte 2	Image Byte 1
:	:
Filler (if n is Odd)	Image 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)	
Image Byte 2	Image Byte 1
:	:
Filler (if n is Odd)	Image Byte N
Packet Trailer	

Each source of dynamic imagery (camera or sensor) shall have its own individual channel ID value. Multiple sources of dynamic imagery (camera or sensor) shall not share the same channel ID value. Dynamic Imagery, Format 2 is defined as image data that has a rate as opposed to Format 1 still imagery, which does not.

Dynamic image information will be specified in the CSDW and in the Computer-Generated Data, Format 1 setup record for each dynamic imagery channel. Only one dynamic imagery format can be defined for each Format 2 image packet channel ID.

If changes are made to the initial dynamic imagery channel settings in the Computer-Generated Data, Format 1 setup record a new setup record packet shall be created prior to any Format 2 image packets to which the new settings are applied. These changes shall be noted as a setup record configuration change in the Computer-Generated Data, Format 1 setup record CSDW.

a. Dynamic Imagery Packet Channel-Specific Data Word. The packet body portion of each dynamic image packet begins with a CSDW. It defines the format of the dynamic imagery format (which will coincide with the dynamic imagery format with the setup record) and indicates if the packet body contains several complete images or partial images (Figure 10-63).

MSB 31 30	29 28	27	26 21	LSB 20 0
PARTS	SUM	IPH	FORMAT	RESERVED

Figure 10-63. Dynamic Imagery Packet Channel-Specific Data Word Format

- Parts. Bits 31-30 indicate which segment of the image is contained in the packet if the packet does not contain one or more complete images.

  00 = Packet does not contain first or last segment of image
  01 = Packet contains first segment of image
  10 = Packet contains last segment of image
  11 = Reserved

- Sum. Bits 29-28 indicate if the packet contains a partial image that spans multiple packets, one complete image, or multiple complete images.

  00 = Packet contains less than one complete image (a segment)
  01 = Packet contains one complete image
  10 = Packet contains multiple complete images
  11 = Reserved

- Intra-Packet Header (IPH). Bit 27 indicates that the IPH (time stamp/data) shall precede each complete image within a packet or the first segment of a multi-segment image. The time stamp applied to each complete image or first segment of an image is dependent on the time stamp mode as defined in Section 10.6.11.3 item b. An IPH (time stamp) is not required for an image segment if it is not the first segment of a image.

  0 = The IPH is not enabled
  1 = The IPH is enabled

- Format. Bits 26-21 indicate a binary value that represents the dynamic image pixel format IAW GenICam Standard Features Naming Convention v1.5[29] or later and GigE Vision v1.2[30] or later.

  0x00 = Mono8
  0x01 = Mono8Signed
  0x02 = Mono10
  0x03 = Mono10Packed
  0x04 = Mono12
  0x05 = Mono12Packed
  0x06 = Mono14
  0x07 = Mono16
  0x08 = BayerGR8
  0x09 = BayerRG8
  0x0A = BayerGB8
  0x0B = BayerBG8
  0x0C = BayerGR10
  0x0D = BayerRG10
  0x0E = BayerGB10
  0x0F = BayerBG10
  0x10 = BayerGR12
  0x11 = BayerRG12
  0x12 = BayerGB12
  0x13 = BayerBG12
  0x14 = BayerGR10Packed
  0x15 = BayerRG10Packed
  0x16 = BayerGB10Packed
  0x17 = BayerBG10Packed
  0x18 = BayerGR12Packed
  0x19 = BayerRG12Packed
  0x1A = BayerGB12Packed
  0x1B = BayerBG12Packed
  0x1C = BayerGR16

---

[29] European Machine Vision Association. *GenICam Standard Features Naming Convention*. Version 1.5. November 2011. Available at http://www.emva.org/cms/upload/Standards/GenICam_Downloads/GenICam_SFNC_1_5.pdf.
[30] Automated Imaging Association. *GiGE Vision*. Version 1.2. n.d. Available for download with registration at http://www.visiononline.org/form.cfm?form_id=735.

0x1D = BayerRG16
0x1E = BayerGB16
0x1F = BayerBG16
0x20 = RGB8Packed
0x21 = BGR8Packed
0x22 = RGBA8Packed
0x23 = BGRA8Packed
0x24 = RGB10Packed
0x25 = BGR10Packed
0x26 = RGB12Packed
0x27 = BGR12Packed
0x28 = RGB16Packed
0x29 = BGR16Packed
0x2A = RGB10V1Packed
0x2B = BGR10V1Packed
0x2C = RGB10V2Packed
0x2D = BGR10V2Packed
0x2E = RGB12V1Packed
0x2F = RGB565Packed
0x30 = BGR565Packed
0x31 = YUV411Packed
0x32 = YUV422Packed
0x33 = YUV444Packed
0x34 = YUYVPacked
0x35 = RGB8Planar
0x36 = RGB10Planar
0x37 = RGB12Planar
0x38 = RGB16Planar
0x39-0x3E = Reserved
0x3F = Device-specific

- Reserved. Bits 20-0 are reserved.

b. Dynamic Imagery Intra-Packet Header. After the CSDW, the Format 2 dynamic imagery data (complete image, multiple complete images, or image segment) is inserted into the packet. The image shall be preceded by an IPH; this IPH shall provide the complete image or first image segment time stamp and the image length. The IPH time stamp value indicates the time of the complete image at sensor/camera capture.

The image time stamp characteristics are further defined within the setup record dynamic imagery packet channel attributes. Due to the fact that dynamic imagery may be captured and then packetized post-capture there maybe uniqueness in regards to time stamping of the data versus packet header/secondary header values related to the first bit of data within the packet as defined in sections 10.6.1.1 item i and 10.6.1.2 item a. Individual image IPH time stamp modes are defined as follows.

(1) Image Capture Time. The IPH TIME value corresponds to the RTC or absolute time when the image was captured by the sensor/camera. The packet header RTC/packet secondary header values indicate when the first bit of data is placed

into the packet. When Image Capture Time mode is indicated in the setup record it is understood there is a delay period between packet header RTC/secondary header time and IPH time.

(2) Image Packetization Time. The IPH TIME value corresponds to the RTC or absolute time when the image was packetized. The packet header RTC/secondary header values indicate when the first bit of data is placed into the packet. Image packetization time is defined as packetizing image data as it is captured by the sensor/camera. When Image Packetization Time mode is indicated in the setup record it is understood there is not a delay period between packet header RTC/secondary header time and the image IPH time.

The IPH consists of an IPTS and intra-packet data. The length of the IPH is fixed at 12 bytes (96 bits) positioned contiguously, in the following sequence (Figure 10-64).

MSB 31	LSB 0
Time (LSLW)	
Time (MSLW)	
Image Length	

Figure 10-64. Dynamic Imagery Intra-Packet Header

- Intra-Packet Time Stamp (TIME). These 8 bytes indicate the time tag of the Format 2 dynamic imagery data as defined in Section 10.6.11.3 item b. First long word bits 31-0 and second long word bits 31-0 indicate the following values.
  - The 48-bit RTC that corresponds to the first data bit in the dynamic image 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and the first data bit in the dynamic image.
- Intra-Packet Data (IMAGE LENGTH). These 4 bytes indicate a binary value that represents the byte length of following complete image.

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 Table 10-44.

Table 10-44. UART Data Packet Format

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	

a. UART Packet Channel-Specific Data Word. The packet body portion of each UART data packet begins with a CSDW as shown in Figure 10-65.

MSB 31	 30	LSB 0
IPH	RESERVED	

Figure 10-65. UART Packet Channel-Specific Data Word Format

- Intra-Packet Header. Bit 31 indicates whether the IPH time stamp is inserted before the UART ID words.

  0 = The IPH time stamp not enabled
  1 = The IPH 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 IPH with optional IPTS and a mandatory UART ID word IPDH. The length of the IPH is either 4 bytes (32 bits) or 12 bytes (96 bits) positioned contiguously, in the following sequence (Figure 10-66).

MSB 31	LSB 0
Time (LSLW)	
Time (MSLW)	
UART ID Word	

Figure 10-66. UART Data Intra-Packet Header

- UART Intra-Packet Time Stamp. These 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 RTC 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and the first data bit in the message.
- UART Intra-Packet Data Header. The IPDH is a UART ID word that precedes the data and is inserted into the packet with the following format. Inclusion of the IPDH is mandatory and is not controlled by the IPH bit in the CSDW (Figure 10-67).

MSB 31	30	29 16	15 LSB 0
PE	RESERVED	SUBCHANNEL	DATA LENGTH

Figure 10-67. 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 indicate 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 IPDH 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 1394-2008.[31] The 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 element size limits prescribed in Subsection 10.6.1. The IEEE 1394 packets have the basic structure shown in Table 10-45. Note that the width of the structure is not related to any number of bits. The table merely represents relative placement of data within the packet.

Table 10-45. IEEE 1394 Data Packet, Format 0
Packet Header

[31] Institute of Electrical and Electronics Engineers. *IEEE Standard for a High-Performance Serial Bus.* IEEE 1394-2008. New York: Institute of Electrical and Electronics Engineers, 2008.

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

a. IEEE 1394 Channel-Specific Data Word. The packet body portion (Figure 10-68) of each IEEE 1394 packet shall begin with a CSDW.

MSB 31 29	28 25	24 16	15 LSB 0
PBT	SY	RESERVED	TC

Figure 10-68. 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 - 111 = 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 IPH shall contain an 8-byte IPTS only. The format of an IEEE 1394 IPH is described by Figure 10-69.

MSB 31	LSB 0
Intra-Packet Time Stamp	
Intra-Packet Time Stamp	

Figure 10-69. IEEE 1394 Intra-Packet Header

- IEEE 1394 Intra-Packet Time Stamp. These 8 bytes indicate the time tag of the IEEE 1394 transaction that immediately follows it in the packet. Time is coded IAW 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 48-bit RTC that corresponds to the first data bit of the transaction, with bits 31-16 in the second long word zero-filled; or
- o The absolute time, if enabled by bit 6 in the packet flags (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item 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 CSDW as described by Subsection 10.6.13.1 item a above. The packet body type is indicated within the CSDW. Depending on the packet body type, the CSDW is followed by 0 or more transactions. Also, dependent on packet body type, each transaction may be preceded by an IPH.

- IEEE 1394 Packet Body Type 0: Bus Status. Type 0 packet bodies shall contain zero IPHs and zero transactions. The CSDW transaction count shall be zero. The packet body shall contain the CSDW 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 10-70.

MSB 31	LSB 30 0
RE	RESERVED

Figure 10-70. IEEE 1394 Event Data Word Format

- o RESET (RE). Bit 31, when set, indicates that an IEEE 1394 bus reset has occurred.
- o RESERVED. Bits 30-0 are 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 IPHs and one transaction. The CSDW transaction count shall be one.

  The CSDW 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. The number of data bytes shall be exactly four less than the value indicated in data length IAW the definition of packet header data length and accounting for the size of the CSDW. 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 CSDW, 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 CSDW shall be followed by a non-zero number of completed transactions as indicated by the CSDW transaction count. Each transaction shall be preceded by an IPH as defined above for IEEE 1394 data packets. Immediately following the IPH, the transaction shall be recorded in its entirety and must be a properly formed IEEE 1394 packet IAW 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. The IEEE 1394 data is packetized in Format 1 packets as physical layer data transfers (IAW Annex J of Standard 1394-1995[32] and Chapter 17 of Standard 1394b-2002[33]). A packet may contain 0 to 65,536 transfers, but is constrained by the common packet element size limits prescribed in Subsection 10.6.1. The IEEE 1394 packets have the basic structure shown in Table 10-46 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.

Table 10-46. IEEE 1394 Data Packet, Format 1

Packet Header
Channel-Specific Data Word
Intra-Packet Header
Data
(Optional) Intra-Packet Header
(Optional) Data
:
(Optional) Intra-Packet Header
(Optional) Data
Packet Trailer

a. IEEE 1394 Channel-Specific Data Word. The packet body portion (Figure 10-71) of each IEEE 1394 packet shall begin with a CSDW.

[32] Institute of Electrical and Electronics Engineers. *IEEE Standard for a High Performance Serial Bus*. IEEE 1394-1995. New York: Institute of Electrical and Electronics Engineers, 1995.
[33] Institute of Electrical and Electronics Engineers. *IEEE Standard for a High Performance Serial Bus: Amendment 2*. IEEE 1394b-2002. New York: Institute of Electrical and Electronics Engineers, 2002.

MSB 31	16	15	LSB 0
RESERVED		IPC	

Figure 10-71. 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 CSDW is followed by 1 or more IEEE 1394 transfers. Each transfer starts with an IPH, followed by 0-32,780 encapsulated data bytes. The length of the IPH is fixed at 12 bytes (96 bits) positioned contiguously, in the following sequence as shown in Figure 10-72.

MSB 31	LSB 0
Intra-Packet Time Stamp	
Intra-Packet Time Stamp	
Intra-Packet ID Word	

Figure 10-72. IEEE 1394 Format 1 Intra-Packet Header

- IEEE 1394 Format 1 Intra-Packet Time Stamp. These 8 bytes indicate the time tag of the IEEE 1394 transfer that immediately follows it in the packet. Time is coded IAW 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 48-bit RTC that corresponds to the first data byte of the transfer, with bits 15-0 in the second long word zero-filled; or
    - o The absolute time, if enabled by bit 6 in the packet flags (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and the first data byte of the transfer.
- Message ID Word. These 4 bytes are an ID word that precedes the message and is inserted into the packet as in Figure 10-73.

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

Figure 10-73. Intra-Packet Data Header - Message ID Word

- o Status Byte. Bits 31-24 define the status byte received from the physical layer devices IAW IEEE 1394b specification.
- o Transmission Speed (SPEED). Bits 23-20 identify 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). Bits 19-18 indicate if a transfer synchronization error occurred.

  00 = IEEE 1394 flow did not exceed maximum intra-packet size
  01 = This IEEE 1394 transfer started correctly but longer than the standard transfer length.
  10 = The previous IEEE 1394 transfer was in 01-type overflow and this IEEE 1394 transfer ended correctly (did not exceed standard transfer length).
  11 = The previous IEEE 1394 transfer was in 01-type overflow and this IEEE 1394 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.

- Local Buffer Overflow (LBO). Bit 17, if set, indicates that some messages are lost before this transfer due to local buffer overflow.

- Reserved (RSV). 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 32,780 for transfers corresponding to isochronous packets.

  If the data length field is not a multiple of 4 bytes, 1-3 fill bytes of 0x00 are 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 Digital Cartridge Recording System (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 CSDW. The data recorded from a parallel interface shall be placed into a Parallel Data Packet, Format 0 as shown in Table 10-47.

Table 10-47. Parallel Data Packet, Format 0

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

a. Parallel Packet Channel-Specific Data Word. The packet body portion of each parallel data packet begins with a CSDW. The CSDW is formatted as shown in Figure 10-74.

MSB 31 … 24	23 … LSB 0
TYPE	RESERVED (0) OR SCAN NUMBER

Figure 10-74. Parallel Packet Channel-Specific Data Word Format

- Type. Bits 31-24 indicate the data type stored

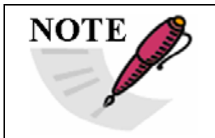
  0x01 - 0x00: Reserved
  0x80 - 0x10: Number of bits of recorded data (parallel data word width in bits)
  0xFD - 0x81: Reserved
  0xFE: DCRsi scan format, contains auxiliary data, DCRsi main data
  0xFF: Reserved

- Scan Number. Bits 23-0 are reserved (0) for general-purpose parallel data packets or contain 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 10-75).

MSB 15			LSB 0
Pad	Data Word 2	Pad	Data Word 1
	:		:
Pad	Data Word n, or Pad if Length is Odd	Pad	Data Word N-1

Figure 10-75. Parallel Data, Up to 8-Bit-Wide Words

	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 8 or less, the word shall be padded with zeros to 8-bit bytes.
- If the number of data bits is between 9 and 16, the words shall be padded with zeros to one 16-bit word, as in Figure 10-76.

MSB 15	LSB 0
Pad	Data Word 1
	:
Pad	Data Word N

Figure 10-76. Parallel Data, 9-16 -Bit-Wide Words

- If the number of data bits is greater than 16 the words shall be padded with zeros to multiples of 16-bit data words. Figure 10-77 shows storing of 28-bit data words.

MSB 15	LSB 0
Data Word 1, LSBs 15-0	
Pad	Data Word 1, MSBs 27-16
	:
Data Word N, LSBs 15-0	
Pad	Data Word N, MSBs 27-16

Figure 10-77. 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 9 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 10-78.

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 10-78. DCRsi Scan, 9-Auxiliary Data Byte + 4326 Bytes

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

DCRsi data without auxiliary data bytes can be stored also as 8-bit general-purpose parallel data as described in Subsection 10.6.15.1 item b.

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 Table 10-48.

Table 10-48. Ethernet Data Packet, Format 1

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	

a. Ethernet Data Packet Format 0, Channel-Specific Data Word. The packet body portion of each Ethernet data packet begins with a CSDW. It indicates how many physical Ethernet messages (media access control [MAC] frame) are placed in the packet body. The CSDW is formatted for the complete type of packet body as shown in Figure 10-79.

MSB 31 28	27 16	LSB 15 0
FORMAT	RESERVED	NUMBER OF FRAMES

Figure 10-79. 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 IPH that has both an IPTS and an IPDH containing a frame ID word. The length of the IPH is fixed at 12 bytes (96 bits) positioned contiguously, in the following sequence as shown in Figure 10-80.

MSB 31 LSB 0
Time (LSLW)
Time (MSLW)
Frame ID Word

Figure 10-80. Ethernet Data Format 1 Intra-Packet Header

- Intra-Packet Time Stamp. These 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.
  - o The 48-bit RTC 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item 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 10-81.

MSB							LSB
31	30	29 28	27 24	23 16	15	14	13 0
FCE	FE	CONTENT	SPEED	NET ID	DCE	LE	DATA LENGTH

Figure 10-81. Intra-Packet Frame ID Word

  - Frame CRC Error (FCE). Bit 31, the frame CRC error bit, is used to indicate that a MAC frame CRC error occurred during frame transmission.

    0 = No frame CRC error
    1 = Frame CRC error encountered.

  - 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 specify 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 indicate the negotiated bit rate for the identified NETID on which the frame was captured.

    0000 = Auto
    0001 = 10 megabits per second (Mbps)
    0010 = 100 Mbps
    0011 = 1 gigabit per second (Gbps)
    0100 = 10 Gbps
    0101 - 1111 = Reserved

  - Network Identifier (NETID). Bits 23-16 contain a binary value that represents the physical network identification of frame origination that follows the ID word. Zero means first and/or only physical network.

  - Data CRC Error (DCE). Bit 15, the data CRC error bit, is used to indicate that a CRC error exists in the payload of the frame.

    0 = No data CRC error
    1 = Data CRC error encountered

  - Data Length Error (LE). Bit 14, the data length error bit, is used to indicate that the frame is too short (less than 64 bytes) or too long (longer than 1518 bytes).

0 = Valid length
1 = Data length ID too short or too long.

- Data Length. Bits 13-0 contain a binary value that represents the length of the frame in bytes (n) that follows the ID word.

10.6.15.2 Ethernet Data Packets, Format 1, ARINC-664.

Any UDP packets from Ethernet and/or ARINC-664 network interfaces can be placed into an Ethernet Data Packet Format 1 as shown in Table 10-49.

Table 10-49. Ethernet Data Format 1

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)	
Intra-Packet Data Header for Msg 1 (Bits 63-48)	
Intra-Packet Data Header for Msg 1 (Bits 79-64)	
Intra-Packet Data Header for Msg 1 (Bits 95-80)	
Intra-Packet Data Header for Msg 1 (Bits 111-96)	
Intra-Packet Data Header for Msg 1 (Bits 127-112)	
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 127-112)	
Byte 2	Byte 1
:	:
Filler (if n is Odd)	Byte N
Packet Trailer	

The ARINC-664 part 7 was based on IEEE Standard 802.3-2012[34] or Ethernet specification. Unlike the Ethernet frame, the last byte of frame payload is used for frame sequence number. This byte is located just before the MAC CRC field, as part of the MAC payload. Ethernet Data Packets, Format 1 ARINC-664 Part 7 shall capture and store the entire ARINC-664 message (the entire UDP payload), including one or more fragmented frames.

The ARINC-664 frame sequence numbers are used by the end system for integrity checking and redundancy management. ARINC-664 requires two redundant switch networks. Each ARINC-664 frame is replicated and sent on both networks. The ARINC-664 receiving end system uses the sequence number to check for dropped frames and to eliminate redundant frames. The link layer of the receiver's ARINC-664 interface discards the sequence number and drops the redundant frame before passing the frame's payload to the IP network layer of the protocol stack. If a UDP datagram is fragmented, the sequence numbers on the fragments are discarded prior to datagram reassembly. Table 10-50 compares a normal Ethernet frame with an ARINC-664 frame.

<table>
<tr><th colspan="8">Table 10-50. Comparison of Normal Ethernet and ARINC-644 Frames</th></tr>
<tr><th colspan="8">Normal Ethernet Frame</th></tr>
<tr><td>7 bytes</td><td>1 byte</td><td>14 bytes</td><td>20 bytes</td><td>8 bytes</td><td><=1472 bytes</td><td colspan="2">4 bytes</td></tr>
<tr><td>Preamble</td><td>Start Delimiter</td><td>MAC Header</td><td>IP Header</td><td>UDP Header</td><td>UDP Payload</td><td colspan="2">FCS</td></tr>
<tr><th colspan="8">ARINC-664 Frame</th></tr>
<tr><td>7 bytes</td><td>1 byte</td><td>14 bytes</td><td>20 bytes</td><td>8 bytes</td><td><=14,721 bytes</td><td>1 byte</td><td>4 bytes</td></tr>
<tr><td>Preamble</td><td>Start Delimiter</td><td>MAC Header</td><td>IP Header</td><td>UDP Header</td><td>ARINC-664 Payload</td><td>Sequence Number</td><td>FCS</td></tr>
</table>

a. Ethernet Data Format 1, Channel-Specific Data Word. The packet body portion of each Ethernet data packet begins with a CSDW. It indicates how many ARINC-664 messages are placed in the packet body. The CSDW is formatted for the complete type of packet body as shown in Figure 10-82.

MSB 31 28 27 16	15 LSB 0
Intra-Packet Header Length	Number of ARINC-664 Messages

Figure 10-82. Ethernet Data Packet Format 1 Channel-Specific Data Word

- Intra-Packet Header Length. Length of the IPH in bytes
- Number of Messages. Bits 15-0 contain a binary value that represents the number of messages included in the packet.

[34] Institute of Electrical and Electronics Engineers. *IEEE Standard for Ethernet*. IEEE 802.3-2012. New York: Institute of Electrical and Electronics Engineers.

b. Ethernet Data Packet Format 1 Intra-Packet Header. After the channel-specific data, ARINC-664 data is inserted into the packet. Each message is preceded by an IPH that has both an IPTS and an IPDH. The length of the IPH is fixed at 24 bytes (192 bits) positioned contiguously, in the following sequence as shown in Figure 10-83.

MSB 31	LSB 0
Time (LSLW)	
Time (MSLW)	
Intra-Packet Data Header 1	
Intra-Packet Data Header 2	
Intra-Packet Data Header 3	
Intra-Packet Data Header 4	
Intra-Packet Data Header 5	

Figure 10-83. Ethernet Data Format 1 Intra-Packet Header

- Intra-Packet Time Stamp. These 8 bytes indicate the time tag of the ARINC-664 message. First long word bits 31-0 and second long word bits 31-0 indicate the following values.
  - The 48-bit RTC 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and the first data bit in the frame.
- Intra-Packet Data Header. These 20 bytes contain ARINC-664 message data length, virtual link, source and destination IP addresses, and source and destination UDP ports, as shown in Figure 10-84.

MSB 31 16	15 8	LSB 7 0
Data Length	Error bits	Flags bits
Reserved	Virtual Link	
Source IP		
Dest IP		
Src Port	Dst Port	

Figure 10-84. Intra-Packet Data Header

  - Data Length (bits 31-16)
    - Message length in bytes
  - ERROR Bits (bits 15-8)
    - 0: No errors
    - 1: Any undefined error
    - 2-15: Reserved

- Flags (bits 7-0)
  - 0: Actual ARINC-664 data
  - 1: Simulation ARINC-664 data
  - 2-15: Reserved
- Reserved (bits 31-16)
- Virtual Link (VL) (bits 15-0)
  - Lower 16 bits of the Ethernet destination MAC address
- Source IP address (bits 31-0)
  - Source IP address from ARINC-664 IP header
- Dest IP Address (bits 31-0)
  - Destination IP address from ARINC-664 IP header
- Src Port (bits 31-16)
  - 16 bits source port from the ARINC-664 UDP header
- Dest Port (bits 15-0)
  - Destination port from the ARINC-664 UDP header

10.6.16 Time Space Position Information and Combat Training Systems Data Packets.

The Time Space Position Information (TSPI) and Combat Training Systems (CTS) data type packets are provided to allow a defined method of TSPI/CTS data encapsulation in Chapter 10 packet format. This will provide interoperability of these data sets between ranges and users along with alignment to other digital data in the recording, such as video and audio.

The TSPI/CTS data packets do not require a specific input interface such as PCM, analog, or MIL-STD-1553. The TSPI/CTS data type will only encapsulate multiple types of TSPI/CTS information IAW governing standards and specifications. Essentially, TSPI/CTS data will be wrapped in its native format by Section 10.6 packet structures and reside on compliant media devices and/or within files. The packet definition will not characterize transmission protocols or requirements because those are provided by the governing standards or specifications.

The TSPI/CTS packets are considered dynamic and Chapter 10 timing requirements apply whether they are generated by the recorder/multiplexer, like computer-generated data packets, or generated via a specific external interface.

10.6.16.1 TSPI/CTS Data Packets, Format 0 (NMEA-RTCM).

Any GPS data as defined by the National Marine Electronics Association (NMEA) and Radio Technical Commission for Maritime Services (RTCM) standards will be encapsulated in the Format 0 packet. The NMEA and RTCM standards specify the electrical signal requirements, data transmission protocol, and message/sentence formats for GPS data. These formatting standards will not be detailed in Chapter 10, but they will be referenced as required for clarity.

The TSPI/CTS Data Packet, Format 0 (NMEA-RTCM) will not support proprietary messages or sentences; therefore, any data containing these will be considered non-compliant with this standard.

A packet with n NMEA-RTCM data has the basic structure as Table 10-51.

Table 10-51. NMEA-RTCM Data Packet Format

MSB 15	LSB 0
Packet Header	
Channel-Specific Data (Bits 15-0)	
Channel-Specific Data (Bits 31-16)	
(Optional) Intra-Packet Time Stamp for Data 1 (Bits 15-0)	
(Optional) Intra-Packet Time Stamp for Data 1 (Bits 31-16)	
(Optional) Intra-Packet Time Stamp for Data 1 (Bits 47-32)	
(Optional) Intra-Packet Time Stamp for Data 1 (Bits 63-48)	
Intra-Packet Data Header (Bits 15-0)	
Intra-Packet Data Header (Bits 31-16)	
Byte 2	Byte 1
:	:
Filler (if n is Odd)	Byte N
:	
(Optional) Intra-Packet Time Stamp for Data N (Bits 15-0)	
(Optional) Intra-Packet Time Stamp for Data N (Bits 31-16)	
(Optional) Intra-Packet Time Stamp for Data N (Bits 47-32)	
(Optional) Intra-Packet Time Stamp for Data N (Bits 63-48)	
Intra-Packet Data Header (Bits 15-0)	
Intra-Packet Data Header (Bits 31-16)	
Byte 2	Byte 1
:	:
Filler (if n is Odd)	Byte N
Packet Trailer	

a. NMEA-RTCM Packet Channel-Specific Data Word. The packet body portion of each NMEA-RTCM data packet begins with a CSDW as shown in Figure 10-85.

MSB 31	30 27	26 LSB 0
IPTS	TYPE	RESERVED

Figure 10-85. NMEA-RTCM Packet Channel-Specific Data Word Format

- IPTS. Bit 31 indicates whether the IPTS is enabled.

  0 = IPTS not enabled

1 = IPTS enabled

- TYPE. Bits 30-27 indicate the type of data NMEA-RTCM contains within the packet.

  0000 = NMEA 0183
  0001 = NMEA 0183-HS
  0010 = NMEA 2000
  0011 = RTCM SC104
  0010 - 1111 = Reserved

- RESERVED. Bits 26-0 are reserved and shall be zero-filled.

b. NMEA-RTCM Intra-Packet Time Stamp. If enabled the optional IPTS is inserted before each NMEA-RTCM message. The length of the IPTS is 8 bytes (64 bits) positioned contiguously, in the following sequence (Figure 10-86).

MSB 31	LSB 0
(Optional) Time (LSLW)	
(Optional) Time (MSLW)	

Figure 10-86. NMEA-RTCM Intra-Packet Time Stamp

- Intra-Packet Time Stamp. These 8 bytes indicate the time tag of the NMEA-RTCM data. First long word bits 31-0 and second long word bits 31-0 indicate the following values.

  - The 48-bit RTC that corresponds to the first data bit. 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 (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and the first data bit.

c. NMEA-RTCM Intra-Packet Data Header. The length of the IPDH is fixed at 4 bytes (32 bits) positioned contiguously, in the following sequence (Figure 10-87).

MSB 31 16	LSB 15 0
RESERVED	LENGTH

Figure 10-87. NMEA-RTCM Intra-Packet Data Header

- RESERVED. Bits 31-16 are reserved.

- LENGTH. Bits 15-0 indicate the length of the message in bytes.

10.6.16.2 TSPI Data Packets, Format 1 (EAG ACMI).

Air Combat Maneuvering Instrumentation (ACMI) data as defined by the European Air Group (EAG) interface control document (ICD) DF29125[35] for post-mission interoperability will be encapsulated in the Format 1 packet. The EAG ACMI ICD defines the data contents and organization. Electrical signal requirements and data transmission protocol are not defined in DF29125 or in this Chapter 10 format. The data type will be 8-bit ASCII. A packet of EAG ACMI data has the basic structure of Table 10-52.

Table 10-52. EAG ACMI Data Packet Format

MSB 15	LSB 0
Packet Header	
Channel-Specific Data (Bits 15-0)	
Channel-Specific Data (Bits 31-16)	
(Optional) Intra-Packet Time Stamp-Data Block 1 (Bits 15-0)	
(Optional) Intra-Packet Time Stamp-Data Block 1 (Bits 31-16)	
(Optional) Intra-Packet Time Stamp-Data Block 1 (Bits 47-32)	
(Optional) Intra-Packet Time Stamp-Data Block 1 (Bits 63-48)	
Intra-Packet Data Header	
(Optional) Static Data	
Byte 2	Byte 1
:	:
Filler (if n is Odd)	Byte N
Packet Trailer	

a. EAG ACMI Packet Channel-Specific Data Word. The packet body portion of each EAG ACMI data packet begins with a CSDW as shown in Figure 10-88.

MSB 31	30 29	28 LSB 0
IPTS	CONTENT	RESERVED

Figure 10-88. EAG ACMI Packet Channel-Specific Data Word Format

- IPTS. Bit 31 indicates whether the IPTS is enabled.

  0 = IPTS not enabled
  1 = IPTS enabled

- CONTENT. Bits 30-29 indicate the content of the EAG ACMI data within the packet.

  00 = TSPI data only (no static data or pod ID)
  01 = Contains pod ID and static data

[35] European Air Group. "European Air Group Interface Control Document for Post Mission Interoperability." DR29125 Draft A Issue 01. July 2004. Available to RCC members with Private Portal access at https://wsdm.wsmr.army.mil/site/rccpri/Publications/106-13_Part%20I%20-%20Telemetry%20Standards/DF29125.pdf.

10 - 11 = Reserved.

- RESERVED. Bits 28-0 are reserved.

b. EAG ACMI Intra-Packet Time Stamp. If enabled the optional IPTS is inserted before EAG ACMI data block. The length of the IPTS is 8 bytes (64 bits) positioned contiguously, in the following sequence (Figure 10-89).

MSB 31	LSB 0
(Optional) Time (LSLW)	
(Optional) Time (MSLW)	

Figure 10-89. EAG ACMI Data Intra-Packet Time Stamp

- EAG ACMI Intra-Packet Time Stamp. These 8 bytes indicate the time tag of the EAG ACMI TSPI data. Pod ID and static data will not be time-tagged but will proceed the TSPI data in the packet. First long word bits 31-0 and second long word bits 31-0 indicate the following values.
    - o The 48-bit RTC that corresponds to the first TSPI data bit. Bits 31 to 16 in the second long word (MSLW) of the IPTS will be zero-filled; or
    - o The absolute time, if enabled by bit 6 in the packet flags (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and the first data bit.

c. EAG ACMI Intra-Packet Data Header. The length of the IPDH is fixed at 4 bytes (32 bits) positioned contiguously, in the following sequence (Figure 10-90).

MSB 31 16	15 LSB 0
RESERVED	LENGTH

Figure 10-90. EAG ACMI Intra-Packet Data Header

- RESERVED. Bits 31-16 are reserved.
- LENGTH. Bits 15-0 indicate the length of the message in bytes.

10.6.16.3 TSPI Data Packets, Format 2 (ACTTS).

Air Combat Test and Training System (ACTTS) data as defined by the USAF ACTTS interface specification WMSP 98-01[36] will be encapsulated in the Format 2 packet. The ACTTS interface specification defines the unique signal interface requirements for the air-to-air, air-to-ground, ground-to-air data links, and aircraft information subsystem recording formats. The ACTTS WMSP 98-01 establishes the requirements for the information recorded on the different

[36] Range Instrumentation System Program Office, Air Armament Center. "Interface Specification for the USAF Air Combat Test and Training System (ACTTS) Air-to-Ground, Air-to-Air, Ground-to-Air Data Links, and AIS Recording Formats." WMSP 98-01, Rev A, Chg 1. 19 May 2003. Available to RCC members with Private Portal access at https://wsdm.wsmr.army.mil/site/rccpri/Publications/106-13_Part%20I%20-%20Telemetry%20Standards/WMSP_98-01.doc.

data transfer units used by the various ACTTS airborne subsystems to support both tethered and rangeless operations.

When encapsulating ACTTS message/word format, data messages or words will not span packets. Each new packet will start with a full message and not a partial message or word. A packet of ACCTS data has the basic structure of Table 10-53.

Table 10-53. ACCTS Data Packet Format

MSB 15	LSB 0
Packet Header	
Channel-Specific Data (Bits 15-0)	
Channel-Specific Data (Bits 31-16)	
(Optional) Intra-Packet Time Stamp for Data 1 (Bits 15-0)	
(Optional) Intra-Packet Time Stamp for Data 1 (Bits 31-16)	
(Optional) Intra-Packet Time Stamp for Data 1 (Bits 47-32)	
(Optional) Intra-Packet Time Stamp for Data 1 (Bits 63-48)	
Intra-Packet Data Header	
Byte 2	Byte 1
:	:
Filler (if n is Odd)	Byte N
:	
(Optional) Intra-Packet Time Stamp for Data N (Bits 15-0)	
(Optional) Intra-Packet Time Stamp for Data N (Bits 31-16)	
(Optional) Intra-Packet Time Stamp for Data N (Bits 47-32)	
(Optional) Intra-Packet Time Stamp for Data N (Bits 63-48)	
Intra-Packet Data Header	
Byte 2	Byte 1
:	:
Filler (if n is Odd)	Byte N
Packet Trailer	

a. ACTTS Packet Channel-Specific Data Word. The packet body portion of each ACTTS data packet begins with a CSDW as shown in Figure 10-91.

MSB 31	30 27	26 LSB 0
IPTS	FORMAT	RESERVED

Figure 10-91. ACCTS Packet Channel-Specific Data Word Format

- IPTS. Bit 31 indicates whether the IPTS is enabled.

  0 = IPTS not enabled
  1 = IPTS enabled

- FORMAT. Bits 30-27 indicate the ACTTS format type of data contained within the packet.

0000 = Kadena Interim Training System (KITS) Recording Formats
0001 = Alpena KITS Recording Formats
0010 = USAF Europe Rangeless Interim Training System Recording Formats
0011 = Alaska ACTS Upgrade Recording Formats
0100 = Goldwater Range Mission and Debriefing System Recording Formats
0101 = P4RC Recording Formats
0110 = Nellis ACTS Range Security Initiative Recording Formats
0111 = P4RC+P5 CTS Participant Subsystem Recording Formats
1000 = P5 CTS Recording Formats
1001 - 1111 = Reserved

- RESERVED. Bits 26-0 are reserved.

b. ACTTS Intra-Packet Time Stamp. If enabled the optional IPTS is inserted before each ACTTS message. The length of the IPH is 4 bytes (32-bits) positioned contiguously, in the following sequence (Figure 10-92).

MSB 31	LSB 0
(Optional) Time (LSLW)	
(Optional) Time (MSLW)	

Figure 10-92. ACCTS Data Intra-Packet Time Stamp

- ACTTS Intra-Packet Time Stamp. These 8 bytes indicate the time tag of the ACTTS data. First long word bits 31-0 and second long word bits 31-0 indicate the following values.
  - The 48-bit RTC that corresponds to the first ACTTS data bit. Bits 31 to 16 in the second long word (MSLW) of the IPTS will be zero-filled; or
  - The absolute time, if enabled by bit 6 in the packet flags (Subsection 10.6.1.1 item g). Time format is indicated by bits 2 and 3 in the packet flags (Subsection 10.6.1.1 item g) and the first data bit.

c. ACTTS Intra-Packet Data Header. The length of the IPDH is fixed at 4 bytes (32 bits) positioned contiguously, in the following sequence (Figure 10-93).

MSB 31 16	LSB 15 0
RESERVED	LENGTH

Figure 10-93. ACCTS Data Intra-Packet Data Header

- RESERVED. Bits 31-16 are reserved.
- LENGTH. Bits 15-0 indicate the length of the message in bytes.

10.6.17 Controller Area Network Bus

10.6.17.1 Controller Area Network Bus Data Packets

Data from one or more controller area network (CAN) bus interfaces are placed into a CAN bus data packet format as shown in Table 10-54.

Table 10-54. Controller Area Network Bus Data Packet Format

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 Message Header for Msg 1 (Bits 15-0)	
Intra-Packet Message Header for Msg 1 (Bits 31-16)	
Intra-Packet ID Word for Msg 1 (Bits 47-32)	
Intra-Packet ID Word for Msg 1 (Bits 63-48)	
Msg 1 Byte 2	Msg 1 Byte 1
:	:
Filler (if n is Odd)	Msg 1 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 Message Header for Msg n (Bits 15-0)	
Intra-Packet Message Header for Msg n (Bits 31-16)	
Intra-Packet ID Word for Msg n (Bits 47-32)	
Intra-Packet ID Word for Msg n (Bits 63-48)	
Msg n Byte 2	Msg n Byte 1
:	:
Filler (if m is Odd)	Msg n Byte m
Packet Trailer	

a. CAN Bus Packet Channel-Specific Data Word. The packet body portion of each CAN bus data packet begins with a CSDW. Figure 10-94 displays a complete CAN bus channel-specific data word.

MSB 31 16	15 LSB 0
RESERVED	N of Messages

Figure 10-94. Complete CAN Bus Channel-Specific Data Word Format

- Reserved. Bits 31-16 are reserved.
- N of Messages. Bits 15-0 contain a binary value indicating the number of messages included in the packet.

b. CAN Bus Data Intra-Packet Header. After the CSDW, CAN bus data is inserted into the packet. Each CAN bus message is preceded by an IPH that has both an IPTS and an intra-packet message header (IPMH) and an intra-packet ID word. The length of the IPH is fixed at 16 bytes (128 bits) positioned contiguously, in the sequence shown in Figure 10-95.

MSB 31	LSB 0
Intra-Packet Time Stamp (LSLW)	
Intra-Packet Time Stamp (MSLW)	
Intra-Packet Message Header	
Intra-Packet ID Word	

Figure 10-95. CAN Bus Data Intra-Packet Data Header

- Intra-Packet Time Stamp. These 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.
  - The RTC that corresponds to the first data bit in the message with bits 31 to 16 in the second long word zero-filled; or
  - Time, if enabled by bit 7 in the packet flags. Time format is indicated by bits 2 and 3 in the packet flags and to the first data bit in the message.
- Intra-Packet Message Header. The IPDH is an information word that is inserted into the packet with the format shown in Figure 10-96.

MSB 31	 30	 29 24	 23 16	 15 4	LSB 3 0
DE	FE	Reserved	SUBCHANNEL	Reserved	MESSAGE LENGTH

Figure 10-96. Intra-Packet Message Header

  - 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

  - Reserved. Bits 29-24 are reserved.
  - Subchannel. Bits 23-16 contain 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, which is valid for the CAN bus.
  - Reserved. Bits 15-4 are reserved.

- Message Length. Bits 3-0 contain a binary value representing the length of the number of the valid bytes in the rest of the message that follows the IPMH. The message length will be 4-12 bytes (4 bytes for the intra-packet ID word + 0-8 bytes data content of the CAN bus message).

- Intra-Packet ID Word. The intra-packet ID word contains the CAN bus message ID word transmitted on the bus. This word precedes the message and is inserted into the packet with the format shown in Figure 10-97.

MSB 31	30	29	28 LSB 0
IDE	RTR	Res	CAN Bus ID

Figure 10-97. Intra-Packet ID Word

- IDE (bit 31 of the 32-bit CAN ID word): use extended CAN identifier.

  0 = 11-bit standard CAN identifier (CAN ID word bits 10-0)
  1 = 29-bit extended CAN identifier (CAN ID word bits 28-0)

- RTR (bit 30 of the CAN ID word): Remote transfer request bit.

- CAN Bus ID: The CAN bus ID transmitted in the message.

  When using the 11-bit standard CAN identifier, bits 29-11 of the 32-bit CAN ID word are unused. For the 29-bit extended CAN identifier all the 29 bits, 28-0, are used.

- CAN Bus Message. The CAN bus message is placed behind the CAN bus IPH. The message can consist up to 8 bytes, which is placed in 0-4 x 16-bit data words. Figure 10-98 displays a CAN bus message format.

BYTE 2	BYTE 1
:	:
Filler (if n is Odd)	BYTE n

Figure 10-98. CAN Bus Message

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 Section 10.4. These interfaces shall support communications using SCSI (Fibre Channel) IAW Subsection 10.4.1, SCSI over SBP-2 (IEEE 1394b) IAW Subsection 10.4.2, or iSCSI (Ethernet) IAW Subsection 10.4.3. Recorder login and Chapter 6 CCM shall be transmitted and received using the SCSI ORB structures IAW subsections 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 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, summarized in Table 10-55, are a subset of the recorder CCM defined in Chapter 6, Subsection 6.2, 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. Commands will not be case-sensitive except for location parameter in .PLAY and file name in .RECORD.

Table 10-55. 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	[textstring]	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
.RCC-106		Returns supported version number of RCC-106 Recorder CCM
.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

Table 10-55. Command Summary		
Command	**Parameters***	**Description**
* 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).		

10.7.9 Required Discrete Control Functions.

Required discrete control functions are noted in Table 10-56.

Table 10-56. Required Discrete Control Functions
Description
RECORD
ERASE
DECLASSIFY
ENABLE
BIT

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 recorder's ground completes the circuit to activate a function as shown in Figure 10-99.

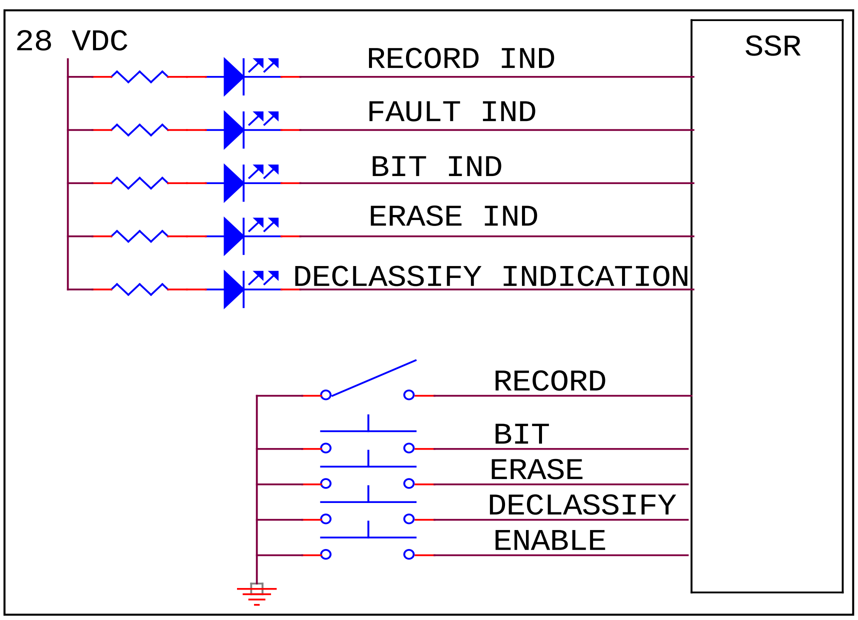


Figure 10-99. Discrete Control and Indicator Functional Diagram

b. Record Command. Activated by toggle switch (normally closed position 0.55 volts or less), this discrete control commands the recorder to start recording. The 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 control 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 control 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 control to operate.

f. Bit Command. Activated by momentary switch (0.55 volts or less), this discrete control commands the recorder to start the built-in test (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.

10.7.10 Voltage.

28 volts direct current (VDC) auxiliary voltage output shall be provided from the discrete/control port (250 milliamp max, short circuit protection).

10.7.11 Status Query.

Status querying shall be limited to intervals not to exceed 2 seconds and not less than one second.

10.7.12 Erase Command.

Activated by momentary switch (.55 volts or less, minimum duration of 100 ms, if ENABLE discrete is also activated for 100 ms), this discrete control commands the recorder to erase its user data and file directory memory provided the ENABLE switch is also activated.

10.7.13 Declassify Command.

Activated by momentary switch (0.55 volts or less, minimum duration of 100 ms, if ENABLE discrete is also activated for 100 ms) this discrete commands the recorder to start the declassify procedure provided the ENABLE switch is also activated.

10.7.14 Command Enable.

Activated by a momentary switch (0.55 volts or less, minimum duration of 100 ms) for either ERASE or DECLASSIFY discrete control 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 ms. Table 10-57 presents the recorder light-emitting diode states.

Table 10-57. Recorder Light-Emitting Diode 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 National Security Agency Manual 9-12,[37] DoD Directive 5200.28,[38] and DCID 6/3[39] 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 (SSDs) 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 Approach.

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

[37] National Security Agency. "NSA/CSS Storage Device Declassification Manual." Manual 9-12. n.d. May be superseded by update. Available at http://www.nsa.gov/ia/_files/government/MDG/NSA_CSS_Storage_Device_Declassification_Manual.pdf.
[38] Department of Defense. "Security Requirements for Automated Information Systems (AIs)." DoDD 5200.28. 21 March 1988. May be superseded by update. Available at http://csrc.nist.gov/groups/SMA/fasp/documents/c&a/DLABSP/d520028p.pdf
[39] Director of Central Intelligence. "Protecting Sensitive Compartmented Information Within Information Systems." DCID 6/3. May be superseded by update. Available at http://www.fas.org/irp/offdocs/DCID_6-3_20Policy.htm.

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 Algorithm.

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. First Erase. 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 Host Platform Interface to Recorder Media

Two interfaces, IEEE 1394b (sometimes known as "FireWire", a trademark of Apple Inc.) and IEEE 802.3 "Ethernet", are defined to provide a communication path to read and/or download data from an RMM and to write an RSCF to an RMM. The selection of these protocols was adopted to facilitate a common interface between the media and the computing platform. It is anticipated that any particular RMM will support only one of the two host platform interfaces.

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 the recorder, the removable media cartridges shall maintain time, allowing for time initialization of the recorder. Removable media cartridges shall allow for a battery back-up real-time clock device. Initialization of time shall be accomplished via the host platform interface.

10.9.2 Physical and Signaling.

Each host platform interface has distinct requirements for the physical interface and signaling levels.

10.9.2.1 IEEE 1394b Interface.

The IEEE 1394b host platform interface shall provide data communications and power using the same connector IAW IEEE 1394b.

10.9.2.2 Ethernet Interface.

The Ethernet host platform interface shall be IAW the IEEE 802.3 standards. Only a subset of the physical interfaces defined by IEEE 802.3 shall be employed. A power input accepting 8-30 VDC and drawing a current of not to exceed 5 amps shall be provided. Additionally, Power Over Ethernet (PoE) IAW IEEE 802.3at-2009[40] may be used to deliver power to the RMM.

a. 100Base-TX. For data rates of up to 100 Mbps, 100Base-TX signaling IAW IEEE 802.3 shall be employed.

b. 1000Base-T. For data rates in excess of 100 Mbps but less than 1000 Mbps, 1000Base-T with auto negotiation to lower speeds as defined in paragraph a above shall be employed IAW IEEE 802.3.

c. 10G-Base-T. For data rates in excess of 1000 Mbps, 10GBase-T with auto negotiation to lower speeds as defined in paragraph b above shall be employed IAW IEEE 802.3.

10.9.3 Removable Media Communication.

Logically, each compliant RMM shall contain two distinct functional entities as per Figure 10-100. The mechanisms used to communicate with the two functional entities vary according to the host platform interface type.

[40] Institute of Electrical and Electronics Engineers. *IEEE Standard for Information technology - Telecommunications and information exchange between systems…Amendment 3: Data Terminal Equipment (DTE) Power via the Media Dependent Interface (MDI) Enhancements*. IEEE 802.3at-2009. October 2009. May be superseded by update. Available with registration at http://standards.ieee.org/about/get/802/802.3.html.

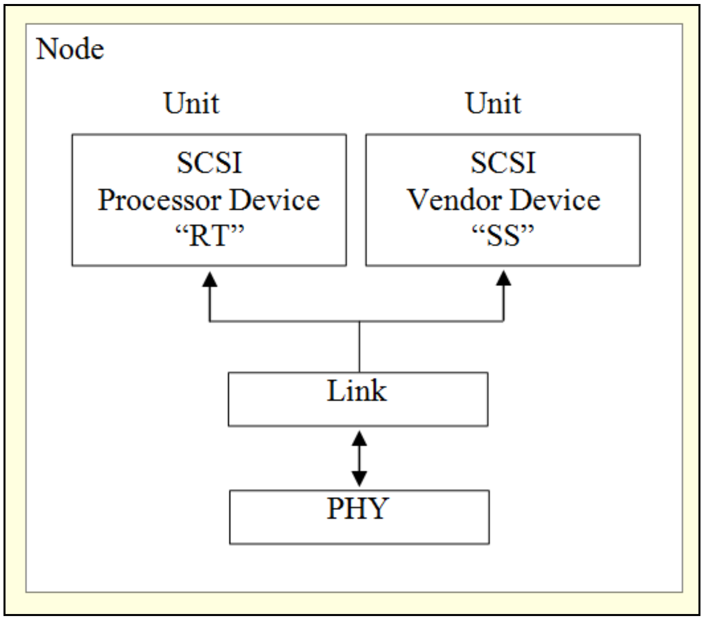


Figure 10-100. Removable Media

10.9.3.1 <u>IEEE 1394b Host Platform Interface.</u>

NEW

The fundamental method of communicating shall be IAW the IEEE 1394b protocol. Packets sent and received shall be asynchronous transmissions. The IEEE 1394b packets shall encapsulate 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 ORBs.

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

10.9.3.2 <u>IEEE 802.3 Ethernet Host Platform Interface.</u>

NEW

The fundamental method of communicating shall be IAW the IPv4 protocol defined by IETF RFC 791[41] and subsequent related documents.

a. <u>Frame size.</u> Following power on or reset, the RMM shall select a frame size or maximum transfer unit (MTU) of 1500 bytes.

b. <u>RMM IP Addressing.</u> Each RMM shall as a default use the static IP addressing using the IP address 10.9.3.2, a net mask of 255.0.0.0, and a default gateway of 10.9.3.1. The default static IP can be changed by sending a .RMMIP IPaddress command as defined in Subsection 10.9.15.2.

   An RMM can optionally also use DHCP to obtain an IP address IAW IETF RFC 2131.[42] The RMM shall send a DHCP vendor class identifier option (code 60) IAW IETF RFC

[41] Internet Engineering Task Force. "Internet Protocol." RFC 791. September 1981. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc791/.

2132[43] to the server, and the first 10 characters of the data string sent with the vendor class identifier option shall be the text “RMM:CH10:”, optionally followed by information further identifying the type of RMM.

c. RMM Discovery. The RMM shall implement a service location protocol (SLP) service agent IAW IETF RFC 2608[44] and Table 10-58. The ground station may implement an SLP user agent or any other suitable method (e.g., tight integration with the DHCP server) to determine the IP address assigned to an RMM. The RMM may provide a set of service attributes IAW Table 10-58. The SLP authentication blocks shall not be required.

Table 10-58. Ethernet Service Location Protocol Characteristics

Characteristic	Provision	Type	Value
Service Name	Required	String	service:RMM:IRIG 106:
Service Location	Required	String	//nnn.nnn.nnn.nnn[:pppp]representing the IP address of the RMM and optionally the port number (pppp) on which the Telnet service will respond if not port 923 (see Paragraph 10.9.4.2)
Naming Authority	Optional	String	RCC. If used, the service name shall be service:RMM.RCC:IRIG 106:
Attributes			
Product	Optional	String	Identification of manufacturer, vendor, and/or part number of the RMM
SerialNo	Optional	String	Identification of the unique RMM
Capacity	Optional	Integer	Size of the RMM in gigabytes, rounded up.
Note: If present, the product string, serial number, and capacity attributes shall be used solely to identify a particular RMM, and shall not be used to modify the behavior of the ground system.			

d. Ping Response. The RMM shall respond to an internet control message protocol echo request IAW RFC 792.[45]

e. Accessing RMM Storage. In addition to the mandatory control interface via Telnet, the RMM bulk storage device shall support at least one of the following two methods of accessing data, and may support both:

(1) iSCSI. To facilitate random access, the iSCSI protocol IAW IETF RFC 3270[46] and the companion RFC 5048[47] may be implemented according to Subsection 10.9.3.3.

---

[42] Internet Engineering Task Force. “Dynamic Host Configuration Protocol.” RFC 2131. March 1997. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc2131/.
[43] Internet Engineering Task Force. “DHCP Options and BOOTP Vendor Extensions.” RFC 2132. March 1997. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc2132/.
[44] Internet Engineering Task Force. “Service Location Protocol, Version 2.” RFC 2608. June 1999. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc2608/.
[45] Internet Engineering Task Force. “Internet Control Message Protocol.” RFC 792. September 1981. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc792/.

(2) File Transfer Protocol. To facilitate efficient downloading with low overhead, the file transfer protocol (FTP) IAW IETF RFC 959[48] with optional extensions IAW RFC 3659[49] may be implemented according to Subsection 10.9.3.4.

10.9.3.3 iSCSI Data Access Method.

NEW

The RMM shall act as an iSCSI target and a host computing platform shall act as the iSCSI initiator. The RMM shall implement the commands defined by Subsection 10.9.13 item a when sent using iSCSI CDBs.

10.9.3.3.1 iSCSI Session Establishment.

The RMM shall support iSCSI features described in this section, sufficient to establish an iSCSI full feature phase between the ground system and the RMM.

a. IPsec. IPsec shall not be used.

b. Login Security. The ground system shall invoke the iSCSI login phase with the *LoginOperationalNegotiation* stage. The *SecurityNegotiation* stage shall not be used.

c. Target Naming. When an iSCSI target name is required, e.g., as a result of a SendTargets exchange, the RMM shall provide exactly one IQN per supported target. The name shall take the form:

iqn.yyyy-10.org.tscc: RMM:CH10.vvvvvvvv-ssssss

Where *yyyy* is the year corresponding to the applicable version of this standard and *vvvvvvvv-ssssss* is a pair of arbitrary length strings separated with a "-" that identify the manufacturer/vendor and part identifier of the type of RMM and the serial number or other unique identifier of that particular RMM. These strings shall not contain a colon (":") symbol.

NOTE	An RMM may support multiple targets. The name format described above shall not be used for any target that does not adhere to this standard, e.g., for non-compliant storage areas.

d. Header and Data Digests. Error detection digests shall not be required, but may be supported.

---

46 Internet Engineering Task Force. "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services." RFC 3270. May 2002. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc3270/.

47 Internet Engineering Task Force. "Internet Small Computer System Interface (iSCSI) Corrections and Clarifications." RFC 5048. October 2007. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc5048/.

48 Internet Engineering Task Force. "File Transfer Protocol (FTP)." RFC 959. October 1985. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc959/.

49 Internet Engineering Task Force. "Extensions to FTP." RFC 3659. March 2007. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc3659/.

e. Redirection. The RMM shall not employ redirection via the TargetAddress and TargetPortalGroupTag keys.

f. Burst and Segment Lengths. The RMM and the ground station shall support the default values per RFC 3720.[50]

g. Other Keys. For features to be negotiated during the login phase not otherwise specified, the RMM and the ground station shall support the default values per RFC 3720.

10.9.3.4 FTP Data Access Method.

The RMM shall implement an FTP server, and shall support image (aka binary) data representation and Passive mode. Unless changed by means of the .TCPPORTS command, the RMM shall employ TCP port 921. By default, the RMM shall accept a login username of “IRIG:CH10” with the associated password “RMM:FTP”. The RMM may also support anonymous FTP. If so the RMM shall provide a mechanism to disable this feature.

The RMM FTP server shall respond with an error code 550 and take no action in response to the DELE, MKD, RMD, RNFR, and RNTO commands.

10.9.4 RMM High-Level Command Handling.

Removable devices shall implement high-level Chapter 6 commands in addition to the data transport commands. These high-level commands and the associated responses shall be transported to the RMM depending on the host platform interface in use.

10.9.4.1 High-Level Commands for IEEE 1394b Host Platform Interface.

When using the IEEE 1394b interface, the SEND and RECEIVE processor device SCSI-2 commands shall be implemented. The Chapter 6 commands and data will be transported using these SCSI commands and the data buffers.

10.9.4.2 High-Level Commands for Ethernet Host Platform Interface.

When using the Ethernet interface, the RMM shall support a Telnet server IAW IETF RFC 854 using TCP port 923

10.9.5 Mandated Connectors.

Distinct from the recorder/RMM data interface, the removable media shall use the connector mandated for the host platform interface type.

10.9.5.1 IEEE 1394b Interface Connector.

The connector type for the removable media shall be an IEEE 1394b Bilingual Socket connector. Power for the removable media shall be derived from the bilingual interface connector.

[50] Internet Engineering Task Force. “Internet Small Computer Systems Interface (iSCSI).” RFC 3720. April 2004. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc3720/.

10.9.5.2 Ethernet Connector - Data.

The connector type for the removable media data connection shall be an 8P8c, commonly known as RJ45, connector. Power may also be supplied using this connector by means of the POE mechanism.

10.9.5.3 Ethernet Connector - Power.

The connector type for power when using Ethernet shall be a socket that accepts a barrel plug with a 5.5-millimeter (mm) outside diameter, a 2.5-mm inside diameter, and a shaft length of 9.5 mm. The plug shall be wired center-positive, and the connector shall carry a current of at least 5 amps.

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.

10.9.6.1 Minimum Operational Requirements.

The time setting accuracy of the real-time clock device shall be better than 1 ms. The short time accuracy of the real-time clock device must be at least 10 parts per million (ppm) in the temperature range 0-40°C, and at least 50 ppm in the temperature range -40°C - +85°C.

10.9.6.2 Accessing time using the IEEE 1394b Host Platform Interface.

The SCSI command set shall be utilized to access time on the cartridge.

a. Real-Time Clock Time Format. Time format shall be IAW Chapter 6 Subsection 6.2.2.39. The date format shall be IAW ISO 8601:2004.[51]

b. Real-Time Clock Logical Unit Number. The standard SCSI media devices are using LUN = 0. The real-time clock shall be assigned LUN =1.

10.9.6.3 Accessing time using the Ethernet Host Platform Interface.

The cartridge time shall be accessed via the .TIME command including the precision time protocol (PTP) extensions if supported.

10.9.7 Mandatory Commands for RMM Devices.

The mandatory commands for all RMM devices are listed in Table 10-59. Additional commands that are mandatory for all RMM devices that support declassification are listed in Table 10-60. Commands that are mandatory for RMM devices that support the Ethernet host platform interface via Telnet are listed in Table 10-61, with optional Ethernet commands listed in Table 10-62.

[51] International Organization for Standardization. *Data elements and interchange formats--Information interchange--Representation of dates and times*. ISO 8601:2004. Geneva: International Organization for Standardization, 2004.

NOTE The operation of these commands is described in Chapter 6, subsections 6.2.2.1-6.2.2.42 .

Table 10-59. Mandatory Processor Commands (All Interfaces)

Command	Parameters	Description
.BIT		Runs all of the RMM built-in tests
.CRITICAL	[n [mask]]	Specifies and views masks that determine which of the .HEALTH status bits are critical warnings
.DATE	[start-date]	Specifies setting or displaying date from RMM
.ERASE		Erases the RMM media
.HEALTH	[feature]	Displays detailed status of the RMM
.IDENTIFY		Queries the RMM for solid-state memory identification and firmware version
.INITIALIZE		Initializes RMM internal components
.IRIG106		Retrieves the IRIG-106 supported version number
.MEDIA P		Queries the RMM for information about the physical media of the RMM and the transfer limits for the required physical input/output (I/O) commands
.STATUS		Displays the current RMM status
.TIME	[start-time]	Displays or sets the internal system time.

Table 10-60. Additional Mandatory Processor Commands for Declassification

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
.DECLASSIFY		Initiates command as specified by user specification or user CONOP overwrite procedures
.PBWRITE P	{block identifier}	Directs the RMM to initiate a physical block write of the specified physical block identifier
.PBWRITE D		Writes 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

Table 10-61. Additional Mandatory Processor Commands for Ethernet Host Platform Interface

Command	Parameters	Description
.MEDIA E		Queries the RMM about which protocols it supports.
.RMMIP		Displays RMM IP address and associated settings. Mandatory only with Ethernet host platform interface.
.RMMIP	keyword [parameter]	Displays and controls RMM IP addressing. Mandatory only with Ethernet host platform interface.
.TIME	[|PTP |PTPSTATUS]	Displays and controls the IEEE 1588 PTP protocol (if implemented)
.TMATS	GET	Recovers the RSCF from the RMM storage
.TMATS	READ	Displays the RSCF
.TMATS	SAVE [n]	Saves the RSCF using n to form file name
.TMATS	WRITE	Uploads an RSCF

Table 10-62. Non-Mandatory Processor Commands for Ethernet Host Platform Interface

Command	Parameters	Description
.RMMFRAME		Displays the current and largest maximum frame size
.RMMFRAME	Frame size	Sets the maximum frame size
.TCPPORTS		Displays a comma-separated list of the TCP port numbers used for the Telnet, FTP, and iSCSI services.
.TCPPORTS	port1,port2,port3	Sets the ports used for the network services.

10.9.7.1 RMM .HEALTH Command Response.

The RMM .HEALTH command response is presented in Table 10-63.

Table 10-63. Removable Memory Module .HEALTH Command Response

	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

10.9.8 Time Setting Requirements.

To set time, the .TIME commands should be used according to Chapter 6, Subsection 6.2.2.39.

10.9.8.1 Time Setting Using IEEE 1394b.

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.

10.9.8.2 Time Setting using Ethernet.

NEW

To minimize inaccuracy, the IEEE 1588 PTP may be used. How an RMM derives time from PTP is not controlled by the standard. The .TIME PTP and .TIME PTPSTATUS variants of the .TIME command shall be used to enable and view the status of the PTP implementation.

10.9.9 Set Time.

To set time the .TIME commands should be used according to Chapter 6, Subsection 6.2.2.39.

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 Standard 8601:2004.

- 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 Chapter 6, Subsection 6.2.2.8 and Subsection 6.2.2.18.

10.9.12 Declassification Supporting Commands.

NOTE	See also Subsection 10.9.15 for commands unrelated to declassification.

10.9.12.1 .IDENTIFY

A .IDENTIFY command queries the RMM for 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 ms. A .STATUS command request prior to 100 ms 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 ms. A .STATUS command prior to 100 ms 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 Advanced Technology Attachment 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 FFs or all 00s. 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 10-101).

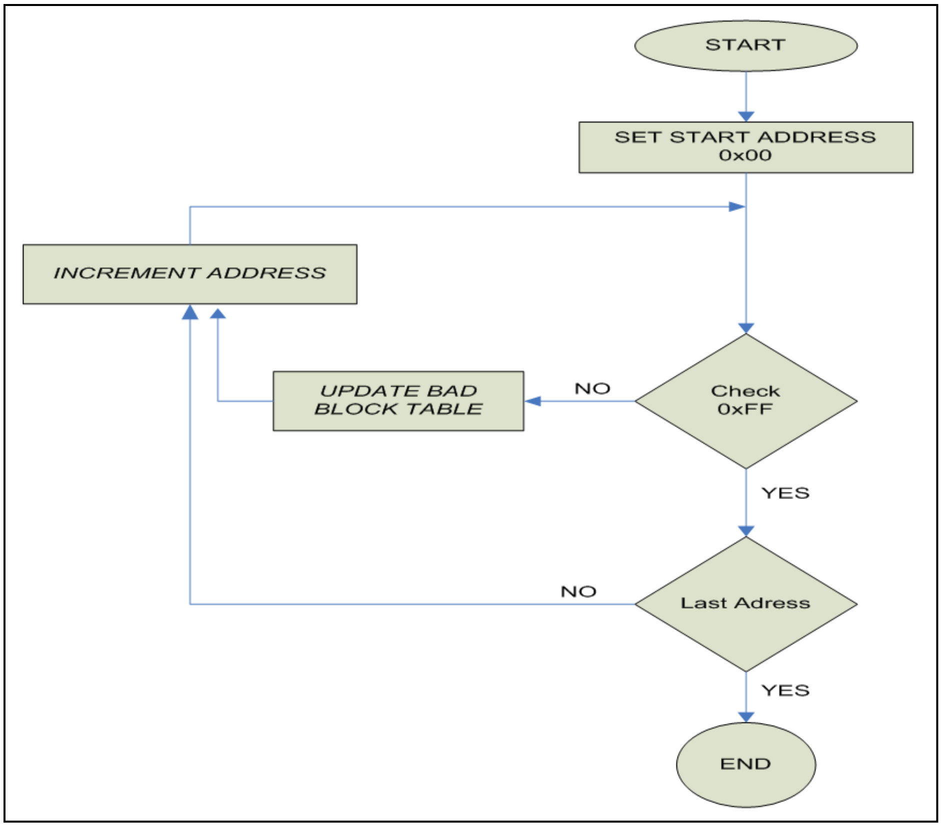


Figure 10-101. Updating the Bad Block Table

- Parameters

  None

- Response

  The RMM responds with an asterisk. Response time shall be within 100 ms. A .STATUS command prior to 100 ms 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 ms. A .STATUS command prior to 100 ms shall return a BUSY state.

- Example

  *.BBLIST

  *

10.9.12.5 .BBLIST R

A .BBLIST R command 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 ms. A .STATUS command prior to 100 ms shall return a BUSY state.

- Example

  *.BBLIST R

  000000E3

  0000034f

  FE0184C9

  *

10.9.12.6 BBREAD P {block_identifier}

A .BBREAD P {block_identifier} command 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 ms. A .STATUS command prior to 100 ms 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 physical block size, 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.

- Example

  *.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

  The parameter block_identifier is the physical block identifier from the BBLIST R response of the block to be secured.

- 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 ms. A .STATUS command prior to 100 ms 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.

- Example

  *.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 ms. A .STATUS command prior to 100 ms 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.

- Example

  *.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 ms. A .STATUS command prior to 100 ms 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 command shall be utilized to retrieve the RCC 106-supported version number.

- Parameters

  None

- Response

The RMM responds with a version number that shall be a two-integer value representing the last two digits of the year of RCC 106 release supported by the device. Response time shall be within 100 ms. A .STATUS command prior to 100 ms shall return a BUSY state.

- Example

  *.IRIG106

  09

  *

10.9.12.14 .STATUS

A .STATUS command shall be utilized to query the RMM for status information (see Table 10-64).

- 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 10-64. Removable Memory Module 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)	

Table 10-64. Removable Memory Module States

State	Description			
	State Code (A)	State Value (B)	State Value (C)	Progress Percentage(D)
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) results in error code responses (with an ASCII "E" identifier) prior to the asterisk response terminator when a command cannot be completed. Table 10-65 shows possible error codes and the conditions under which they occur.

Table 10-65. 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

- Example

  .CLEAR

  E 00

  *

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

CHANGE

NOTE	RMMs using either IEEE 1394b or iSCSI shall support as a minimum the SCSI command set to support data download IAW Section 10.4.

10.9.14 Mandatory ORB Formats for the Processor Device Using IEEE 1394b..

10.9.14.1 Minimum Operational Requirements.

The time setting accuracy of the real-time clock device shall be better than 1 ms. The short time accuracy of the real-time clock device must be better than 10 ppm in the temperature range 0-40°C and better than 50 ppm in the temperature range -40°C - +85°C.

10.9.14.2 IEEE 1394b ORB Format.

a. Login ORB format. The login ORB format is illustrated in Figure 10-102.

MSB						LSB
31	30 29	28	27 24	23 20	19 16	15 0
Password						
Login_response						
n	Rq_fmt	x	Reserved	reconnect	function	LUN
password_length						login_response_length
Status_FIFO						

Figure 10-102. Login ORB Format

- Password. In this 32-bit field, the password shall be "RTC." The password field shall contain the immediate data and the password_length shall be zero.
- Login_response. 32 bits.
- login_response_length. 16 bits.
    - The Login_response field and login_response_length fields 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. In this one-bit field, the notify bit "n" shall be one.
- Rq_fmt. In this two-bit field, the rq_fmt shall be zero.

- x. In this one-bit field, the exclusive bit “x” shall be one.
- Reserved. A four-bit field, Reserved shall be zero.
- reconnect. The four-bit reconnect field shall specify the reconnect time as a power of 2 seconds. A value of zero shall mean one second.
- Function. This field is four bits. The function shall be zero.
- LUN. This is 16 bits. The LUN shall be one.
- Status_FIFO. The 64-bit 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.

b. Login Response. The login response format is illustrated in Figure 10-103.

<table>
<tr><td>MSB
31 16</td><td>LSB
15 0</td></tr>
<tr><td>Length</td><td>login_ID</td></tr>
<tr><td colspan="2">command_block_agent</td></tr>
<tr><td>reserved</td><td>reconnect_hold</td></tr>
</table>

Figure 10-103. Login Response Format

- Length. This 16-bit field contains the length, in bytes, of the login response data.
- login_ID. This 16-bit field is used in all subsequent requests to the SSMC’s management agent.
- command_block_agent. This is a 64-bit field that contains the base address of the agent’s control and status register.
- Reserved. This 16-bit field shall be zero.
- Reconnect_hold. This 16-bit field is to be defined.

c. Send. The send command ORB format is illustrated in Figure 10-104, and the send data buffer format is illustrated in Figure 10-105.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.

<table>
<tr><td colspan="12">MSB LSB
31 30 29 28 27 26 24 23 21 20 19 18 17 16 15 8 7 0</td></tr>
<tr><td colspan="12">next_ORB</td></tr>
<tr><td colspan="12">data_descriptor</td></tr>
<tr><td>n</td><td>Rq_fmt</td><td>r</td><td>d</td><td>spd</td><td colspan="2">max_payload</td><td>p</td><td colspan="2">page_size</td><td colspan="2">data size</td></tr>
<tr><td colspan="5">0Ah</td><td>LUN</td><td colspan="3">Res</td><td>AEN</td><td colspan="2">Xfer Lng - upper bits</td></tr>
<tr><td colspan="5">Xfer Lng - lower bits</td><td colspan="5">Control</td><td>00h</td><td>00h</td></tr>
<tr><td colspan="5">00h</td><td colspan="5">00h</td><td>00h</td><td>00h</td></tr>
</table>

Figure 10-104. Send Command ORB Format

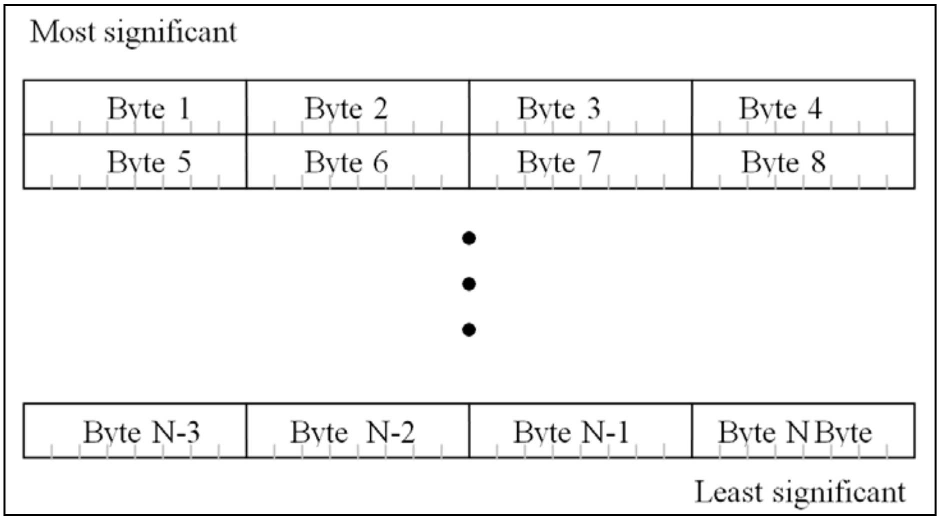


Figure 10-105. Send Data Buffer Format

- <u>next_ORB.</u> This 64-bit field contains the ORB pointer format, which shall be IAW SBP-2 specifications.
- <u>data_descriptor.</u> The 32-bit data_descriptor field shall contain the address of the data buffer.
- <u>n.</u> The completion notification "n" in this one-bit field shall be one. The target shall store a status block at the Status_FIFO address at the address supplied in the login request.
- <u>Rq_fmt.</u> Required format in this two-bit field shall be zero.
- <u>r.</u> Reserved in this one-bit field shall be zero.
- <u>d.</u> Direction bit in this one-bit field shall be zero.
- <u>spd.</u> This is a three-bit field that contains speed, which shall have a value of two.
- <u>max_payload.</u> A four-bit field, the maximum data transfer length shall be nine.
- <u>p.</u> This is a one-bit field. 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 process.
- <u>page_size.</u> This is three 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.
- <u>data size.</u> This is 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 (0x50).
- <u>LUN.</u> The LUN shall be one in this three-bit field.
- <u>Res.</u> This is a four-bit field. Reserved shall be zero.
- <u>AEN.</u> In this one-bit field, AEN shall be zero.
- <u>Xfer Lng.</u> This is 24 bits. The length must be at least 80 (0x50).

- Control. In this 8-bit field, control shall be 128.

d. Receive. The receive command block ORB format is illustrated in Figure 10-106.

<table>
<tr><td colspan="11">MSB</td><td>LSB</td></tr>
<tr><td>31</td><td>30 29</td><td>28</td><td>27</td><td>26 24</td><td>23 21</td><td>20</td><td>19</td><td>18 17</td><td>16</td><td>15 8</td><td>7 0</td></tr>
<tr><td colspan="12">next_ORB</td></tr>
<tr><td colspan="12">data_descriptor</td></tr>
<tr><td>n</td><td>Rq_fmt</td><td>r</td><td>d</td><td>spd</td><td colspan="2">max_payload</td><td>p</td><td colspan="2">page_size</td><td colspan="2">data size</td></tr>
<tr><td colspan="5">0Ah</td><td>LUN</td><td colspan="3">Res</td><td>AEN</td><td colspan="2">Xfer Lng - upper bits</td></tr>
<tr><td colspan="5">Xfer Lng - lower bits</td><td colspan="5">Control</td><td>00h</td><td>00h</td></tr>
<tr><td colspan="5">00h</td><td colspan="5">00h</td><td>00h</td><td>00h</td></tr>
</table>

Figure 10-106. Receive Commadn Block ORB Format

- next_ORB. This 64-bit field contains the ORB pointer format, which shall be IAW SBP-2 specifications.
- data_descriptor. The 32-bit data_descriptor field shall contain the address of the data buffer.
- n. The completion notification "n" in this one-bit field 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. Required format in this two-bit field shall be zero.
- r. Reserved in this one-bit field shall be zero.
- d. Direction bit in this one-bit field shall be zero.
- spd. This is a three-bit field that contains speed, which shall have a value of two.
- max_payload. A four-bit field, the maximum data transfer length shall be nine.
- p. This is a one-bit field. 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 process.
- page_size. This is three 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. This is 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 (0x50).
- LUN. The LUN shall be one in this three-bit field.
- Res. This is a four-bit field. Reserved shall be zero.
- AEN. In this one-bit field, AEN shall be zero.
- Allocation Lng. This is 24 bits. Allocation_Lng = length of the Chapter 6 response string.
- Control. In this 8-bit field, control shall be 128.

The receive data buffer can be returned in ASCII format (see Figure 10-107) or in binary format (see Figure 10-108) if the retrieved data contains binary information. Multiple ORBs may be used to retrieve the data required.

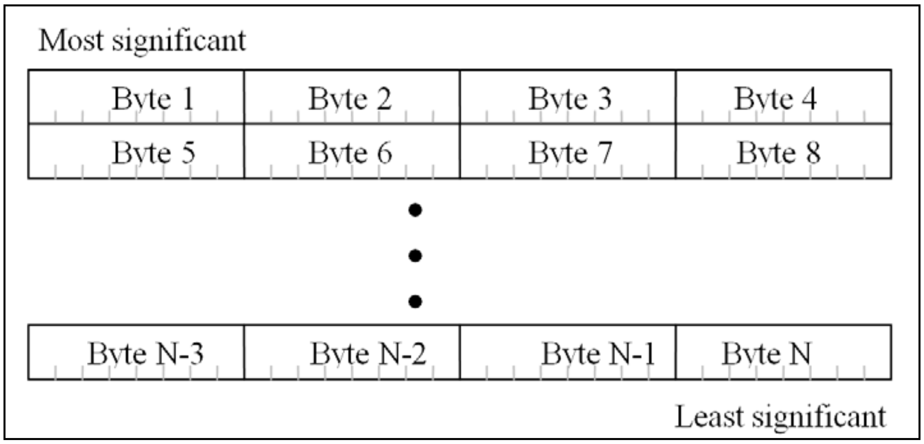


Figure 10-107. Receive Data Buffer Format ASCII Format

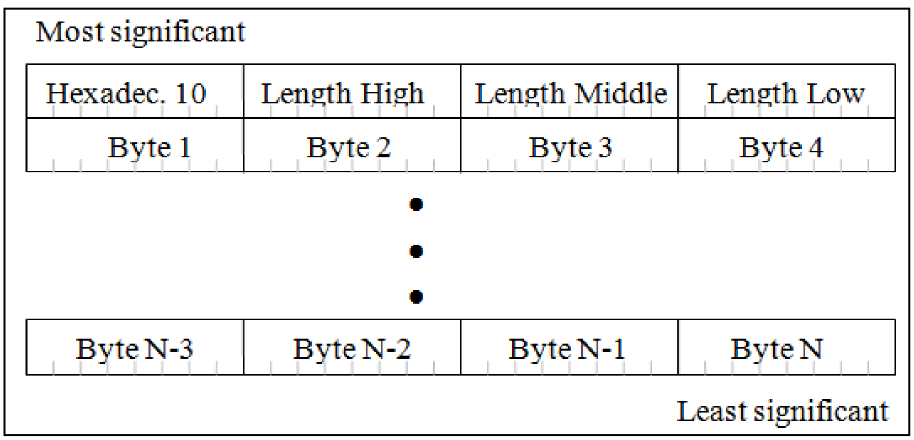


Figure 10-108. Receive Data Buffer Binary Format

- The returned remote answer is an ASCII text terminated by the "*" character IAW Chapter 6 Section 6.2. If the "*" terminator is missing, multiple receive commands must be used to retrieve the data until the "*" terminator is received.
- 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 "*" character IAW Chapter 6 Subsection 6.2.1. If the "*" terminator is missing, multiple receive commands must be used to retrieve the data until the "*" terminator is received.

10.9.15 Additional Mandatory Commands When Using Ethernet.

10.9.15.1 MEDIA E.

The .MEDIA E command is utilized to query the RMM for information regarding which of the data access protocols is supported.

- Parameters

  The parameter “E” distinguishes this command from the standard .MEDIA command.

- Response

  The RMM responds with one line containing the tag “PROTOCOLS” and at least one of the tags “FTP”, “ISCSI”, and “PTP” in alphabetical order each separated by a space. Response time shall be within 100 ms. A .STATUS command prior to 100 ms may return a BUSY state.

- Example

  *.MEDIA E

  PROTOCOLS FTP PTP

  *

10.9.15.2 .RMMIP

The .RMMIP command shall be utilized to display RMM IP address and addressing mode.

- Parameters

  None

- Response

  The RMM responds with one line containing the tag “IP_ADDRESS”, either the tag “STATIC” or “DHCP”, and three space-separated “dotted quad” IPv4 addresses, representing the IP address of the RMM, the net mask associated with that address, and the default gateway for the network associated with the net mask. If DHCP is being used and no DHCP address has been obtained, all three address fields shall be set to 0.0.0.0. Response time shall be within 100 ms. A .STATUS command prior to 100 ms may return a BUSY state.

- Examples

  *.RMMIP

  IP_ADDRESS STATIC 10.6.9.2 255.0.0.0 10.6.9.1

  *.RMMIP

  IP_ADDRESS DHCP 192.168.2.1 255.255.255.0 192.168.2.254

  *.RMMIP

  IP_ADDRESS DHCP 0.0.0.0 0.0.0.0 0.0.0.0

*

10.9.15.3 .RMMIP keyword [parameters]

The .RMMIP command shall be utilized to control RMM IP address and addressing mode.

- Keywords

  DHCP - used to set the RMM to DHCP mode.

  RESET - used to reset the Ethernet RMM to defaults, including IP addresses, frame size, and login passwords.

  xxx.xxx.xxx.xxx - used to set the RMM to static mode with the indicated IPv4 address; requires parameters. "xxx" indicates any number between 0 and 255.

- Parameters

  NetMask Gateway- used to specify the net mask for the static IP address and the default gateway for the network associated with the net mask. Each has the form xxx.xxx.xxx.xxx

- Response

  The RMM responds with an asterisk. Response time shall be within 100 ms. A .STATUS command prior to 100 ms may return a BUSY state.

- Examples

  .RMMIP DHCP

  *

  .RMMIP RESET

  *

  .RMMIP 192.168.10.99 255.255.255.0 192.169.10.254

  *

10.9.15.4 .TIME PTP

A .TIME PTP command shall be used to initiate the process of synchronizing the RMM real-time clock with a IEEE 1588 network time source. Note that successful synchronization with a time source will implicitly set the date as well as the time.

- Parameters

  The parameter "PTP" distinguishes this command from the standard .TIME command.

- Response

  The RMM responds with an asterisk. Response time shall be within 100 ms. A .STATUS command prior to 100 ms may return a BUSY state.

10.9.15.5 .TIME PTPSTATUS

A .TIME PTPSTATUS command shall be used to report the state of synchronization between the RMM real-time clock and an IEEE 1588 network time source.

- Parameters

  The parameter “PTPSTATUS” distinguishes this command from the standard .TIME command.

- Response

  The RMM responds with one line containing one of the words “LOCKED” or “NONE”, followed by an asterisk on a new line. “NONE” indicates that no sync has been obtained; “LOCKED” indicates that the RMM’s clock has been synchronized with a network clock. Response time shall be within 100 ms. A .STATUS command prior to 100 ms may return a BUSY state.

10.9.15.6 .TMATS GET

A .TMATS GET command shall be used to transfer the contents of the RSCF on the RMM media into a volatile buffer. No additional parameter is required, and if one is specified it shall be ignored.

- Parameters

  The parameter “GET” distinguishes this command from other .TMATS commands.

- Response

  The RMM responds with an asterisk. If no valid RSCF IAW Subsection 10.3.8.1 is located on the RMM media, an error is returned and the volatile buffer is erased. A .STATUS command prior to 100 ms may return a BUSY state.

10.9.15.7 .TMATS READ

A .TMATS READ command shall be used to display the contents of the volatile buffer created by either a .TMATS GET or a .TMATS WRITE command for the RSCF.

- Parameters

  The parameter “READ” distinguishes this command from other .TMATS commands.

- Response

  The RMM responds by displaying the contents of the volatile buffer followed by a line containing an asterisk. If the buffer contains no RSCF, no error shall be returned.

10.9.15.8 .TMATS SAVE n

A .TMATS SAVE command shall be used to transfer the contents of the volatile buffer created by a .TMATS WRITE command to the media. If the media already contains any data (except for a previous RSCF), an error shall be returned. The created file shall be IAW Subsection 10.3.8.1.

- Parameters

The parameter "SAVE" distinguishes this command from other .TMATS commands. The number following is used to generate the file name of the RSCF, "recorder_configuration_file_SAVE_n".

- Response

  The RMM responds with an asterisk. A .STATUS command prior to 100 ms may return a BUSY state.

10.9.15.9 .TMATS WRITE

A .TMATS WRITE command shall be used to transfer a TMATS file to the RMM for subsequent use as an RSCF.

- Parameters

  The parameter "WRITE" distinguishes this command from other .TMATS commands.

- Response

  The RMM responds by entering TMATS data transfer mode. All data sent to the RMM will be added to a volatile buffer until a line with the single word "END" is received, following which the RMM responds with an asterisk.

NEW

10.9.16 Additional Non-Mandatory Commands When Using Ethernet.

10.9.16.1 .RMMFRAME

The .RMMFRAME command shall be utilized to display the current and maximum values for the Ethernet frame size or MTU.

- Parameters

  None

- Response

  The RMM responds with one line containing two integers separated by a "/". The first integer indicates the currently configured frame size (default: 1500 bytes), and the second is the largest frame size supported by the RMM.

- Example

  *.RMMFRAME

  1500/9200

  *.RMMFRAME

  1500/1500

  *.RMMFRAME

  1300/9000

NOTE	An RMM command error code of 00 (“Invalid Command”) shall be interpreted to mean that the default value of 1500 bytes only is supported, and thus is synonymous with a response of “1500/1500”.

10.9.16.2 .TCPPORTS ffff.

A .TCPPORTS command with a parameter of an integer shall be used to configure the Ethernet frame size or MTU to be used.

- Parameters

  ffff where ffff is the value to be used.

- Response

  The RMM responds with an asterisk. A .STATUS command prior to 100 ms may return a BUSY state.

  Once the RMM has responded, all devices connecting to the RMM shall adjust their own frame size settings to match the new setting.

- Example

  *.RMMFRAME 9000

  *

10.9.16.3 .TCPPORTS

The .TCPPORTS command shall be utilized to display the port numbers used for the network services (Telnet, FTP, iSCSI).

- Parameters

  None

- Response

  The RMM responds with one line containing three comma-separated integers between 0 and 65535. The first integer indicates the port at which the Telnet server is listening, the next is the port used by the FTP server, and the third is for iSCSI. If an RMM does not support one of the two data access methods, it may report “0”.

- Example

  *.TCPPORTS

  923,921,3260

  *.TCPPORTS

  923,0,3260

  *.TCPPORTS

  928,921, 0

NOTE	Note: a response of “0,0,0” or an RMM command error code of 00 (“Invalid Command”) shall be interpreted to mean that the default ports are being used, and thus is synonymous with a response of “923,921,3260”.

10.9.16.4 .TCPPORTS ppp,qqq,rrr

A .TCPPORTS command with a parameter of three comma-separated integers between 0 and 65535 shall be used to configure TCP ports used by each of the three services defined for Ethernet RMM devices.

- Parameters

  ppp,qqq,rrr where ppp is the port to be used for the Telnet service, qqq is the port to be used for the FTP service, and rrr is the port to be used for iSCSI. A value of “0” in any one of the positions indicates that the current port configuration for that service is not to be changed.

- Response

  The RMM responds with an asterisk. A .STATUS command prior to 100 ms may return a BUSY state.

  If the port for the Telnet service is changed, the RMM may unilaterally disconnect (close the Telnet TCP connection) following the asterisk. The currently configured Telnet port shall be accessible by means of the SLP IAW Subsection 10.9.3.2 item b.

- Example

  *.TCPPORTS923,921,3260

  *

10.10 Ground-Based Recorders

This section of the standard specifies the basic requirements of ground-based recorders. The main functional requirements of ground-based recorders areas follows.

a. Recorder Interface

b. Recorder Data Format

c. Recorder Media

d. Recorder Command and Control (if the ground-based recorder is to be controlled remotely)

Optionally, ground-based recorders may support replay, reproduction, and display of 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-based recorder interface shall be Ethernet for remote command and control IAW sections 10.4 and 10.7 of this standard.

b. Optionally, ground-based recorders can implement additional interfaces for remote command and control, remote data access, and/or data streaming. If a ground-based recorder uses iSCSI or contains an RS-232/422, IEEE 1394, and/or Fibre Channel for these interfaces, it shall be IAW paragraphs 10.4 and 10.7 of this standard.

c. Data streaming.

- The recorder can optionally have the capability to stream Chapter 10 format data (Subsection 10.10.2) out of its required Ethernet interface IAW Subsection 10.3.9.1.
- Stream commit time as defined in Subsection 10.6.1 item g of this standard shall apply to Ethernet interface data streaming.

10.10.2 Data Format.

Ground-based recorders shall format, multiplex, and record all data IAW Section 10.6 of this standard.

10.10.3 Recording Media.

Ground-based recorders shall record data IAW Subsection 10.10.2 to COTS media. The term COTS is defined as any recording media (such as hard disks, solid-state drives, tape, Reduntant Array of Independent Disks, and Just a Bunch of Disks) that is ready-made and available for sale to the general public.

COTS media shall have an electrical interface (such as Parallel Advanced Technology Attachment, Serial Advanced Technology Attachment, IEEE 1394, Universal Serial Bus, SCSI, Ethernet) to the ground-based recorders that is ready-made and available for sale to the general public

NOTE	If ground-based recorders use COTS media for recording of the Subsection 10.10.2 data format, the recorded data remote data access at a minimum shall be across the required ground-based recorder Ethernet interface using iSCSI IAW Subsection 10.4.3 and Section 10.5 of this standard.

NOTE	If ground-based recorders provide remote data access across the ground-based recorder Ethernet interface, the interface file structure described in Section 10.5 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 interface file structure is presented at the recorder Ethernet interface.

COTS media used by ground-based recorders shall provide the capability of recording valid Chapter 10 original recording file(s) IAW Section 10.11. All Section 10.11 data transfer and file management requirements of this standard shall apply to ground-based recorders.

10.10.4 Remote Command and Control.

a. Optionally, if a ground-based recorder is controlled remotely, it shall provide command and control IAW subsections 10.7.8 across the Ethernet interface port as defined in Subsection 10.10.1.

b. Ground-based recorders at a minimum are required to use iSCSI or Telnet as the command and control Ethernet transport mechanism as defined in Sections 10.4 and 10.7.

c. Ground-based recorders providing remote command and control capability shall provide the functionality for all commands defined in Subsection 10.7.8.

d. Optionally, if a ground-based recorder contains an RS-232/422/485, IEEE 1394b, and/or Fibre Channel interface as defined in Subsection 10.10.1 the recorder will provide command and control IAW Section 10.7 and Chapter 6.

10.10.5 Data Replay and Reproduction.

10.10.5.1 Channel Mapping.

a. Optionally, if a ground-based recorder provides data playback capability, it shall provide for the logical assignment of recorded channels to physical channels on the ground-based 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-based recorder provides a data playback capability, it shall provide information using the Chapter 6 .HEALTH and .CRITICAL commands (Subsection 6.2.2.8 and Subsection 6.2.2.18) if the bandwidth of data to be played back exceeds the aggregate bandwidth of the ground-based recorder.

10.10.5.3 Network Recording Playback.

a. Optionally, if a ground-based recorder provides a data playback capability, it shall provide replay from COTS media (Subsection 10.10.3) to the Ethernet interface. The Ethernet format of the network recording playback will be IAW Subsection 10.3.9.1.

b. If the network recording playback capability is commanded remotely, ground-based recorders shall support the functionality specified in 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 Subsection 10.5.1 and data format in Section 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 Chapter 10 compliancy:
    - The Computer-Generated Data, Format 1 setup record shall always contain the required attributes IAW Section 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 Chapter 10 compliancy:

a. The Computer-Generated Data, Format 1 setup record shall always contain the required attributes IAW Section 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 that 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 Section 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 that 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 section.

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 the 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 On-Board Recorder.

a. Data Recording Directory Name. Each directory block from an RMM to be downloaded to a ground computing or storage platform shall use VolName as defined in Table 10-4 as the directory name where the data files will be placed. The directory name shall use lower-case letters.

   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 item a above and shall use the following naming convention. The data file name shall use lower-case letters:

   “filennnn”; where nnnn is the sequential RMM file count from each directory block file entry (must be 8 alpha-numeric characters).

   Example: “file0001,” “file0002,” …:”file9999.”

   If available, File Create Date, File Create Time, and File Close Time from Table 10-5, 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 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 letters.

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 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 subsections 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 Section 10.5 and a single or multiple 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 32 KB.

- Logical address 1 will contain a directory and file structure IAW Subsection 10.5.2.
- The corresponding 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 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 Subsection 10.5.2 and a single or multiple Chapter 10 original recording files or modified recording files. Multiple data transfer files can be contained on a random access device.

- The Subsection 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 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 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 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 Subsection 10.11.5

10.11.6 Recording Directory File.

A recording directory file is a binary file that is a byte-for-byte copy of the RMM or recorder directory structure presented at the interface. This file should represent the contents of an RMM or recorder directory at the time of 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 Chapter 10-compliant recording directory files shall use the file extension *.df10 (or *.d10 extension for use on systems with a three-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 Subsection 10.11.6.

Appendix 10-A. Definitions

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

Absolute Time: 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.

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

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).

Extended Relative Time Counter: A 1-GHz extension to the existing 10-MHz RTC.

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

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".

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

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.

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

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

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.

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

Playback: See Replay

Reconstruction: The output of a recorder where the timing and data content of the output signal are identical to the timing and data content of the originally recorded signal. This is generally the case where the input signal is captured using digital sampling techniques. Also see Reproduction.

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.

Removable Memory Module: The element of the on-board recorder that contains the stored data.

Replay: The virtual reconstruction of a recorded signal. This virtually reconstructed signal exists for the purposes of display, presentation, extraction, or retransmission.

Reproduction: The output of a recorder where the electrical characteristics of the output signal are identical to the characteristics of the originally recorded signal. This is generally only achievable when the input signal is captured using analog recording techniques. Also see Reconstruction.

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

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.

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

Appendix 10-B. References

Automated Imaging Association. *GiGE Vision*. Version 1.2. n.d. Available for download with registration at http://www.visiononline.org/form.cfm?form_id=735.

Department of Defense. "Aircraft Internal Time Division Command/Response Multiplex Data Bus." MIL-STD-1553B. 30 April 1975. May be superseded by update. Available at http://quicksearch.dla.mil/basic_profile.cfm?ident_number=36973&method=basic.

———. "National Imagery Transmission Format Version 2.1." MIL-STD-2500C. May 2006. May be superseded by update. Available at http://quicksearch.dla.mil/basic_profile.cfm?ident_number=112606&method=basic.

———. "Security Requirements for Automated Information Systems (AIs)." DoDD 5200.28. May be superseded by update. Available at http://csrc.nist.gov/groups/SMA/fasp/documents/c&a/DLABSP/d520028p.pdf

Director of Central Intelligence. "Protecting Sensitive Compartmented Information Within Information Systems." DCID 6/3. May be superseded by update. Available at http://www.fas.org/irp/offdocs/DCID_6-3_20Policy.htm.

European Air Group. "European Air Group Interface Control Document for Post Mission Interoperability." DR29125 Draft A Issue 01. July 2004. Available to RCC members with Private Portal access at https://wsdm.wsmr.army.mil/site/rccpri/Publications/106-13_Part%20I%20-%20Telemetry%20Standards/DF29125.pdf.

European Machine Vision Association. *GenICam Standard Features Naming Convention*. Version 1.5. November 2011. Available at http://www.emva.org/cms/upload/Standards/GenICam_Downloads/GenICam_SFNC_1_5.pdf.

Institute of Electrical and Electronics Engineers. *IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems.* IEEE 1588-2008. Geneva: International Electrotechnical Commision, 2008.

———. *IEEE Standard for a High-Performance Serial Bus.* IEEE 1394-2008. New York: Institute of Electrical and Electronics Engineers, 2008.

———. IEEE Standard for a High Performance Serial Bus: Amendment 2. IEEE 1394b-2002. New York: Institute of Electrical and Electronics Engineers, 2002.

———. *IEEE Standard for a High Performance Serial Bus*. IEEE 1394-1995. New York: Institute of Electrical and Electronics Engineers, 1995.

———. *IEEE Standard for Ethernet*. IEEE 802.3-2012. New York: Institute of Electrical and Electronics Engineers.

———. IEEE Standard for Information technology - Telecommunications and information exchange between systems…Amendment 3: Data Terminal Equipment (DTE) Power via the Media Dependent Interface (MDI) Enhancements. IEEE 802.3at-2009. October 2009. May be superseded by update. Available with registration at http://standards.ieee.org/about/get/802/802.3.html.

International Committee for Information Technology Standards. "Fibre Channel - Private Loop SCSI Direct Attach (FC-PLDA)." INCITS TR-19-1998. January 1998. Available for purchase at http://www.techstreet.com/incits/searches/385689. Replaced by "INCITS Technical Report - for Information Technology - Fibre Channel - Device Attach (FC-DA)." INCITS TR-36-2004. February 2005. Available for purchase at http://www.techstreet.com/incits/searches/385707.

International Organization for Standardization. *Data elements and interchange formats--Information interchange--Representation of dates and times*. ISO 8601:2004. Geneva: International Organization for Standardization, 2004.

International Organization for Standardization/International Electrotechnical Commission. *Information Technology - Coding of Audio-Visual Objects - Part 3: Audio*." ISO/IEC 14496-3 ed4.0. May be superseded by update. Available for purchase at http://webstore.iec.ch/Webstore/webstore.nsf/ArtNum_PK/43306!opendocument&preview=1.

———. *Information Technology - Coding of Audio-Visual Objects - Part 10: Advanced Video Coding*. ISO/IEC 14496-10:2012. April 2012. May be superseded by update. Available at http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html.

———. Information technology--Generic coding of moving pictures and associated audio information -- Part 3, Audio. ISO/IEC 13818-3:1998. Geneva: International Organization for Standardization, 1998.

———. Information technology -- Generic coding of moving pictures and associated audio information -- Part 7: Advanced Audio Coding (AAC). ISO/IEC 13818-7:2006(E). Geneva: International Organization for Standardization, 2006.

———. *Information technology -- Generic coding of moving pictures and associated audio information: Systems*. ISO/IEC 13818-1:2007. October 2007. May be superseded by update. Available for purchase at http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=44169.

———. *Information Technology -- JPEG 2000 Image Coding System: Core Coding System.* ISO/IEC 15444-1:2004. September 2004. May be superseded by update. Available for purchase at http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=37674.

———. *Information Technology -Universal Coded Character Set (UCS)*. ISO/IEC 10646:2012. May 2012. May be superseded by update. Available at http://standards.iso.org/ittf/PubliclyAvailableStandards/index.html.

International Telecommunications Union Telecommunication Standardization Sector. "Information technology - Generic coding of moving pictures and associated audio information: Systems." ITU-T Rec.H.222.0 (06/12). June 2012. May be superseded by update. Available to ITU-T members and other subscribers at http://www.itu.int/rec/T-REC-H.222.0/en.

Internet Engineering Task Force. "DHCP Options and BOOTP Vendor Extensions." RFC 2132. March 1997. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc2132/.

———. "Dynamic Host Configuration Protocol." RFC 2131. March 1997. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc2131/.

———. "Extensions to FTP." RFC 3659. March 2007. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc3659/.

———. "File Transfer Protocol (FTP)." RFC 959. October 1985. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc959/.

———. "Internet Control Message Protocol." RFC 792. September 1981. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc792/.

———. "Internet Protocol." RFC 791. September 1981. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc791/.

———. "Internet Small Computer Systems Interface (iSCSI)." RFC 3720. April 2004. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc3720/.

———. "Internet Small Computer System Interface (iSCSI) Corrections and Clarifications." RFC 5048. October 2007. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc5048/.

———. "Multi-Protocol Label Switching (MPLS) Support of Differentiated Services." RFC 3270. May 2002. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc3270/.

———. "Service Location Protocol, Version 2." RFC 2608. June 1999. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc2608/.

———. "Telnet Linemode Option." D. Borman, ed. RFC 1184. October 1990. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc1184/.

———. "Telnet Option Specifications." RFC 855. May 1983. May be superseded or amended by update. Available at http://datatracker.ietf.org/doc/rfc855/.

———. "Telnet Protocol Specification." RFC 854. May 1983. May be superseded or amended by update. Available at http://tools.ietf.org/html/rfc854.

Lockheed Martin Corporation. "Advanced Weapons Multiplex Data Bus." 8 June 2010. May be superseded by update. Available to RCC members with Private Portal access at https://wsdm.wsmr.army.mil/site/rccpri/Publications/106-13_Part%20I%20-%20Telemetry%20Standards/16PP362B.pdf.

Motion Imagery Standards Board. *Bit and Byte Order for Metadata in Motion Imagery Files and Streams*. STD 107.1. June 2011. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/standards/Standard010701.pdf.

———. "Data Encoding Using Key-Length-Value." STD 9713 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

———. "Imbedded Time Reference for Motion Imagery Systems." STD 9708 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

———. "Intelligence Motion Imagery Index, Content Description…". STD 9712 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

———. "Intelligence Motion Imagery Index, Geospatial Metadata." STD 9711 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

———. "Motion Imagery Systems Matrix, Standard Definition Motion Imagery." RP 9720d in *Motion Imagery Standards Profile.* MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

———. "Packing KLV Packets into MPEG-2 Systems Streams." RP 9717 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

———. "Standard Definition Digital Motion Imagery, Compression Systems." STD 9601 in Motion Imagery Standards Profile. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

———. "Time Reference Synchronization." STD 9715 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

———. "Use of MPEG-2 System Streams in Digital Motion Imagery Systems." RP 0101.1. 27 January 2011. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/rp/RP010101.pdf.

———. "Xon2". Subsection D-1.2 in *Motion Imagery Standards Profile*. MISP 6.4. 4 October 2012. May be superseded by update. Available at http://www.gwg.nga.mil/misb/docs/MISP64.pdf.

National Security Agency. “NSA/CSS Storage Device Declassification Manual.” Manual 9-12. n.d. May be superseded by update. Available at http://www.nsa.gov/ia/_files/government/MDG/NSA_CSS_Storage_Device_Declassification_Manual.pdf.

North Atlantic Treaty Organization. “NATO Advanced Data Storage Interface (NADSI).” STANAG 4575 (Edition 3). 8 May 2009. May be superseded by update. Available at http://www.nato.int/structur/AC/224/standard/4575/ag4_4575_E_ed3_nu.pdf.

Range Instrumentation System Program Office, Air Armament Center. “Interface Specification for the USAF Air Combat Test and Training System (ACTTS) Air-to-Ground, Air-to-Air, Ground-to-Air Data Links, and AIS Recording Formats." WMSP 98-01, Rev A, Chg 1. 19 May 2003. Available to RCC members with Private Portal access at https://wsdm.wsmr.army.mil/site/rccpri/Publications/106-13_Part%20I%20-%20Telemetry%20Standards/WMSP_98-01.doc.

****** END OF CHAPTER 10 ******