820-013
Deep Space Network (DSN)
External Interface Specification

JPL D-16765

0161-Telecomm

Telemetry Standard Formatted Data Unit (SFDU) Interface

Revision A: August 29, 2008

Document Owner:

Signature on file in DSN Library

Harvey Soldan Date
System Engineer for Telemetry, Tracking and Command

Concurred by:

Signature on file in DSN Library

Jeff Berner Date
Development and Operations Chief Engineer

Approved by:

Signature on file in DSN Library

T. Pham Date
Chief DSN System Engineer

Approved by:

Signature on file in DSN Library

J. Stipanuk Date
Interface Engineer and Release Authority

Prepared By:

Signature on file in DSN Library

Tracy Clark Date
JPL CCSDS Control Authority

Reviewed By:

Signature not available

C. Miyatake Date
Downlink Channel Control Software CDE

Signature not available

Ana Guerrero Date
Manager, DDOSO Telemetry, Tracking, Command
End-to-End Data Office

Signature not available

Richard D. Benson Date
Telecommunications and Mission System Manager

Signature not available

Eugene S. Burke Date
Telecommunications and Mission System Manager

Signature not available

Daniel F. Finnerty Date
Telecommunications and Mission System Manager

Signature not available

Dwight P. Holmes Date
Telecommunications and Mission System Manager

Signature not available

Andrew Kwok Date
Telecommunications and Mission System Manager

Signature not available

Peter T. Poon Date
Telecommunications and Mission System Manager

Signature not available

_______________________________ _________
Stefan Waldherr Date
Telecommunications and Mission System Manager

Signature not available

_______________________________ _________
Byron G. Yetter Date
Telecommunications and Mission System Manager

Signature not available

_______________________________ _________
M. Stoloff Date
DSN Telemetry System Engineer

Change Log

Rev.	Check if Minor Rev.	Issue Date	Affected Sections	Change Summary
--		11/15/2002	All	Initial Release (was originally released as TLM 3-29, Revision A)
A		08/29/2008	Section 1, 3 and Appendix Section 3.5	Added new references and used some spare bits for indicating data states. Replaced 'DSMS' with 'DSN' throughout Changed Byte 12 bits 2 and 3 of secondary CHDO

This page is intentionally left blank.

Contents

Figures

Tables

Section 1
Introduction

1.1 Purpose

This interface module defines and controls the format and contents of telemetry data blocks that are produced by the Deep Space Network (DSN) in support of missions that require telemetry to be delivered as Standard Formatted Data Units (SFDUs). A telemetry SFDU, as specified by this module, is a self-identifying, self-delimiting data structure that is used to encapsulate a portion of the telemetry data acquired from a spacecraft by the DSN for delivery to a mission. Typically, each SFDU encapsulates one telemetry transfer frame. A given sequence of telemetry SFDUs may encapsulate one physical channel from a spacecraft, or it may encapsulate only a part of a physical channel (e.g., a virtual channel). Each SFDU also contains additional information related to the telemetry data encapsulated in the SFDU (e.g., their Earth Received Time).

The scope of this module is limited to the format and contents of the telemetry SFDU. The communications interfaces and protocols used to effect delivery of telemetry SFDUs from the DSN to a mission are outside the scope of this module. The telemetry SFDU is compatible with a variety of transmission and storage media. For example, a given sequence of telemetry SFDUs may be delivered to the mission as a real-time data stream using a communications protocol such as TCP/IP; or it may be stored as a file and delivered to the mission in non-real-time using a file transfer protocol such as FTP. Communications interfaces and protocols are specified elsewhere (e.g., in other modules of this document).

The decision to format telemetry data for a mission in accordance with the specifications of this module is the result of an agreement between the DSN and the mission. Such an agreement also must specify certain additional information:

a) the mission-specific values of certain parameters referenced within this module (e.g., the spacecraft identifier);

b) mission-specific choices with respect to certain options defined by this module (e.g., whether the data from one physical channel is to be formatted as one sequence of telemetry SFDUs or whether the physical channel is to be split by virtual channel into multiple sequences of telemetry SFDUs); and

c) the particular communications interfaces and protocols to be used to transport the telemetry SFDUs from the DSN to the mission.

1.2 Applicability

This module was originally released as TLM-3-29, Revision A. After that release, it was decided that the original version of TLM-3-29 would remain in effect, so this module is being released with this new module number.

The applicability of this module to a particular mission typically is specified in a separate, signed agreement between the DSN and the mission (e.g., a Detailed Mission Requirements document or a Project Level Service Agreement).

1.3 Revision Control

Revision or changes to information contained in this module shall be made according to the policies and procedure specified in reference[2]. As described there, any interested person may initiate such revision or changes by consulting with the Document Owner identified on the signature page of this module. This revision uses some of the spare bits for conveying additional configuration data and maps in new references to higher level documentation. This revision uses some of the spare bits for conveying additional configuration data and maps in new references to higher level documentation.

Documents controlling this version include

[1]	DSN 820-062	*DSN Terms and Abbreviations* (DSN internal document, for reference only.)
[2]	DSN 813-109	*Preparation Guidelines and Procedures for Deep Space Mission System (DSMS) Interface Specifications* (DSN internal document, for reference only.)

1.4 Relationship to Other Documents

This module is produced, approved, and revised in accordance with the policies and procedures specified in reference **Error! Reference source not found.**. The structure and construction of the telemetry SFDU conforms to the SFDU specifications given in reference [7]. Certain fields within the telemetry SFDU contain identifiers, codes, or other values that are specified in references [4], [5], and [12]. Telemetry acquired from a spacecraft by the DSN and encapsulated in telemetry SFDUs frequently (but not necessarily) conforms to CCSDS Recommendations (e.g., references [8], [9], and [11]). To facilitate end-to-end processing within the DSN, telemetry SFDUs may be encapsulated within the user data field of a Standard Deep Space Network (DSN) Block (SDB) as specified in reference [3].

1.5 Terminology and Notation

1.5.1 Terminology

Many of the terms used in this module are taken from the literature describing the Standard Formatted Data Unit (SFDU) concept (e.g., references [7] and [10]). The SFDU concept was developed by the Consultative Committee for Space Data Systems (CCSDS) to provide a standardized and internationally recognized methodology for information interchange. As the SFDU concept evolved over time, the meaning of some terms also has evolved. The definitions provided here are intended to clarify the use of certain terms as they apply to this module:

a) A *label-value-object* (LVO) is a data structure that is comprised of a *label field* and a *value field*. The label field provides for the data structure to be self-identifying and self-delimiting. The value field contains user-defined data in any format. The

LVOs themselves are made up of a sequence of octets. In this module, LVO is used in a generic sense to refer to any data structure with these attributes.

b) An LVO may be a *simple LVO* or a *compound LVO*. If the value field of the LVO contains purely user data, it is a simple LVO. If the value field of the LVO contains purely LVOs, it is a compound LVO. The value field of a compound LVO consists of a sequence of one or more LVOs, each of which can be a simple or compound LVO itself.

c) A *standard formatted data unit* (SFDU) is an LVO that conforms to a defined set of structure and construction rules, namely the specification in reference [7] or the specification in reference [10]. Unfortunately, the two specifications are slightly different, leading to two different definitions of what an SFDU is. The term *DSN telemetry SFDU* (or, more simply, *telemetry SFDU*) refers to the SFDU defined and controlled by this module. The DSN telemetry SFDU conforms to the structure and construction rules specified in reference [7]. It does not strictly conform to the internationally recognized SFDU structure and construction rules recommended by CCSDS in reference [10].[1]

d) A *compressed header data object* (CHDO), as defined in reference [7], is an LVO. Its design is modeled on the SFDU concept, but a CHDO is not an SFDU. The CHDO derives its name from that fact that the label field of a CHDO is considerably shorter that the label field of an SFDU (four octets instead of twenty). The CHDO provides a means of structuring user data with less overhead than would be obtained if an SFDU were used. However, with respect to SFDU structure and construction rules, a CHDO (or a sequence of CHDOs) is merely user data contained in the value field of an SFDU.

e) The term *type attribute* is used to refer to the subfield(s) of an LVO label field that effects the self-identifying property of the LVO. Within the applicable domain, the type attribute is a unique reference to a description of the format and interpretation of the data contained in the value field of the LVO. For SFDUs, type attributes are assigned by a *control authority*, which is also responsible for maintaining the associated data descriptions. The CCSDS maintains a registry of control authorities.

f) All of the LVOs described in this module contain a *length attribute* in their label field. The length attribute is a subfield of the LVO label field; it contains the length, in octets, of the value field of the LVO. When interpreted in the context of the structure and construction rules specified in reference [7], the length attribute effects the self-delimiting property of the LVO. The use of a length attribute is not the only means by which an LVO can be self-delimiting; reference [10], for example, provides several mechanisms that do not rely on an explicit length.

g) The terms *byte* and *octet* are used interchangeably in this module to refer to an eight-bit quantity. The term *word* is used to refer to a sixteen-bit quantity.

[1] The DSMS telemetry SFDU conforms to the CCSDS Recommendation except for the use of the SFDU class identifier field (the sixth octet of the SFDU label).

h) The term *ASCII* refers to the American Standard Code for Information Interchange, a seven-bit code for representing letters, digits, and symbols that has been standardized by the American National Standards Institute (reference [13]). This code has been incorporated into the ISO code of the same nature (reference [14]), which includes other symbols and alphabets. Since the ISO code is an eight-bit code, the ASCII code is embedded in an eight-bit field in which the most significant bit is set to zero. In this module, ASCII always refers to the seven-bit ASCII code embedded, as described, in an eight-bit field. When applied to a multi-octet field, it implies that each octet in the field contains an ASCII code.

i) The term *restricted ASCII (RA)* refers to the subset of ASCII consisting of the codes for the twenty-six uppercase letters (‘A’–‘Z’) and the ten decimal digits (‘0’–‘9’). When applied to a multi-octet field, it implies that each octet in the field contains an RA code.

1.5.2 Conventions

The following conventions are used in this module.

a) All numbers are expressed in decimal unless explicitly indicated otherwise by means of a subscript designating another base (e.g., $4F_{16}$ for hexadecimal).

b) Data structures in this module are illustrated as sequences of eight-bit bytes. Given a data structure that is *N* bytes in length, the first byte in the structure is drawn in the most top-justified position and is identified as “byte 0.” The following byte is identified as “byte 1” and so on, to “byte *N-1*,” which is drawn in the most bottom-justified position. Within each byte, the most significant bit is drawn in the most left-justified position and is identified as “bit 1.” The next most significant bit is identified as “bit 2” and so on, to “bit 8” which is drawn in the most right-justified position. Any bit in a data structure is uniquely identified by specifying the byte within which it occurs and its position within that byte (e.g., “byte 5, bit 8”).

c) Data structures are divided into fields, where a field is a sequence of bits. Fields are identified by specifying the starting and ending bits of the field (e.g., “byte 2, bit 5, through byte 2, bit 8,” identifies the right 4 bits in byte 2). For fields that cross byte boundaries, bit 8 of byte M is more significant than, and is immediately followed by, bit 1 of byte M+1. A field may be divided into subfields in a similar manner.

d) Several conventions for expressing the length of a data structure, or a part of a data structure, are used in this module. The length attribute of an LVO is always given in octets and always refers to the length of the value field of the LVO (i.e., excluding the label field). In other contexts, a length may be given in bits or octets, and the length may refer to the entire data structure or only a part of it. In all such cases, the units and meaning are explicitly stated.

e) In the data structure descriptions in this module, many fields are defined to contain a numerical value. Several different formats for expressing numbers are used, as follows:

1) *Binary unsigned integer*. A non-negative integer is expressed in binary, using all bits of the field as necessary. Negative quantities cannot be expressed. For an *n*-bit field, the range of values that can be represented is from 0 to $2^n–1$.

2) *Binary integer*. An integer is expressed in binary, using two's complement notation. For an *n*-bit field, the range of values that can be represented is from $–2^{n–1}$ to $2^{n–1}–1$.

3) *Restricted ASCII*. Each decimal digit of an integer is expressed by its corresponding RA code. The field must be an integral number of octets in length. For multi-digit fields, the first octet of the field contains the most significant digit, the second octet contains the next most significant digit, and so on. If the number of digits is less than the number of octets in the field, leading zeroes are used to fill the field. Negative quantities cannot be expressed. For an *n*-octet field, the range of values that can be represented is from 0 to $10^n–1$.

4) *Binary-coded decimal (BCD)*. Each decimal digit of an integer is expressed in binary in a four-bit subfield. The length of the entire field, in bits, must be an integral multiple of four. For multi-digit fields, the first four-bit subfield contains the most significant digit, the second four-bit subfield contains the next most significant digit, and so on. If the number of digits is less than the number of four-bit subfields, leading zeroes are used to fill the field. Negative quantities cannot be expressed. For an *n*–bit field, the range of values that can be represented is from 0 to $10^{n/4}–1$. In certain rare cases, where it is known that a digit cannot be greater than three (e.g., the hundreds digit of the day-of-year), a BCD digit is allowed to occupy a two-bit subfield, and the length of the field and the range of values are adjusted accordingly.

5) *Floating point*. A 32-bit, single precision, floating point format is used to express real numbers. Refer to paragraph 1.5.3 belowfor details.

6) *Time*. The format of the Earth Received Time (ERT) field (in the secondary CHDO of the telemetry SFDU) is described in paragraph 1.5.4 below. The format of other fields related to time use the basic numeric formats already described.

7) *Facility/subfacility code*. Facility/subfacility codes are written using a dotted notation (e.g., 10.14). In this usage, the dot is not a decimal point; rather, it is a separator that delimits the facility code (10 in the example) from the subfacility code (14 in the example). In a data structure, the facility and subfacility codes are contained in separate subfields (a seven-bit subfield and a four-bit subfield, respectively) and each is expressed as a binary unsigned integer.

f) For fields defined to contain a constant value, the constant value will be enclosed in single quotes (e.g., ‘2’) if the information is expressed in ASCII or RA and not so enclosed (e.g., 2) if the information is expressed in binary.

g) Unless explicitly stated otherwise, fields defined as “reserved” are to be set to binary zero by the originator and are to be ignored by the recipient.

1.5.3 Thirty-two-bit Floating Point Format Description

Floating point numbers are expressed in the ANSI/IEEE standard single precision format (reference[12]) with a sign, 8-bit exponent, and 23-bit significant as illustrated in Figure 1-1.

```
          BIT
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
   BYTE 0| S | EXPONENT              | SIGNIFICANT         |
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
        2| SIGNIFICANT (continued)                       |
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
         |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
         +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Figure 1-1. Floating Point Format

The value *v* of the number expressed in this format is determined as follows:

Let:

s be the value (0 or 1) of the S (sign) field;
e be the value in the EXPONENT field (interpreted as an unsigned integer); and
f be the value in the SIGNIFICANT field (interpreted as an unsigned integer).

Then:

if $e = 255$ and $f \neq 0$, then *v* is not a number (NaN) regardless of s;
if $e = 255$ and $f = 0$, then $v =(-1)^S$, (i.e., 1 or –1);
if $0 < e < 255$, then $v =(-1)^S(2^{e-127})(1+f/(2^{23}))$;
if $e = 0$ and $f = 0$, then $v = (-1)^S 0$, (i.e., +0 or –0).

Bit strings of the form $e = 0$ and $f \neq 0$ (denormalized numbers) are not permitted in this interface.

1.5.4 Earth Received Time Format Description

The format and interpretation of the Earth Received Time field in the secondary CHDO (see paragraph 3.5 below) is illustrated in Figure 1-2 and described in the following paragraphs.

```
             BIT
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
            |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     BYTE 0| DAYS SINCE JANUARY 1, 1958                      |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           2| MILLISECONDS-OF-DAY                            |
            +                                               +
           4|                                                |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
           6| EXTENDED RESOLUTION                            |
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
            |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
            +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Figure 1-2. Earth Received Time Format

BYTES 0 AND 1

BITS

1 thru 8 Days since January 1, 1958 (e.g., January 2, 1958, is 1); value range 0-65535; binary unsigned integer.

BYTES 2 TO 5

BITS

1 thru 8 Milliseconds-of-day (i.e., milliseconds since midnight UTC); value range 0-86,400,000 (allows for one leap second); binary unsigned integer.

BYTES 6 AND 7

BITS

1 thru 8 Extended resolution; binary unsigned integer. The use of this field to provide extended ERT resolution must be negotiated between the mission and the DSN. Normally, the ERT resolution is one millisecond, and this field is reserved. When extended resolution is provided, it can be defined as the number of tenths of microseconds since the millisecond indicated by bytes 2 to 5 (range 0–9,999), or the number of microseconds since the millisecond (range 0 – 999). The fact that the ERT may be provided in a field that allows for up to tenths-of-microseconds resolution should not be interpreted as implying that the DSN is capable of providing ERTs with tenth-of-microseconds accuracy.

1.6 References

The latest releases of the following published documents form a part of this document to the extent specified herein. In case of conflict between this document and those listed below, this document shall govern.

Documents

[3]	DSN 820-019, Module DFL-1-1	*Interface Design Standards—Network-Level Data Flow Standard* (DSN internal document, for reference only.)
[4]	810-047	*DSN Antenna and Facility Identifiers* (DSN internal document, for reference only.)
[5]	DSN 820-013, Module OPS-6-21	*DSN External Interface Specification—Standard Code Assignments*
[6]	DSN 810-005, Rev. E	*DSN Telecommunications Link Design Handbook*
[7]	820-013, Module 0172-Telecomm-CHDO	*DSN External Interface Specification—DSN Created CHDO Structures*
[8]	CCSDS 101.0-B-4	*CCSDS Recommendation for Space Data System Standards—Telemetry Channel Coding* (Issue 4, May 1999)
[9]	CCSDS 102.0-B-4	*CCSDS Recommendation for Space Data System Standards—Packet Telemetry* (Issue 4, November 1995)
[10]	CCSDS 620.0-B-2	*CCSDS Recommendation for Space Data System Standards—Standard Formatted Data Units—Structure and Construction Rules* (Issue 2, May 1992)
[11]	CCSDS 701.0-B-2	*CCSDS Recommendation for Space Data System Standards—Advanced Orbiting Systems, Networks and Data Links—Architectural Specification* (Issue 2, November 1992)
[12]	ANSI T-49-12	*ANSI/IEEE STD 754-1985—IEEE Standard for Binary Floating-Point Arithmetic*
[13]	ANSI X3.4-1986 (R1997)	*Information Systems - Coded Character Sets - 1 Bit American National Standard Code for Information Interchange (7-Bit ASCII)*
[14]	ISO/IEC 646-1991	*Information Technology - ISO 7-bit Coded Character Set for Information Interchange*
[15]	820-013 module 0171-Telecomm-NJPL	*DSN External Interface Specification – JPL Created SFDU Structures*

Web Sites

https://jaguar.jpl.nasa.gov	DSN Interface Server for DSN interface documentation [Access restrictions]
http://www.ccsds.org/	The Consultative Committee for Space Data Systems For CCSDS documentation

1.7 Abbreviations and Acronyms

Abbreviations and acronyms used in this document are defined where they first occur in the text. A complete list is provided here for the convenience of the reader.

ADID	Authority and Description Identifier
AMMOS	Advanced Multi-Mission Operations System
ANSI	American National Standards Institute
APC	Automatic Polarity Correction
ASCII	American Standard Code for Information Interchange
ASM	Attached Sync Marker
BCD	Binary-Coded Decimal
BET	Bit Error Tolerance
BPSK	Binary Phase Shift Keying
BSN	Block Serial Number
BVR	Block V Receiver
BWG	Beam Waveguide
CCSDS	Consultative Committee for Space Data Systems
CDE	Cognizant Development Engineer
CDR	Central Data Recorder
CHDO	Compressed Header Data Object
CRC	Cyclic Redundancy Code
DC	Downlink Channel
DDD	DSN Data Delivery
DSCC	Deep Space Communications Complex
DSN	Deep Space Network
DSS	Deep Space Station
DTF	Development and Test Facility
DTM	DSCC Telemetry Subsystem
ERT	Earth Received Time
FTP	File Transfer Protocol
GCF	Ground Communications Facility
GCS	Ground Communications System
GOS	Grade of Service
HEF	High Efficiency

HSB	High Speed Beam Waveguide
ID	Identifier
IDR	Intermediate Data Record
IND	Interplanetary Network Directorate
IP	Internet Protocol
ISO	International Organization for Standardization
JPL	Jet Propulsion Laboratory
K	Constraint Length
LVO	Label-Value-Object
MCC	Mission Control Center
MFR	Multifunction Receiver
MGDS	Multi-Mission Ground Data System
MIL	Merritt Island
N	Length of DSN Telemetry SFDU, in 8-bit bytes
NASA	National Aeronautics and Space Administration
NJPL	NASA/Jet Propulsion Laboratory (control authority identifier)
NOPE	Network Operations Project Engineer
NRZ-L	Non-return to Zero—Level
PM	Phase Modulation
R	Code Rate
RA	Restricted ASCII
RCP	Receiver Channel Processor
RF	Radio frequency
RS	Reed-Solomon
RSN	Record Sequence Number
SDB	Standard DSN Block
SFDU	Standard Formatted Data Unit
SNR	Signal-to-Noise Ratio
TCA	Telemetry Channel Assembly
TCP	Transmission Control Protocol
TLM	Telemetry
TMS	Telecommunications and Mission Services
UTC	Coordinated Universal Time
VC	Virtual Channel

VCID	Virtual Channel Identifier
VCDU	Virtual Channel Data Unit

This page intentionally left blank.

Section 2
Functional Overview

2.1 General Information

The DSN uses its telecommunications and data acquisition facilities to acquire telemetry data from a mission's spacecraft. DSN facilities are located at various sites around the world to enable continuous tracking of spacecraft in deep space. For the purposes of this module, each such site is referred to as a Deep Space Station (DSS). At a DSS, the RF downlink from the spacecraft is down-converted, amplified, demodulated, and bit synchronized to produce a stream of digital symbols. Depending on the requirements of the mission, further processing may be performed by the DSN, including convolutional decoding, frame synchronization, Reed-Solomon decoding, and turbo decoding. (Note that turbo coding and convolutional/Reed-Solomon coding are mutually exclusive.) The resultant telemetry data stream is annotated with DSN status, configuration, and performance information and formatted into telemetry SFDUs. The turbo telemetry SFDUs are transported from the DSS to the mission control center (or other mission-specified sites) where they are made available to mission controllers, engineers, and scientists. To protect against potential data losses that may be caused by certain classes of ground system failure, all telemetry SFDUs are recorded by the DSN and are recoverable from that recording for some time after they are acquired.

Within the general framework just described, there are a number of variations with regard to the nature of the spacecraft telemetry, the kinds of processing that are performed by the DSN, and the methods by which the data are delivered. The remainder of this section describes those aspects in more detail.

2.2 Spacecraft Telemetry

Spacecraft telemetry systems differ with respect to carrier frequency, modulation scheme, coding, data rate, frame size, and other characteristics. In order to be supported by the DSN, the characteristics of a spacecraft's telemetry system must be compatible with the DSN, as specified in reference [6]. At a minimum, there must be compatibility in terms of the radio frequency (RF) system, the modulation scheme, the symbol rate, and the link performance, such that the DSN is capable of acquiring a *physical channel* (i.e., a stream of digital symbols) from the spacecraft.

Other characteristics of the spacecraft telemetry system affect the extent of the processing that can be performed by the DSN for the mission. The physical channel may be *coded* or *uncoded*. The DSN is capable of decoding a coded channel if it conforms to the CCSDS Recommendation on telemetry channel coding (reference [8]). The CCSDS Recommendation defines a *convolutional code* (K=7, R=1/2) and a *Reed-Solomon code* (255, 223), which may be used separately or together (*concatenated code*). It also defines different types of turbo codes. The DSN also supports certain advanced decoding methods for missions with special requirements.

The physical channel typically is formatted as a stream of fixed-length *frames* with an *attached sync marker* (ASM) between them. A frame is a bit string, typically several thousand bits long. An attached sync marker is a bit string of fixed length (for non-turbo coding, it is typically 32-bits, and for turbo codes, it is 32 bits divided by the coding rate) and of a known, fixed pattern. The DSN is capable of recognizing and delimiting frames if the spacecraft telemetry system conforms to the CCSDS Recommendation on synchronization (reference [8]).

Many spacecraft conform to the CCSDS Recommendation on packet telemetry (reference [9]); for those spacecraft, a frame is a *transfer frame* or a *turbo codeblock* (i.e., a transfer frame that is either concatenated with *Reed-Solomon check symbols* or turbo encoded). Other spacecraft conform to the CCSDS Recommendation on advanced orbiting systems (reference [11]); for those spacecraft, a frame is a *virtual channel data unit* (VCDU) or *coded virtual channel data unit* (CVCDU). Transfer frames, VCDUs, and CVCDUs all have a well-defined *primary header* whose first two bits are a *version number*. The version number enables one to distinguish a *Version-1 frame* (a transfer frame) from a *Version-2 frame* (a VCDU or CVCDU). The primary header of a Version-1 or a Version-2 frame also provides a *spacecraft identifier* and a *virtual channel identifier*. The time-ordered sequence of all frames from a physical channel that have the same version number, the same spacecraft identifier, and the same virtual channel identifier constitutes a *virtual channel*. The use of virtual channels allows logically distinct data streams to share one physical channel. The DSN is capable of distinguishing virtual channels and processing them in varying ways (e.g., to deliver different virtual channels to different destinations). The most common use of this capability is to optimize the use of ground communications bandwidth. One virtual channel, containing low rate spacecraft health and safety information, is delivered to the mission in real-time; the other virtual channels, containing high rate science data, are recorded in real-time and delivered later by means of a file transfer protocol using available ground communications bandwidth. A *virtual stream* is made up of one or more virtual channels. The virtual stream assignment is determined by configuration of the ground equipment; the virtual channel is determined by the spacecraft data.

2.3 DSN Telemetry Processing

The exact processing performed by the DSN in order to provide telemetry services to a mission is controlled by an agreement between the DSN and the mission. Certain processing must be performed in order for the DSN to provide any telemetry services at all. The minimum required processing is that which is needed to acquire the physical channel (i.e., the stream of digital symbols). Typically, the minimum processing includes, if applicable, demodulation of the RF carrier and subcarrier, symbol synchronization, differential decoding, and convolutional decoding. Other processing is discussed below.

2.3.1 Reed-Solomon Coding

In most cases, the DSN will perform certain additional telemetry processing functions—in particular, frame synchronization and Reed-Solomon decoding—whenever the capabilities of the DSN are compatible with those of the spacecraft.[2] In the most typical case, a spacecraft transmits CCSDS-

[2] Besides frame synchronization and Reed-Solomon decoding, the DSMS may provide additional services to a mission such as source packet re-assembly, data channelization, engineering unit conversion, and level zero processing. However, if any of the latter are provided, they will be provided by means of a different interface than the one described in this module. In such cases, the interface described here may not be exposed to the mission but may be used internally within the DSMS.

compatible frames that are convolutionally and Reed-Solomon encoded. The DSN acquires the physical channel and performs convolutional decoding, frame synchronization, and Reed-Solomon decoding. The result of DSN processing is to produce a stream of decoded, error-free telemetry frames that are delivered to the mission.

It is strongly recommended that—whenever applicable and under ordinary circumstances—the DSN perform frame synchronization and Reed-Solomon decoding on behalf of the mission and that only frames that are synchronized and decoded successfully be delivered to the mission. To support anomalous circumstances and missions with unique requirements, the exact processing performed by the DSN and the characteristics of the DSN telemetry output may be configured differently. For that purpose, the interface specified by this module allows for certain options, as follows:

a) whether or not the DSN performs Reed-Solomon decoding;

b) if the DSN performs Reed-Solomon decoding, whether or not the Reed-Solomon check symbols for a frame are delivered to the mission when Reed-Solomon decoding of that frame is successful;

c) if the DSN performs Reed-Solomon decoding, whether or not frames that cannot be Reed-Solomon decoded are delivered to the mission;

d) whether or not the DSN performs frame synchronization;

e) if the DSN performs frame synchronization, whether or not frames that contain a bit slip are delivered to the mission;

f) if the DSN performs frame synchronization, whether or not bits that cannot be frame synchronized by the DSN are delivered to the mission; and

g) whether or not the DSN performs convolutional decoding.

2.3.2 Turbo Coding

Turbo decoding requires that the frame synchronization be performed before the decoding from symbols to bits is done. It is strongly recommended that—whenever applicable and under ordinary circumstances—the mission use a Cyclical Redundancy Code (CRC) in the transfer frame (the CCSDS optional Frame Error Control field) and that only frames that are synchronized and decoded successfully be delivered to the mission. To support anomalous circumstances and missions with unique requirements, the exact processing performed by the DSN and the characteristics of the DSN telemetry output may be configured differently. For that purpose, the interface specified by this module allows for certain options, as follows:

a) whether or not the DSN performs CRC check;

b) if the DSN performs CRC check, whether or not the frames that fail the CRC check are delivered to the mission; and

c) whether or not frames that cannot be decoded are delivered to the mission as encoded symbols (i.e., the raw data).

2.4 DSN Telemetry SFDU

2.4.1 Telemetry SFDU Structure

The telemetry data acquired by the DSN for delivery to the mission are encapsulated into time-ordered sequences of DSN telemetry SFDUs. A detailed specification of the format and contents of the DSN telemetry SFDU is given in section 3 belog. The following paragraphs provide an overview and general description.

A telemetry SFDU is a sequence of octets that conforms to the SFDU structure and construction rules specified in reference [7] and the detailed format and contents specification given in section 3 belog. Paragraph 1.5.1 abovebriefly describes the SFDU concept. As described there, a telemetry SFDU is an LVO that adheres to the SFDU structure and construction rules specified in reference [7]. It is comprised of an SFDU label field and a value field. Viewed in a strict sense, the telemetry SFDU is a simple SFDU in that its value field contains purely user data; i.e., the value field of the telemetry SFDU is not itself an SFDU or a sequence of SFDUs. However, the value field of the SFDU does have structure in that it contains a sequence of CHDOs, which are a form of LVO (albeit a non-standard, locally defined form). In that sense, the telemetry SFDU is a compound LVO (i.e., its value field contains purely LVOs), and its LVO structure is depicted in Figure 2-1.

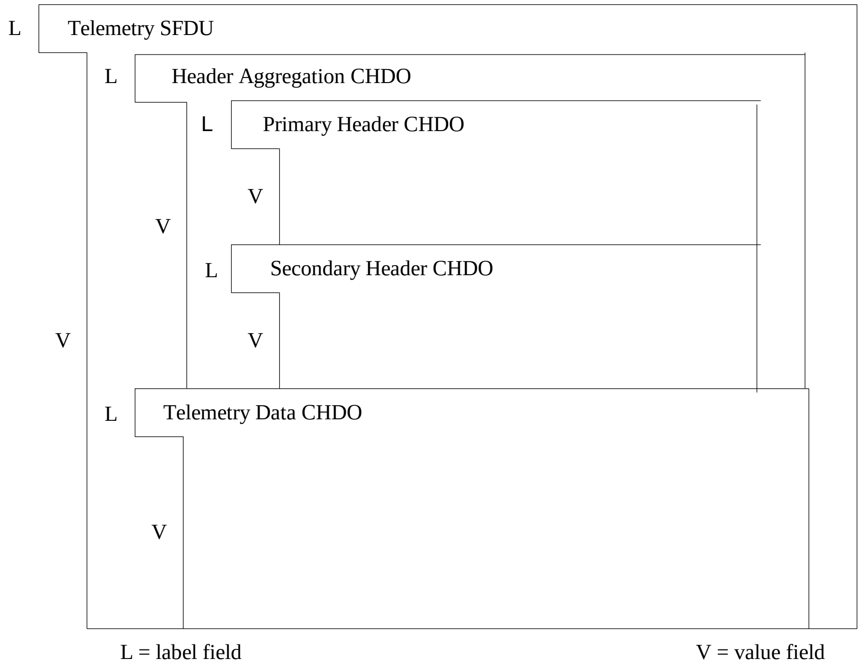

Figure 2-1. LVO Structure of the Telemetry SFDU

Viewed as a compound LVO, the value field of the telemetry SFDU contains two LVOs, an *aggregation CHDO* and a *telemetry data CHDO*. The aggregation CHDO is a compound LVO; its value field contains two simple LVOs, a *primary CHDO* and a *secondary CHDO*. The aggregation CHDO exists solely for the purpose of allowing the primary and secondary CHDOs to be grouped together and treated as a single object. The value fields of the primary and secondary CHDOs contain annotation data (identification, status, configuration, and performance information) that pertain to the telemetry data in the telemetry SFDU. The telemetry data CHDO is a simple LVO; its value field contains the actual received telemetry data from the spacecraft.

2.4.2 Received Telemetry Data

The telemetry data CHDO within a DSN telemetry SFDU contains the actual received telemetry data from the spacecraft. Typically, it contains one frame of decoded telemetry received from the spacecraft. Alternatively, it may contain a length of received telemetry data bits that are not frame synchronized or a length of received telemetry data symbols (quantized to eight-bits). What type of received telemetry is in the telemetry data CHDO depends on the configuration of DSN processing, which is established in accordance with the agreement between the mission and the DSN. It may also depend on the results of DSN processing and the setting of certain processing options (e.g, equipment lock, whether or not the DSN was configured for performing frame synchronization, etc.). An indication of the state of the received telemetry in the telemetry data CHDO is provided by annotation data in the secondary CHDO of the SFDU.

2.4.3 Annotation Data

In addition to the received telemetry from the spacecraft, the telemetry SFDU also contains annotation data, which provides information applicable to the received telemetry contained in that SFDU. For example, annotation data includes the Earth Received Time of the first (or, optionally, the last) bit of telemetry contained in the SFDU. Annotation data is provided in the primary and secondary CHDOs. In general, annotation data identifies the received telemetry contained in the SFDU and provides information about the configuration, status, and performance of the DSN when the received telemetry was acquired. Since some of the status and performance information is the result of measurements integrated over time, its applicability is a best approximation.

The telemetry SFDU specified by this module is intended for use by many different missions; consequently, not all fields in the telemetry SFDU are applicable to all missions. For example, annotation information pertaining to Reed-Solomon decoding is not applicable to a mission that does not use Reed-Solomon encode its telemetry.

2.4.4 Relation To Other Protocol Layers

For consistent processing within the DSN, the telemetry SFDU typically is encapsulated in the user data field of a Standard DSN Block (SDB) as described in reference [3]. An SDB consists of a fixed-length header (sometimes referred to as a DSN Data Delivery [DDD] header), a variable-length user data field (in this case, the telemetry SFDU), and a fixed-length trailer (sometimes referred to as a DDD trailer). In most cases, it is a sequence of SDBs, not bare telemetry SFDUs, that is delivered to the mission. The SDB is defined and controlled by reference [3] and is outside the scope of this module. Some information in the telemetry SFDU duplicates, or essentially duplicates, information in the SDB header. That is a consequence of the design principle that all information required for interpreting and processing the telemetry SFDU should be contained within the telemetry SFDU itself and should not depend on other protocol layers.

Depending on the communications interfaces and protocols employed to deliver telemetry to the mission, the telemetry SFDU (or the SDB in which it is encapsulated) may be encapsulated further, as required by those supporting protocols.

Section 3
Detailed Interface Description

3.1 Telemetry SFDU Physical Layout

The logical structure of the telemetry SFDU is described in paragraph 2.4.1 above. The physical layout is shown in Figure 3-1. The physical layout is divided into five sections: the telemetry SFDU label, the aggregation CHDO label, the primary CHDO, the secondary CHDO, and the telemetry data CHDO. (The primary CHDO and the secondary CHDO together constitute the value field of the aggregation CHDO; the aggregation CHDO and the telemetry data CHDO together constitute the value field of the telemetry SFDU.)

```
       BIT
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
BYTE 0|                                                |
      |                                                |
     ...                  SFDU LABEL                  ...
     ...                  [20 Bytes]                  ...
      |                                                |
    18|                                                |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    20|                                                |
      +             AGGREGATION CHDO LABEL             +
    22|                   [4 Bytes]                    |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    24|                                                |
      |                                                |
     ...                PRIMARY CHDO                  ...
     ...                  [8 Bytes]                   ...
      |                                                |
    30|                                                |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    32|                                                |
      |                                                |
     ...               SECONDARY CHDO                 ...
     ...                  [84 Bytes]                  ...
      |                                                |
   114|                                                |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
   116|                                                |
      |                                                |
     ...             TELEMETRY DATA CHDO              ...
     ...                [(N-116) Bytes]               ...
      |                                                |
   N-2|                                                |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Figure 3-1. Physical Layout of the Telemetry SFDU

The byte and bit numbering conventions used in Figure 3-1 are described in paragraph 1.5.2 above. As shown in Figure 3-1, the length of the telemetry SFDU (in 8-bit bytes) is designated as *N* in this module. In general, the telemetry SFDU is a variable length data structure. In particular applications, it may be possible to constrain the telemetry SFDU always to be fixed in length (depending on the characteristics of the spacecraft and the configuration of DSN processing). Regardless, *N*, the length of the telemetry SFDU, is easily ascertained given the length attribute in the SFDU label (SFDU length attribute = *N*-20). It happens that the only part of the telemetry SFDU that is variable in length is the telemetry data CHDO, so the length of the telemetry SFDU also may be ascertained given the length attribute in the telemetry data CHDO label (telemetry data CHDO length attribute = *N*-120).

Each of the sections of the telemetry SFDU is described in more detail in the following paragraphs.

3.2 Telemetry SFDU Label

Bytes 0 through 19 of the telemetry SFDU (see Figure 3-1) contain the SFDU label field, which is illustrated in Figure 3-2 and defined in [15]. The following paragraphs are provided for reference. The concatenation of bytes 0 to 3 and 8 to 11 constitutes the type attribute of the SFDU. In CCSDS parlance, that concatenated field is known as the Authority and Description Identifier (ADID). Bytes 12 through 19 constitute the length attribute of the SFDU.

```
       BIT
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
BYTE 0| CONTROL AUTHORITY ID = 'NJPL'                 |
      +                                               +
     2|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     4| VERSION ID = '2'      | CLASS ID = 'I'        |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     6| SPARE = '00'                                  |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     8| DATA DESCRIPTION ID = '0800'                  |
      +                                               +
    10|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    12| LENGTH ATTRIBUTE                              |
      +                                               +
    14|                                               |
      +                                               +
    16|                                               |
      +                                               +
    18|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Figure 3-2. Telemetry SFDU Label

<u>BYTES 0 TO 3</u>
BITS

1 thru 8 Control authority identifier; value = ‘NJPL’ (indicates that the data description information for this type of SFDU is maintained and disseminated by NASA/JPL); restricted ASCII. Control authority identifiers are assigned by the CCSDS (see http://www.ccsds.org).

BYTE 4

BITS

1 thru 8 SFDU label version identifier; value = ‘2’ (indicates that the length attribute field in bytes 12 to 19 of the SFDU label is formatted as a binary unsigned integer); restricted ASCII. For additional information on SFDU label versions, see references [7] and [10].

BYTE 5

BITS

1 thru 8 SFDU class identifier; value = ‘I’ (indicates that this SFDU is an application data object); restricted ASCII. For additional information, see reference [7]

BYTES 6 AND 7

BITS

1 thru 8 Reserved; value = ‘00’; restricted ASCII.

BYTES 8 TO 11

BITS

1 thru 8 Data description identifier; value = ‘0800’ (uniquely identifies the data description information maintained for this type of SFDU within the domain of the control authority identified in bytes 0 to 3); restricted ASCII. The value shown here is registered with the identified control authority (i.e., NJPL).

BYTES 12 TO 19

BITS

1 thru 8 Length attribute of the telemetry SFDU (indicates the length, in octets, of the value field of the telemetry SFDU, bytes 20 through *N-1* in Figure 3-1); binary unsigned integer. The length of the value field of the telemetry SFDU is the sum of the total lengths of the aggregation CHDO and the telemetry data CHDO.

3.3 Aggregation CHDO Label

Bytes 20 to 23 of the telemetry SFDU (see Figure 3-1) contain the aggregation CHDO label field, which is illustrated in Figure 3-3 and defined in module 001 of [7]. The following paragraphs are provided for reference. The value field of the aggregation CHDO is composed of the primary CHDO and the secondary CHDO, which are defined in paragraphs 3.4 and 3.5 below.

```
       BIT
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
BYTE 0| TYPE ATTRIBUTE = 1                            |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     2| LENGTH ATTRIBUTE = 92                         |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Figure 3-3. Aggregation CHDO Label

BYTES 0 AND 1

BITS

1 thru 8 Type attribute of the aggregation CHDO; value = 1 (indicates that this CHDO is an aggregation of CHDOs); binary unsigned integer. The NJPL control authority maintains a registry of CHDO type attributes.

BYTES 2 AND 3

BITS

1 thru 8 Length attribute of the aggregation CHDO; value = 92 (indicates the length, in octets, of the value field of the aggregation CHDO, bytes 24 through 115 in Figure 3-1); binary unsigned integer.

3.4 Primary CHDO

Bytes 24 through 31 of the telemetry SFDU (see Figure 3-1) contain the primary CHDO, which is illustrated in Figure 3-4 and defined in module 002 of [7]. The following paragraphs. Bytes 0 through 3 of the primary CHDO are the label field; bytes 4 to 7 are the value field.

```
       BIT
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
BYTE 0| TYPE ATTRIBUTE = 2                            |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     2| LENGTH ATTRIBUTE = 4                          |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     4| MAJOR DATA CLASS = 1  | MINOR DATA CLASS      |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     6| MISSION ID            | FORMAT CODE = 0       |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Figure 3-4. Primary CHDO

BYTES 0 AND 1

BITS

1 thru 8 Type attribute of the primary CHDO; value = 2 (indicates that this CHDO is a primary CHDO); binary unsigned integer. The NJPL control authority maintains a registry of CHDO type attributes.

BYTES 2 AND 3

BITS

1 thru 8 Length attribute of the primary CHDO; value = 4 (indicates the length, in octets, of the value field of the primary CHDO); binary unsigned integer.

BYTE 4

BITS

1 thru 8 Major data class; value = 1 (indicates that this SFDU contains spacecraft telemetry); binary unsigned integer.

BYTE 5

BITS

1 thru 8 Minor data class; value = 7 to 17; binary unsigned integer. This field indicates the results of the processing of the data in this SFDU. Values are used to convey whether the data is frame aligned or not, whether it was derandomized, whether bit level decoding was attempted, or whether any processing was done. Also, the type of output (bits or symbols) for turbo decoding is indicated. Note that frame aligned implies that frame synchronizer state is either flywheel, verify, or in-lock and that non-framed aligned implies the state is search mode.

The meanings of the minor data class values depend on the type of data. Values 7 to 11 are for non-turbo encoded data. Values 12 to 16 are for turbo encoded data. Value 17 is for data that has no processing done; the raw, undecoded eight-bit bytes are just forwarded on. The value meanings are provided in Table Section 3–1. The possible minor class values versus the different configuration options are given in Table Section 3–2.

Table Section 3–1. Minor Data Class Definitions

Class	Type of Processing	Value Meaning
7	Non-turbo	Non-frame aligned data
8	Non-turbo	Frame aligned data, no further processing

Class	Type of Processing	Value Meaning
9	Non-turbo	Frame aligned data, derandomized, no further processing
10	Non-turbo	Frame aligned data, derandomized, decoding attempted
11	Non-turbo	Frame aligned data, decoding attempted
12	Turbo	Decoded bits, passed CRC (if CRC used)
13	Turbo	Decoded bits, failed CRC
14	Turbo	Bits, failed to decode
15	Turbo	Eight-bit symbols, failed to decode
16	Turbo	One-bit symbols, no decoding attempted (AMMOS test mode)
17	No Processing	Eight-bit symbols, no processing done.

Table Section 3–2. Minor Class Values versus Configuration

Possible Minor Class States	**Frame Sync**	**Reed Solomo n**	**CR C**	**P N**	**Turbo**
7, 16, 17	D	D	D	D	D
7, 8	E	D	D	D	D
7, 11	E	E	D	D	D
7, 10	E	E	D	E	D
7,11	E	E	E	D	D
7, 10	E	E	E	E	D
7,11	E	D	E	D	D
7, 10	E	D	E	E	D
7,9	E	D	D	E	D
12, 14, 15	E	D	D	D	E
12, 14, 15	E	D	D	E	E
12, 13, 14, 15	E	D	E	D	E
12, 13, 14, 15	E	D	E	E	E

D => Function Disabled
E => Function Enabled

BYTE 6

BITS

1 thru 8 Mission identifier; value range 0–255 (identifies the mission whose telemetry is contained in this SFDU); binary unsigned integer. Mission identifiers are defined and controlled by reference [5]. Not all missions are assigned a mission identifier; in those cases, the value of this field shall be 254. Mission identifier is to be distinguished from spacecraft identifier (see 3.5 below). In general, a spacecraft identifier is associated with at most one mission identifier, but a mission identifier may have several spacecraft identifiers associated with it.

BYTE 7

BITS

1 thru 8	Format code; value = 0 (indicates this is a DSS-formatted SFDU); binary unsigned integer.

3.5 Secondary CHDO

Bytes 32 through 115 of the telemetry SFDU (see Figure 3-1) contain the secondary CHDO, which is illustrated in Figure 3-5 and defined in the following paragraphs. Bytes 0 to 3 of the secondary CHDO are the label field; bytes 4 through 83 are the value field.

```
       BIT
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
BYTE 0| TYPE ATTRIBUTE = 78                           |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     2| LENGTH ATTRIBUTE = 80                         |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     4| ORIGINATOR ID = 48    | LAST MODIFIER ID = 48 |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     6| RESERVED        | SPACECRAFT ID               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     8| PASS NUMBER                                   |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    10| DATA SOURCE ID        | ARRAYED STATIONS      |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    12| FLAGS                                         |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    14| EARTH RECEIVED TIME                           |
      +                                               +
    16|                                               |
      +                                               +
    18|                                               |
      +                                               +
    20|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    22| RECORD SEQUENCE NUMBER                        |
      +                                               +
    24|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    26| U/L BAND              | D/L BAND              |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    28| PDX MODE              | U/L SOURCE            |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    30| VIRTUAL STREAM ID     | VIRTUAL CHANNEL ID    |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    32| TELEMETRY LOCK STATUS                         |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    34| NUMBER OF BITS                                |
      +                                               +
    36|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    38| MEASURED BIT RATE                             |
      +                                               +
    40|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    42| SYSTEM NOISE TEMPERATURE                      |
      +                                               +
    44|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    46| SYMBOL SNR                                    |
      +                                               +
    48|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    50| RECEIVER SIGNAL LEVEL                         |
      +                                               +
    52|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    54| ACQUISITION BET       | MAINTENANCE BET       |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    56| VERIFY COUNT          | FLYWHEEL COUNT        |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    58| FRAME SYNC MODE FLAGS | SYNC STATUS  |BIT SLIP|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    60| FS ERROR COUNT        | FS BUFFER SIZE        |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    62| RS DECODER STATUS     | RS SYMBOL ERROR COUNT |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    64| TURBO DECODER STATUS  | PROCESSOR NUMBER      |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    66| NUMBER OF ITERATIONS  | RESERVE               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    68| CODE RATE NUMERATOR   | CODE RATE DENOMINATOR |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    70| FRAME DATA BITS                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    72| DECODER CONFIDENCE                            |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    74| TELEMETRY EQUIPMENT ID                        |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    76| TELEMETRY SOFTWARE ID                         |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    78| RESERVED                                      |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    80| RESERVED                                      |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
    82| RESERVED                                      |
```

Figure 3-5. Secondary CHDO

BYTES 0 AND 1

BITS

1 thru 8 Type attribute of the secondary CHDO; value = 78 (indicates that this CHDO is a telemetry secondary CHDO); binary unsigned integer. The NJPL control authority maintains a registry of CHDO type attributes.

BYTES 2 AND 3

BITS

1 thru 8 Length attribute of the secondary CHDO; value = 80 (indicates the length, in octets, of the value field of the secondary CHDO); binary unsigned integer.

BYTE 4

BITS

1 thru 8 Originator identifier; value = 48 (indicates that this SFDU originated within the DSN); binary unsigned integer.

BYTE 5

BITS

1 thru 8 Last modifier identifier; value = 48 (indicates that the contents of this SFDU were last modified by the DSN); binary unsigned integer.

BYTES 6 AND 7

BITS

1 thru 6 Reserved

7 thru 16 DSN-assigned spacecraft identifier as per reference [5]; value range 0–1023; binary unsigned integer. The DSN-assigned spacecraft identifier, which is used here and at other places within the NASA ground data system, is different from the CCSDS-assigned spacecraft identifier, which is used, for example, in the headers of transfer frames and VCDUs. For a given spacecraft, the assigned values for the two spacecraft identifiers may or may not be the same.

BYTES 8 AND 9

BITS

1 thru 8 Pass number for this data; value range 0 to 9999, binary unsigned integer. Value of 0 indicates pass number was not available.

BYTE 10

BITS

1 thru 8 Data source identifier (identifies the DSS used to acquire the telemetry in this SFDU); value range 0–255; binary unsigned integer. Data source identifiers (also known as DSN Location Identifiers or DSN Antenna/Facility Identifiers) are defined and controlled by reference [4]; an example value would be 15 for "34M HEF - Goldstone."

BYTE 11

BITS

1 thru 8 Arrayed stations; comprises eight one-bit subfields. The subfields indicate whether or not the selected antenna type is included in the array (0 => not in array, 1 => in the array). Table Section 3–3 specifies the one-bit subfields and the antenna type each refers to. Valid only if byte 13, bit 5 is one.

Table Section 3–3. Arrayed Stations Subfields

Subfield	Antenna Type
Bit 1	70m (e.g., DSS-14, 43, 63)
Bit 2	HEF (e.g., DSS-15, 45, 65)
Bit 3	BWG1 (e.g., DSS-24, 34, 54)
Bit 4	BWG2 (e.g., DSS-25, 55)
Bit 5	BWG3 (e.g., DSS-26)
Bit 6	26m (e.g., DSS-16, 46, 66)
Bit 7	HSB1 (e.g., DSS-27)
Bit 8	HSB2 (e.g., DSS-28)

BYTE 12

BITS

1 Reserved for AMMOS usage.

2 DTT QPSK Mode; the value is 0 (if the DTT is being operated in the standard DSN QPSK mode in which the I and Q channels are interleaved and processed as one physical channel) or 1 (if the DTT is being operated in the split QPSK mode in which a given DTT processes only the I channel or only the Q channel but not both) (1=Odd, 0=Even); binary unsigned integer.

3 DTT QPSK Configuration; this field is meaningful only if the DTT is being operated in the split QPSK mode (that is, only if bit 2 is 1); if bit 2 is 0, then bit 3 shall be ignored; if bit 2 is 1 and bit 3 is 1, then the received telemetry bits in this SFDU comprise the I or Q channel that includes the first bit of the transfer frame attached sync marker ("ODD half-frame"); if bit 2 is 1 and bit 3 is 0, then the received telemetry bits in this SFDU comprise the I or Q channel that includes the second bit of the transfer frame attached sync marker ("EVEN half-frame"); binary unsigned integer.

4 MCD sync status change; 0 => no change or MCD is not used, 1 => sync change; binary unsigned integer. Set to 1 if a sync change was detected anytime during the period that the data in the block were accumulated.

5 Earth Received Time (ERT) reference point; value = 0 (if the ERT refers to the trailing edge of the last received telemetry bit in the SFDU) or 1 (if the ERT refers to the leading edge of the first received telemetry bit in the SFDU); binary unsigned integer.

6 ERT extended resolution field (see 1.5.4 above) status; value = 0 (if the ERT extended resolution field is reserved and its value should be ignored) or 1 (if the ERT extended resolution field is valid); binary unsigned integer.

7 ERT extended resolution field (see 1.5.4 above) units; value = 0 (if the units of the extended resolution field is microseconds [of millisecond]) or 1 (if the units of the extended resolution field is tenths-of-microseconds [of millisecond]); binary unsigned integer.

8 ERT status; value = 0 (if ERT is valid) or 1 (if ERT is known to be invalid); binary unsigned integer.

BYTE 13
BITS

1 CRC check mode; value = 0 (disabled) or 1 (enabled); binary unsigned integer.

2 System noise temperature (SNT) measurement flag; value = 0 (if the SNT in bytes 42 to 45 of the secondary CHDO is measured) or 1 (if the SNT is not measured); binary unsigned integer.

3	CRC check status flag; value = 0 (failed CRC check) or 1 (passed CRC check); binary unsigned integer. This is only valid if the system is configured to perform a CRC check. This field is meaningless and ignored otherwise.
4	Pseudo-derandomizer flag; value is to zero if no pseudo-derandomization is done and is one if pseudo-derandomization is performed; binary unsigned integer.
5	Array status; value = 0 (not arrayed) or 1 (arrayed); binary unsigned integer. This indicates whether the data is from an array of antennas.
6	Signal-to-Noise Ratio (SNR) type; value = 0 (if the SNR in bytes 46 to 49 of the secondary CHDO is estimated in the symbol domain) or 1 (if the SNR is estimated in the bit domain); binary unsigned integer.
7	Low threshold telemetry processing flag; value = 0 (normal processing) or 1 (low threshold processing); binary unsigned integer. Low threshold telemetry processing is a method of using non-causal processing of low data rates to lower the detection threshold.
8	DSN Diagnostic mode; value = 0 (if diagnostic mode is disabled) or 1 (if diagnostic mode is enabled); binary unsigned integer. This bit should always be zero in data delivered to a mission.

<u>BYTES 14 TO 21</u>

BITS

1 thru 8	Earth received time (ERT); the format of this field is defined in paragraph 1.5.4 above. ERT is the UTC time that the telemetry data contained in this SFDU were acquired by the DSS. For most missions, ERT is defined as the time that the telemetry signal event corresponding to the trailing edge of the last bit of telemetry data contained in this SFDU was received at the input to the DSS low noise amplifier. Optionally, for some missions, ERT may be defined as the time that the telemetry signal event corresponding to the leading edge of the first bit of telemetry data contained in this SFDU was received at the input to the DSS low noise amplifier. Byte 12, bit 5 determines which option is selected. For most missions, ERT is reported with a resolution of one millisecond and an accuracy of plus or minus one millisecond. The agreement between the DSN and the mission may require a finer ERT resolution and accuracy. Byte 12, bit 6 indicates if the extended resolution is valid. Byte 12, bit 7 defines the extended resolution units and Byte 12, bit 8 informs of the validity of the field.

<u>BYTES 22 TO 25</u>

BITS

1 thru 8	Record sequence number (RSN); binary unsigned integer. The RSN is a sequence counter for the telemetry SFDUs in a given virtual stream. It starts at one and increments by one for each successive telemetry SFDU in the stream. It wraps around from 2^{32}–1 to zero. Separate sequences are maintained for each

virtual stream. A virtual stream is defined as the ERT-ordered sequence of all telemetry SFDUs with the same global virtual stream identifier. The global virtual stream identifier is defined as the concatenation of the following fields from the secondary CHDO: spacecraft identifier (byte 6, bits 7-8, and byte 7, bits 1-8), data source identifier (byte 10, bits 1-8), telemetry equipment ID (bytes 74 and 75, bits 1-8), and virtual stream identifier (byte 30, bits 1-8). The RSN may reset to one at any time (e.g., whenever the DSS telemetry processor is started or restarted); however, such resets should be infrequent. The RSN is provided by the originator of the telemetry SFDU and shall not be changed subsequently; e.g., a telemetry SFDU replayed from a recording shall retain the RSN that was assigned when the SFDU was created.

BYTE 26

BITS

1 thru 8 Uplink frequency band; value = U, S, X, or K (indicates Unknown, S-band, X-band, or Ka-band, respectively); restricted ASCII. Only valid when predicts mode is two- or three-way (byte 28, bits 7 and 8 equal to 2 or 3).

BYTE 27

BITS

1 thru 8 Downlink frequency band; value = U, S, X, or K (indicates Unknown, S-band, X-band, or Ka-band, respectively); restricted ASCII.

BYTE 28

BITS

1 thru 6 Reserved.

7 and 8 Predicts mode (0 => no predicts used, 1 => one-way, 2 => two-way, 3 => three-way); value range 0 to 3; binary unsigned integer.

BYTE 29

BITS

1 thru 8 Uplink station number for three-way predicts (Predicts mode equal to 3); value range 0-255; binary unsigned integer. Uplink station numbers (also known as DSN Location Identifiers or DSN Antenna/Facility Identifiers) are defined and controlled by reference [4]; an example value would be 15 for “34M HEF - Goldstone.”

BYTE 30

BITS

1 thru 8 Virtual stream identifier; value range 0–255; binary unsigned integer. (Refer to paragraph 2.2 above for details about the virtual stream.) Note that virtual stream numbers 126 and 127 are reserved for the 26m Telemetry and Command Processor time-out and keep alive blocks, respectively.

BYTE 31

BITS

1 thru 8 Virtual channel identifier; value range 0–255; binary unsigned integer. (Refer to paragraph 2.2 above for details about the virtual stream.) The virtual channel ID is obtained from the spacecraft data.

BYTES 32 AND 33

BITS

1 thru 8 Telemetry lock status code; comprised of eight two-bit subfields. The subfields indicate the lock status of selected telemetry processes (if applicable to the configuration in use). Table Section 3–4 specifies the two-bit subfields and the telemetry process each refers to. Table Section 3–5 specifies the meaning of the lock status codes that may be contained in each subfield.

Table Section 3–4. Telemetry Lock Status Subfields

Subfield	Telemetry Process
Byte 32, Bits 1-2	Carrier demodulation
Byte 32, Bits 3-4	Array
Byte 32, Bits 5-6	Subcarrier demodulation
Byte 32, Bits 7-8	Symbol synchronization
Byte 33, Bits 1-2	Convolutional decoding
Byte 33, Bits 3-4	Frame synchronization
Byte 33, Bits 5-6	Reed-Solomon decoding
Byte 33, Bits 7-8	Turbo decoder

Table Section 3–5. Telemetry Lock Status Codes

Code	Interpretation
00_2	Not in use or lock status unknown
01_2	(Invalid code)
10_2	In lock
11_2	Out of lock

BYTES 34 TO 37

BITS

1 thru 8 Number of bits; indicates the number of bits that are contained in the value field of the telemetry data CHDO in this SFDU; binary unsigned integer. If the number of received telemetry bits is less than the length of the value field of the telemetry data CHDO, unused bits appear at the end of the field and shall be ignored.

For turbo encoded data, if the frame is successfully decoded, the received bits are output. If the frame is not successfully decoded, and the system is so configured, the undecoded symbols are delivered in the telemetry data CHDO. In this case, the number of bits would be the frame size times 8 (eight-bit quantization) divided by the coding rate. Byte 64, bit 8 indicates which type of data is in the telemetry data CHDO.

BYTES 38 TO 41

BITS

1 thru 8 Measured bit rate of the received telemetry data, in bits/second (with an accuracy of 0.1 bits/second); value range 2.0 to 13.2e6; 32-bit floating point format (refer to paragraph 1.5.3 above for format details).

BYTES 42 TO 45

BITS

1 thru 8 System noise temperature (in Kelvin with an accuracy of 0.1 Kelvin); value range 10.0 to 2000.0; 32-bit floating point format (refer to paragraph 1.5.3 above for format details).

BYTES 46 TO 49

BITS

1 thru 8 Estimated signal-to-noise ratio (in dB) in the symbol domain (i.e., prior to any decoding); value range -10.0 to 40.0; 32-bit floating point format (refer to paragraph 1.5.3 above for format details). If byte 13, bit 6 equals one, then the

estimate is in the bit domain (post decoding). Valid only if carrier, subcarrier, and symbol demodulators are in-lock (bytes 32 and 33).

BYTES 50 TO 53

BITS

1 thru 8 Receiver signal level (in dBm); value range -190.0 to -85.0; 32-bit floating point format (refer to paragraph 1.5.3 above for format details). Valid only if carrier demodulator is in-lock (byte 32). Depending on the receiver configuration, the value will either be the carrier power level or the data power level.

BYTE 54

BITS

1 thru 8 Acquisition bit error tolerance (BET); number of allowed bit errors in the attached sync marker during search and verify modes; value range 0–31; binary unsigned integer.

BYTE 55

BITS

1 thru 8 Maintenance BET; number of allowed bit errors in the attached sync marker during lock and flywheel modes; value range 0–31; binary unsigned integer.

BYTE 56

BITS

1 thru 8 Verify count; number of within-tolerance frames required, during verify mode, to transition to lock mode; value range 0–31; binary unsigned integer.

BYTE 57

BITS

1 thru 8 Flywheel count; number of out-of-tolerance frames required, during flywheel mode, to transition to search mode; value range 0–31; binary unsigned integer.

BYTE 58

BITS

1 thru 8 Frame sync mode flags; as shown in Table Section 3–6, the specified condition is true if the corresponding bit is set equal to one.

Table Section 3–6. Frame Sync Mode Flags

Bit	Interpretation If Bit = 1
1	Operator forced resynchronization
2	Reserved
3	Automatic polarity correction (APC) is enabled
4	Frame synchronizer is in flywheel mode
5	Frame synchronizer is in lock mode
6	Frame synchronizer is in verify mode
7	Frame synchronizer is in search mode
8	Frame synchronizer is in bypass mode

The following qualifications apply to the frame sync mode flags:

a) The bypass mode flag (bit 8) is set to one if and only if the DSS frame synchronizer is configured to bypass mode (i.e., frame synchronization is disabled).

b) When the bypass mode flag (bit 8) is set to one, bits 4-7 of this byte are meaningless and should be ignored.

c) When the bypass mode flag (bit 8) is set to zero, one and only one of bits 4-7 will be set to one (i.e., frame synchronization is enabled, and the frame synchronizer is in the specified mode).

d) When the frame synchronizer is in bypass mode or search mode, all of bytes 59 to 63 are meaningless and should be ignored.

e) When the frame synchronizer is in bypass mode or search mode, the value field of the telemetry data CHDO contains a length of telemetry bits that is not frame aligned. When the frame synchronizer is in verify mode, flywheel mode, or lock mode, the value field of the telemetry data CHDO contains a length of telemetry bits that has been identified as being frame aligned by the frame synchronizer.

f) When APC is enabled (bit 3 = 1) and the frame synchronizer is in verify, lock, or flywheel mode, an attempt is made to resolve the ambiguity between true and complemented data. If APC is enabled and Byte 59, bit 1, is set to one, it indicates that the frame synchronizer inverted all bits in the frame to achieve true data polarity.

g) For turbo encoded data, bits 4, 6, 7, and 8 are meaningless.

BYTE 59

BITS

1 Data polarity flag; indicates the data polarity of the attached sync marker detected by the frame synchronizer. The flag is set to zero if the true sync marker is detected; it is set to one if the complemented sync marker is detected. If the complemented sync marker is detected and APC is enabled, the frame synchronizer inverts all bits in the frame. This field is meaningless and should be ignored if the frame synchronizer is in bypass or search mode.

2 Sync marker in block flag; indicates whether or not the attached sync marker is included in the data block provided in this frame. The flag is set to zero if the sync marker is included in the data; it is set to one if the sync marker is not included in the data. This field is meaningless and should be ignored if the frame synchronizer is in bypass or search mode.

3 thru 5 Reserved.

6 thru 8 Bit slip status code; set as indicated in Table Section 3–7. This field is meaningless and should be ignored if the frame synchronizer is in bypass mode or search mode. Due to signal disturbances and ground processing artifacts, the frame synchronizer may occasionally emit a frame that is a few bits longer or shorter than the nominal frame length. This condition is termed *bit slip* (or *symbol slip* if turbo encoded data) and is indicated by a non-zero code in this field. When this field is non-zero, it indicates that the value field of the telemetry data CHDO contains a telemetry frame that is longer or shorter than the nominal frame length. When this field is zero, it indicates that the value field of the telemetry data CHDO contains a frame of nominal length. The 26m Telemetry and Command Processor does not provide this measurement (value is always 0). The type of telemetry equipment used is provided in bytes 74 and 75.

Table Section 3–7. Bit Slip Status Codes

Code	Interpretation
011_2	Frame is three bits longer than a nominal frame
010_2	Frame is two bits longer than a nominal frame
001_2	Frame is one bit longer than a nominal frame
000_2	Frame equals a nominal frame in length
111_2	Frame is one bit shorter than a nominal frame
110_2	Frame is two bits shorter than a nominal frame
101_2	Frame is three bits shorter than a nominal frame

BYTE 60

BITS

1 thru 8 Attached sync marker error count; indicates the number of bit errors (or symbol errors, if turbo encoded) in the attached sync marker detected by the frame synchronizer; binary unsigned integer. This field is meaningless and should be ignored if the frame synchronizer is in bypass mode or search mode. Note that this value cannot exceed the acquisition bit error tolerance (see Byte 54) and takes into account the data polarity (see Byte 59).

BYTE 61

BITS

1 thru 4 Reserved.

5 thru 8 Frame synchronizer buffer size; value range 0 to 15; unsigned binary integer. Indicates the size of the buffering, in number of frames, done by the frame synchronization process. The buffering contributes to the latency of the system. A value of 0 indicates that no buffering is done.

BYTE 62

BITS

1 Parity bits included flag; indicates whether or not the Reed-Solomon parity bits are included in the data of the block. A value of zero indicates that the bits are included; a value of one indicates that the parity bits are not included. This field is meaningless and should be ignored if the frame synchronizer is in bypass mode or search mode or if turbo decoding is enabled.

2 thru 4 Reserved.

5 thru 8 Reed-Solomon decoder status code; value = 0, 1, 2, or 3 (indicates the Reed-Solomon decoder status, as defined in Table Section 3–8); binary unsigned integer. This field is meaningless and should be ignored if the frame synchronizer is in bypass mode or search mode or if turbo decoding is enabled.

Table Section 3–8. Reed-Solomon Decoder Status Codes

Code	Interpretation
0	No attempt was made to Reed-Solomon decode the contents of the value field of the telemetry data CHDO. The value field contains the unmodified output of the frame synchronizer.
1	The value field of the telemetry data CHDO contains an attached sync marker and a transmitted codeblock. The codeblock is a Reed-Solomon codeblock that was received without errors (i.e., no symbol errors were detected during Reed-Solomon decoding).
2	The value field of the telemetry data CHDO contains an attached sync marker and a transmitted codeblock. The codeblock is a Reed-Solomon codeblock that was received with errors but was corrected by Reed-Solomon decoding.
3	An attempt was made to interpret the contents of the value field of the telemetry data CHDO as an attached sync marker and a transmitted codeblock, but the candidate codeblock is NOT a Reed-Solomon codeblock, nor could it be corrected to be a Reed-Solomon codeblock. The value field of the telemetry data CHDO contains the unmodified output of the frame synchronizer.

<u>BYTE 63</u>

BITS

1 thru 8 Number of corrected Reed-Solomon symbol errors in this frame; value range is 0-80; binary unsigned integer. This field is valid only when the DSS Reed-Solomon decoder status (byte 62, bits 1-8) is 1 or 2.

<u>BYTE 64</u>

BITS

1 thru 5 Reserved.

6 Turbo decoder extra bits flag; indicates whether an equivalent 32 bit sync marker and the four trellis termination bits are included in the data block or not. A value of zero indicates that the extra bits are not included; a value of one indicates that a 32 bit sync marker is attached before the data bits and the 4 trellis termination bits are appended to the end of the data bits. This field is meaningless and ignored if data is not turbo encoded.

7 Turbo decoder success flag; value = 0 (decoder failed) or 1 (decoder successful); binary unsigned integer. This is an indication by the decoder whether or not it successfully decoded the frame. This field is meaningless and ignored if data is not turbo encoded.

8 Output type flag; value = 0 (output is bits from the turbo decoder) or 1 (output is the undecoded 8-bit symbols input to the turbo decoder); binary unsigned integer. This indicates the format of the data in the telemetry CHDO. This field is meaningless and ignored if data is not turbo encoded.

<u>BYTE 65</u>

BITS

1 thru 3 Reserved.

4 thru 8 Processor number; value range 1-31; unsigned binary integer. This indicates which of the decoder elements decoded this frame.

<u>BYTE 66</u>

BITS

1 thru 8 Number of iterations; value range 1-255; unsigned binary integer. This is the number of iterations used by the decoder to decode this frame.

<u>BYTE 67</u>

BITS

1 thru 8 Reserved.

<u>BYTE 68</u>

BITS

1 thru 8 Turbo code rate numerator; value range 1-31; unsigned binary integer. This is the numerator in the coding rate ratio (bits to symbols).

<u>BYTE 69</u>

BITS

1 thru 8 Turbo code rate denominator; value range 1-31; unsigned binary integer. This is the denominator in the coding rate ratio (bits to symbols).

<u>BYTES 70 AND 71</u>

BITS

1 thru 8 Turbo frame size in bits; value range 1 to 2^{16}-1; unsigned binary integer. This is the number of information bits in the turbo encoded frame. Currently, the only values allowed are 1784, 3568, 7136, and 8920.

BYTES 72 AND 73

BITS

1 thru 8 Decoder confidence value; value range 0 to 2^{16}-1; unsigned binary integer. This value reflects the "confidence" of the decoding result. It is used to determine whether or not the decoder declares the frame successfully decoded.

BYTES 74 AND 75

BITS

1 thru 8 Telemetry equipment identifier (identifies the telemetry equipment (receiver and telemetry processor), within the facility indicated by the data source identifier, used to acquire the data in this SFDU); binary unsigned integer. Value is equipment dependent; definition of values is provided in Appendix A.

BYTES 76 AND 77

BITS

1 thru 8 Telemetry software identifier; restricted ASCII. Identifies the telemetry software version in use. Value is equipment dependent; definition of values is provided in Appendix A.

BYTES 78 TO 83

BITS

1 thru 8 Reserved.

3.6 Telemetry Data CHDO

Bytes 116 through *N-1* of the telemetry SFDU (see Figure 3-1) contain the telemetry data CHDO, which is illustrated in Figure 3-6 and defined in module 010 of [7]. The following paragraphs are provided for reference. Bytes 0 to 3 of the telemetry data CHDO are the label field; Bytes 4 through (*N* - 117) are the value field.

```
       BIT
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
BYTE 0| TYPE ATTRIBUTE = 10                           |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     2| LENGTH ATTRIBUTE                              |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
     4| RECEIVED TELEMETRY BITS                       |
      +                                               +
      |                                               |
     ...                                             ...
     ...                                             ...
      |                                               |
      +                                               +
 N-118|                                               |
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
      |1 |2 |3 |4 |5 |6 |7 |8 |9 |10|11|12|13|14|15|16|
      +--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

Figure 3-6. Telemetry Data CHDO

<u>BYTES 0 AND 1</u>

BITS

1 thru 8 Type attribute of the telemetry data CHDO; value = 10 (indicates that this CHDO contains binary data); binary unsigned integer.

<u>BYTES 2 AND 3</u>

BITS

1 thru 8 Length attribute of the telemetry data CHDO (indicates the length, in octets, of the value field of the telemetry data CHDO); binary unsigned integer. The length attribute shall always be an even number. The telemetry data CHDO length attribute only indicates the length of the value field of the telemetry data CHDO; it does not indicate the number of received telemetry bits that are contained within the value field. The actual number of received telemetry bits that are contained in the value field is given by bytes 34 to 37 of the secondary CHDO. The actual number of received telemetry bits may be less than the number of bits in the field; any unused bits appear at the end of the field and shall be ignored.

<u>BYTES 4 THRU (*N* - 117)</u>

BITS

1 thru 8 Received telemetry bits; contains spacecraft telemetry data acquired by the DSS; variable-length bit string. The number of received telemetry bits contained in this field (the length of the variable-length bit string) is given by bytes 34 to 37 of the secondary CHDO. This field may contain a telemetry frame from the spacecraft or it may contain undecoded symbols (which is indicated by either secondary CHDO byte 64, bit 8, or the minor data class indicator, byte 5, of the primary CHDO).

Appendix A
Hardware Specific Values

A.1 Hardware Issues

There are three distinct sets of hardware used for telemetry reception in the DSN. The two fields, Telemetry Equipment Identifier and Telemetry Software Identifier are used to annotate the telemetry SFDU with the specific hardware and software used in generating the SFDU. This section defines how those fields are used for each type of configuration.

The three sets of hardware are:

1. Block V Receiver (BVR) with Telemetry Channel Assembly (TCA) - The BVR is specified by the Receiver Channel Processor (RCP) number. The TCA is identified by the telemetry group number and channel number inside the group.

2. Multifunction Receiver (MFR) with Telemetry and Command Processor - The MFR is specified by its number. The Telemetry and Command Processor is specified by its number. This equipment is only at the 26m subnet.

3. Downlink Channel (DC) - This is the replacement for the BVR-TCA. It is a receiver and telemetry processor integrated together and is specified by its number.

A.1.1 Telemetry Equipment Identifier

For the BVR-TCA combination, the bits are defined as follows:

<u>BYTE 0</u>

BITS

1 thru 4 Indicates the type of equipment; value = 0; unsigned binary integer.

5 thru 8 Reserved.

<u>BYTE 1</u>

BITS

1 thru 4 RCP number minus 1; value = 0 to 15; unsigned binary integer.

5 thru 7 Telemetry Group number minus 1; value = 0 to 5; unsigned binary integer.

8 TCA number minus 1; value = 0 to 1; unsigned binary integer.

For the MFR-Telemetry and Command Processor combination, the bits are defined as follows:

<u>BYTE 0</u>

BITS

1 thru 4 Indicates the type of equipment; value = 1; unsigned binary integer.

5 thru 8 Reserved.

<u>BYTE 1</u>

BITS

1 thru 4 MFR number minus 1; value = 0 to 2; unsigned binary integer.

5 thru 8 Telemetry and Command Processor number minus 1; value = 0 to 1; unsigned binary integer.

For the DC, the bits are defined as follows:

<u>BYTE 0</u>

BITS

1 thru 4 Indicates the type of equipment; value = 2; unsigned binary integer.

5 thru 8 Reserved.

<u>BYTE 1</u>

BITS

1 thru 2 Full spectrum processor number; value = 0 to 2 (zero indicates no arraying); unsigned binary integer. Note, byte 13, bit 5 of the secondary CHDO indicates whether or not arraying is used.

3 thru 4 Reserved.

5 thru 8 DC number minus 1; value = 0 to 15; unsigned binary integer.

A.1.2 Telemetry Software Identifier

Software versions are identified by an operational level letter (e.g., "OP-A") and a version number, expressed in the form X.Y.Z. The Telemetry Software Identifier provides the software identity of the telemetry processing equipment as follows:

<u>BYTE 0</u>

BITS

1 thru 8 Software operational level; value = ‘A’ to ‘Z’; restricted ASCII.

<u>BYTE 1</u>

BITS

1 thru 8 Revision level; value = 0 to 255; unsigned binary integer. This is the value ‘Y’ in the version number ‘X.Y.Z’.