The Consultative Committee for Space Data Systems

Draft Recommendation for Space Data System Standards

LOSSLESS DATA COMPRESSION

DRAFT RECOMMENDED STANDARD

CCSDS 121.0-P-2.1

PINK SHEETS

November 2019

The Consultative Committee for Space Data Systems

Draft Recommendation for Space Data System Standards

LOSSLESS DATA COMPRESSION

DRAFT RECOMMENDED STANDARD

CCSDS 121.0-P-2.1

PINK SHEETS

November 2019

DOCUMENT CONTROL

Document	**Title**	**Date**	**Status**
CCSDS 121.0-B-1	Lossless Data Compression, Blue Book, Issue 1	May 1997	Original issue, superseded
CCSDS 121.0-B-2	Lossless Data Compression, Recommended Standard, Issue 2	May 2012	Current issue
CCSDS 121.0-P-2.1	Lossless Data Compression, Draft Recommended Standard, Issue 2.1	November 2019	Current draft update: addresses potential ambiguities and adds option for a file header.

NOTE – Only pages containing changes to the current issue are included in this review document.

1 INTRODUCTION

1.1 PURPOSE

The purpose of this document is to establish a Recommended Standard for a source-coding data-compression algorithm applied to digital data and to specify how these compressed data shall be inserted into ~~source~~space packets and files for retrieval and decoding.

Source coding for data compression is a method utilized in data systems to reduce the volume of digital data to achieve benefits in areas including, but not limited to,

a) reduction of transmission channel bandwidth;

b) reduction of the buffering and storage requirement;

c) reduction of data-transmission time at a given rate.

1.2 SCOPE

The characteristics of source codes are specified only to the extent necessary to ensure multi-mission support capabilities. The specification does not attempt to quantify the relative bandwidth reduction, the merits of each approach discussed, or the design requirements for coders and associated decoders. Some performance information is included in reference [B2].

This Recommended Standard addresses only Lossless source coding, which is applicable to a wide range of digital data, both imaging and non-imaging, where the requirement is for a moderate data-rate reduction constrained to allow no distortion to be added in the data compression/decompression process. The decompression process is not addressed. See reference [B2] for an outline of an implementation.

1.3 APPLICABILITY

This Recommended Standard applies to data compression applications of space missions anticipating packetized telemetry cross support. In addition, it serves as a guideline for the development of compatible CCSDS Agency standards in this field, based on good engineering practice.

1.4 RATIONALE

The concept and rationale for the Lossless source coding for data compression algorithm described herein may be found in reference [B2].

~~1.5 BIT NUMBERING CONVENTION AND NOMENCLATURE~~

1.5 CONVENTIONS AND DEFINITIONS

1.5.1 MATHEMATICAL NOTATIONS AND DEFINITIONS

In this document, for any real number x, the largest integer n such that $n \le x$ is denoted by

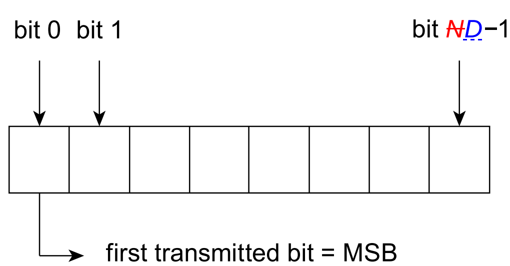
$$n = \lfloor x \rfloor$$

The modulus of an integer M with respect to a positive integer divisor n, denoted $M \bmod n$ is defined to be

$$M \bmod n = M - n\lfloor M / n \rfloor$$

1.5.2 CONVENTIONS

In this document, the following convention is used to identify each bit in ~~an N~~a D−bit word. The first bit in the word to be transmitted (i.e., the most left justified when drawing a figure) is defined to be ‘Bit 0’, the following bit is defined to be ‘Bit 1’, and so on up to ‘Bit ~~N~~D−1’. When the word is used to express an unsigned binary value (such as a counter), the Most Significant Bit (MSB) shall correspond to the highest power of two, i.e. ~~2^{N-1}~~2^{D-1} .



In accordance with modern data communications practice, spacecraft data words are often grouped into 8-bit ‘words’ which conform to the above convention. Throughout this Recommended Standard, the following nomenclature is used to describe this grouping:

8-Bit Word = ‘Byte’

~~1.6 PATENTED TECHNOLOGIES~~

~~The Consultative Committee on Space Data Systems (CCSDS) draws attention to the fact that it is claimed that compliance with this document may involve the use of a patent concerning the method for coding low entropy data given in section 3.~~

~~The CCSDS takes no position concerning the evidence, validity, and scope of these patent rights.~~

~~The holders of these patent rights have assured the CCSDS that they are willing to negotiate licenses under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statements of the holders of these patent rights are registered with CCSDS. Information can be obtained from the CCSDS Secretariat at the address indicated on page i. Contact information for the holder of these patent rights is provided in annex A.~~

~~Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. The CCSDS shall not be held responsible for identifying any or all such patent rights.~~

1.6 REFERENCES

The following documents contain provisions which, through reference in this text, constitute provisions of this Recommended Standard. At the time of publication, the editions indicated were valid. All documents are subject to revision, and users of this Recommended Standard are encouraged to investigate the possibility of applying the most recent editions of the documents indicated below. The CCSDS Secretariat maintains a register of currently valid CCSDS Recommended Standards.

[1] *TM Synchronization and Channel Coding*. Issue 3. Recommendation for Space Data System Standards (Blue Book), CCSDS 131.0-B-3. Washington, D.C.: CCSDS, September 2017.

[2] *Flexible Advanced Coding and Modulation Scheme for High Rate Telemetry Applications*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 131.2-B-1. Washington, D.C.: CCSDS, March 2012.

[3] *CCSDS Space Link Protocols over ETSI DVB-S2 Standard*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 131.3-B-1. Washington, D.C.: CCSDS, March 2013.

[4] *Space Packet Protocol*. Recommendation for Space Data System Standards, CCSDS 133.0-B-1. Blue Book. Issue 1. Washington, D.C.: CCSDS, September 2003.

[5] *Low-Complexity Lossless and Near-Lossless Multispectral and Hyperspectral Image Compression*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 123.0-B-2. Washington, D.C.: CCSDS, February 2019.

[6] *CCSDS File Delivery Protocol (CFDP)*. Issue 4. Recommendation for Space Data System Standards (Blue Book), CCSDS 727.0-B-4. Washington, D.C.: CCSDS, January 2007.

[7] *TM Space Data Link Protocol*. Issue 2. Recommendation for Space Data System Standards (Blue Book), CCSDS 132.0-B-2. Washington, D.C.: CCSDS, September 2015.

[8] *AOS Space Data Link Protocol*. Issue 3. Recommendation for Space Data System Standards (Blue Book), CCSDS 732.0-B-3. Washington, D.C.: CCSDS, September 2015.

[9] *Unified Space Data Link Protocol*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 732.1-B-1. Washington, D.C.: CCSDS, October 2018.

The inputs to the source coder areThe sequence of input samples is partitioned into blocks of length J so that the inputs to the source coder are

$$\boldsymbol{x} = x_1, x_2, \ldots x_J$$

which is a block of J n-bit samples, where n is a constant value.

When the input sequence length is not a multiple of J, a user may append additional 'padding' samples as needed. Compressed data size will be minimized when padding samples are chosen so that the corresponding padded preprocessed samples are zero.

Preprocessor:

The preprocessor applies a reversible function to input data samples $\boldsymbol{x}$, to produce a 'preferred source':

$$\boldsymbol{\delta} = \delta_1, \delta_2, \ldots \delta_i, \ldots \delta_J$$

where each δ_i is an n-bit integer, $0 \le \delta_i \le (2^n-1)$. For an ideal preprocessing stage, $\boldsymbol{\delta}$ will have the following properties:

a) the $\{\delta_i\}$ is statistically independent and identically distributed;

b) the preferred probability, p_m, that any sample δ_i will take on integer value m is a nonincreasing function of value m, for $m = 0, 1, \ldots (2^n-1)$.

The preprocessor function is a reversible operation, and, in general, the best Lossless preprocessor will meet the above conditions and produce the lowest entropy, which is a measure of the smallest average number of bits that can be used to represent each sample.

This Recommended Standard does not attempt to explain methods for choosing a preprocessing stage. This Recommended Standard does provide the definition of a basic preprocessing stage that may be suitable for many applications. However, it is important that the user carefully address this issue since careful selection of an appropriate preprocessing stage is essential for efficient compression and depends on the source-data characteristics. Interested users should refer to reference [B2].

Adaptive Entropy Coder:

The function of the Adaptive Entropy Coder is to calculate uniquely decipherable, variable-length codewords corresponding to each block of samples input from the preprocessor. The entropy coder incorporates multiple coding options, each exhibiting efficient performance over different yet overlapping ranges of entropy. The coder selects the coding option that gives the highest compression ratio among the various options on the same block of J samples. A code-option 'identifier', requiring only a few bits, is attached before the first codeword bit in a coded block to signal the coding option to the decoder for proper decompression. Since the block size J can be small and a new code option is selected for each block, the overall coding can adapt to rapid changes in data statistics.

2.3 PACKETIZATION OF CODED DATA

The variable-length encoded bit stream representing a J-sample block forms a Coded Data Set (CDS). CCSDS telemetry source packet structure is recommended to transport the CDSes, which will be contained in the source data field of the packet. The information related to, for example, the sensor, mission, time, and other mission-specific details necessary for the routing and accounting of the packets, will be contained in the Packet Primary Header and (if present) in the Packet Secondary Header (see reference [2]).

2.3 DATA TRANSMISSION

Individual channel bit errors have greater consequences when data are compressed. The effects of a small error or data loss event can propagate to corrupt an entire compressed sequence of samples. Therefore, measures should be taken to minimize errors and data loss in the compressed data.

This Recommended Standard does not incorporate sync markers or other mechanisms to flag the packets or file headers, or the beginning of a reference sample interval; it is assumed that the transport mechanism used for the delivery of the encoded bit stream will provide the ability to locate the beginning and end of the compressed data and, in the event of data corruption, the beginning of the next packet, file or reference sample interval.

When transmission over a CCSDS space link occurs, application of one of the set of Channel Coding and Synchronization Recommended Standards (references [1], [2], and [3]) will significantly reduce the loss of portions of transmitted data caused by data corruption.

The variable-length encoded bit stream representing a J-sample block forms a Coded Data Set (CDS). In case the encoded stream is to be transmitted over a CCSDS space link, several protocols can be used to transfer the CDSes, including but not limited to:

– Space Packet Protocol (see reference [4]);

– CCSDS File Deliver Protocol (CFDP) (see reference [6]);

– Packet service as provided by the CCSDS Space Data Link Protocols (see references [7], [8], and [9]).

Limits on the maximum size data unit that can be transmitted may be imposed by the protocol used or by other practical implementation considerations. The user is expected to take such limits into account when using this Recommended Standard.

~~2.4 ERROR CONTROL~~

~~Individual channel bit errors have greater consequences when data are compressed. Even then, the consequences need not be catastrophic. For this reason, to limit error propagation when utilizing the source coding algorithm described in this document, the following is recommended:~~

a) ~~use telemetry channel coding as described in reference [1];~~

b) ~~use packetized telemetry as described in reference [2].~~

with an FS codeword (see figure 3-2). This produces a varying codeword length. The FS codewords for the current block of *J* preprocessed samples are transmitted along with the removed LSBs, preceded by an ID field indicating the value of *k*. This process enables the adaptation of codeword length to source-data statistics.

3.3.2 The FS option described in 3.2 is a special case of sample splitting where $k = 0$.

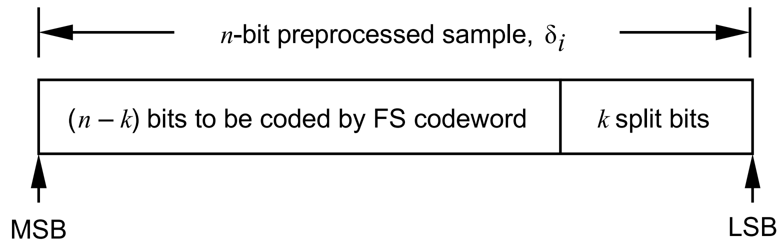


Figure 3-2: Split-Sample Format

3.4 LOW ENTROPY OPTIONS

3.4.1 GENERAL

Two code options, the Second-Extension option[1] and the Zero-Block option, provide more efficient coding than other options when the preprocessed data are highly compressible.

3.4.2 THE SECOND-EXTENSION OPTION

When the Second-Extension option is selected, each pair of preprocessed samples in a *J*-sample block is transformed and encoded using an FS codeword. A pair of consecutive samples ~~δ_i and δ_{i+1}~~ from a *J*-sample preprocessed data block are transformed into a single new symbol ~~γ~~ γ_j by the following equation.

~~$$\gamma = (\delta_i + \delta_{i+1})(\delta_i + \delta_{i+1} + 1)/2 + \delta_{i+1}$$~~

$$\gamma_j = (\delta_{2j-1} + \delta_{2j})(\delta_{2j-1} + \delta_{2j} + 1)/2 + \delta_{2j},$$

where $j = 1, 2, \ldots, J/2$. The *J*/2 transformed symbols in a block are encoded using the FS codeword of table 3-1. ~~The above process requires *J* to be an even integer which the recommended values in 3.1.4 obey (J = 8, 16, 32, or 64).~~

[1] The first extension of a preprocessed sample is the preprocessed sample itself.

3.4.3 ZERO-BLOCK OPTION

3.4.3.1 The Zero-Block option is selected when one or more blocks of preprocessed samples are all zeros. In this case, a single codeword may represent several blocks of preprocessed samples, unlike other options where an FS codeword represents only one or two preprocessed samples.

~~**3.4.3.2** The set of *r* blocks between consecutive reference samples, as described in 4.3, is partitioned into one or more segments. Each segment, except possibly the last, contains s blocks. The recommended value of s is 64.~~

3.4.3.2 The set of *r* blocks between consecutive reference samples (or possibly fewer than *r* blocks at the end of the input sequence), as described in 4.3, is partitioned into one or more segments. Each segment contains 64 blocks except possibly the last, which may be smaller

3.4.3.3 Within each segment, each group of adjacent all-zeros blocks is encoded by the FS codewords, specified in table 3-2, which identify the length of each group. The Remainder-Of-Segment (ROS) codeword in table 3-2 ~~is~~shall be used to denote that the remainder of a segment consists of five or more all-zeros blocks. This applies to every segment, including the last segment of each reference sample interval or the last segment at the end of the input sequence, which can be smaller than 64 blocks.

Table 3-2: Zero-Block Fundamental Sequence Codewords As a Function of the Number of Consecutive All-Zeros Blocks

Number of All-Zeros Blocks	FS Codeword
1	1
2	01
3	001
4	0001
ROS	00001
5	000001
6	0000001
7	00000001
8	000000001
.	.
.	.
.	.
63	0000 … 0000000001 (63 0s and a 1)

NOTE – An implementation that does not use the ROS codeword for segments smaller than 64 blocks (end of reference sample interval or end of input sequence) would still produce an encoded bit stream that can be decoded.

4.2.2 UNIT-DELAY PREDICTOR

The unit-delay prediction technique illustrated in figure 4-2 uses the one-sample delayed input data signal as the predictor for the current data signal, and the prediction error is passed to the following mapper along with the predicted value for mapping to a nonnegative integer. That is, the predicted value, $\hat{x}_i$, is equal to the preceding sample value, except for the first sample in a reference interval (as defined in 4.3) for which the predicted value is the current sample value, x_i.

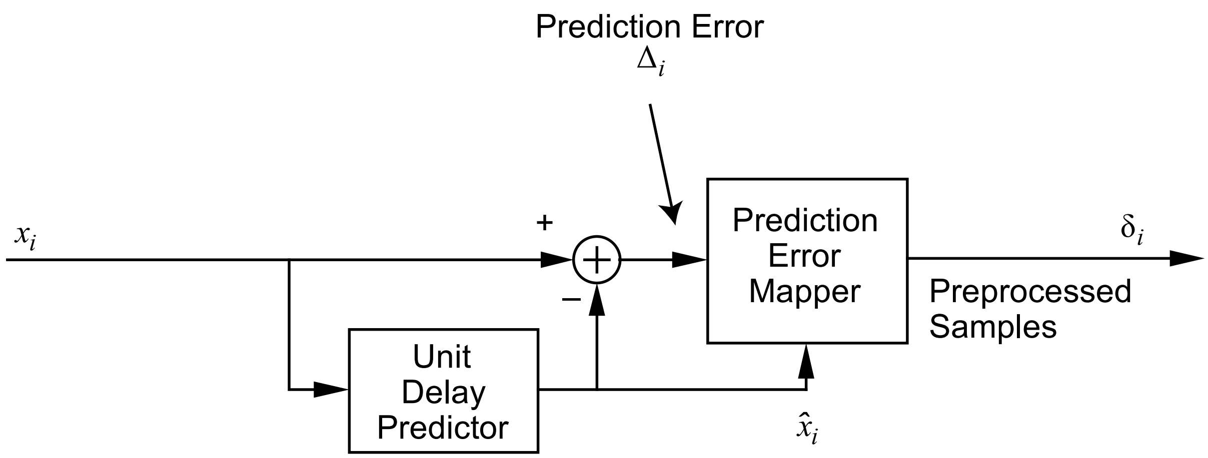


Figure 4-2: Preprocessor Using a Unit-Delay Predictor

4.3 REFERENCE SAMPLE

A reference sample is an unaltered input data sample upon which succeeding sample prediction is based. When a unit-delay predictor or other higher-order predictors that use the previous data signal in prediction are used, reference samples are required by the decoder to recover the sample values from decoded predictor errors. When the reference sample is inserted, there are $J - 1$ preprocessed samples in the CDS, except for sequences of zero blocks, which are encoded in the same CDS. ~~The user must determine how often to insert references.~~The user-specified reference sample interval, r, is limited to a maximum value of 4096 ~~CDSes~~blocks (e.g., 262144 samples when $J = 64$). When a reference sample is not required by the preprocessor, the parameter r serves to define an interval of input data sample blocks that will be further segmented under the zero-block option described in 3.4.3.2.

4.4 PREDICTION ERROR MAPPER

The Prediction Error mapper takes the prediction error values and maps them into non-negative integers suitable for the Adaptive Entropy Coder. The prediction error Δ_i resulting from taking the difference between a signal value, x_i, and a predictor value, $\hat{x}_i$, both n-bit integers, will have an $(n+1)$-bit dynamic range of $[-2^n+1, 2^n-1]$. However, for every predictor

5.1.2.3 For applications not requiring the full entropy range of performance provided by the specified options, a subset of the options at the source may be implemented. The ID is always required, even if a subset of the options is used.

5.1.3 REFERENCE SAMPLE

When the preprocessor is present and the reference sample is required, the first CDS of the Source Packet Data Field or the first CDS or the compressed file shall contain a reference sample. References shall then be inserted ~~in the Source Packet Data Field at least every 4096 CDSes~~every *r* blocks as specified in 4.3. When the preprocessor is absent, or it does not require a reference sample, the reference sample shall not be inserted in the CDS.

5.1.4 CODED DATA SET FORMAT

5.1.4.1 The CDS format when a sample-splitting option is selected is shown in figure 5-1. Figure 5-1a shows the case where there is a reference sample; figure 5-1b shows the format when no reference sample is present. The CDS has the following structure when a sample-splitting option is selected: 1) ID bit sequence optionally followed by an *n*-bit reference sample, 2) compressed data, and 3) concatenated *k* least-significant bits from each sample.

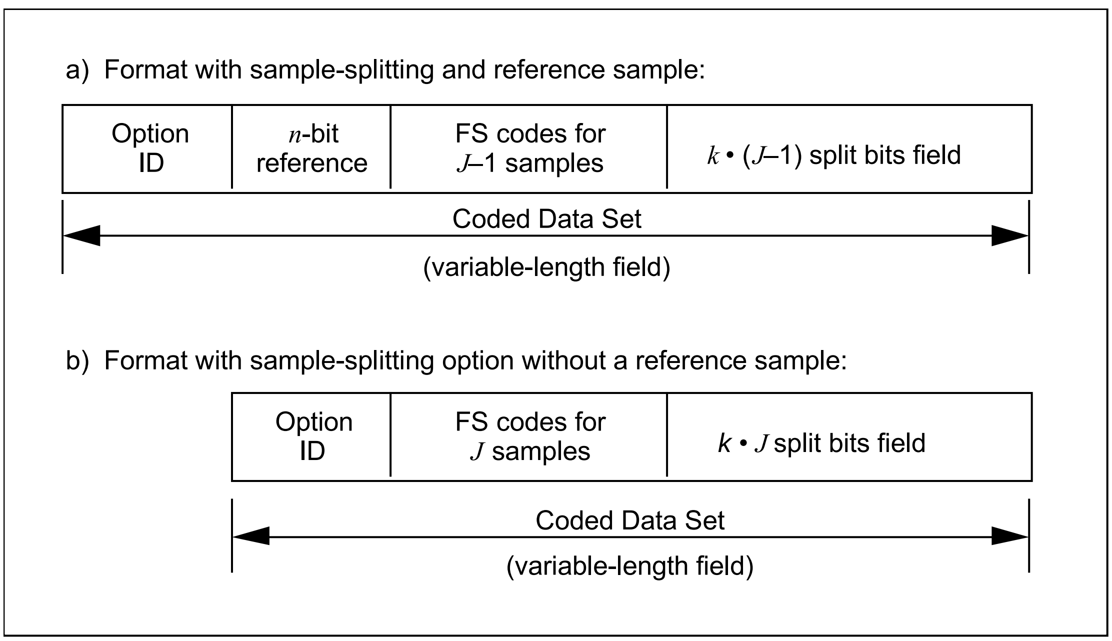


Figure 5-1: CDS Format When Sample-Splitting Option Is Selected

5.1.4.2 When the no-compression option is selected, the CDS is fixed length containing the option ID field, optionally followed by an *n*-bit reference sample, and *J* preprocessed samples. The case where a reference is present is shown in figure 5-2a; the non-reference case is shown in figure 5-2b.

5.1.4.4 When the Second Extension option is selected, the CDS contains the option ID field, optionally followed by an n-bit reference sample, and required FS codewords for $\frac{J}{2}$ transformed samples. The case where a reference is present is shown in figure 5-4a; the non-reference case is shown in figure 5-4b. In the case when a reference is inserted, a ‘0’ sample is added in front of the J–1 preprocessed samples, so $\frac{J}{2}$ samples are produced after the transformation.

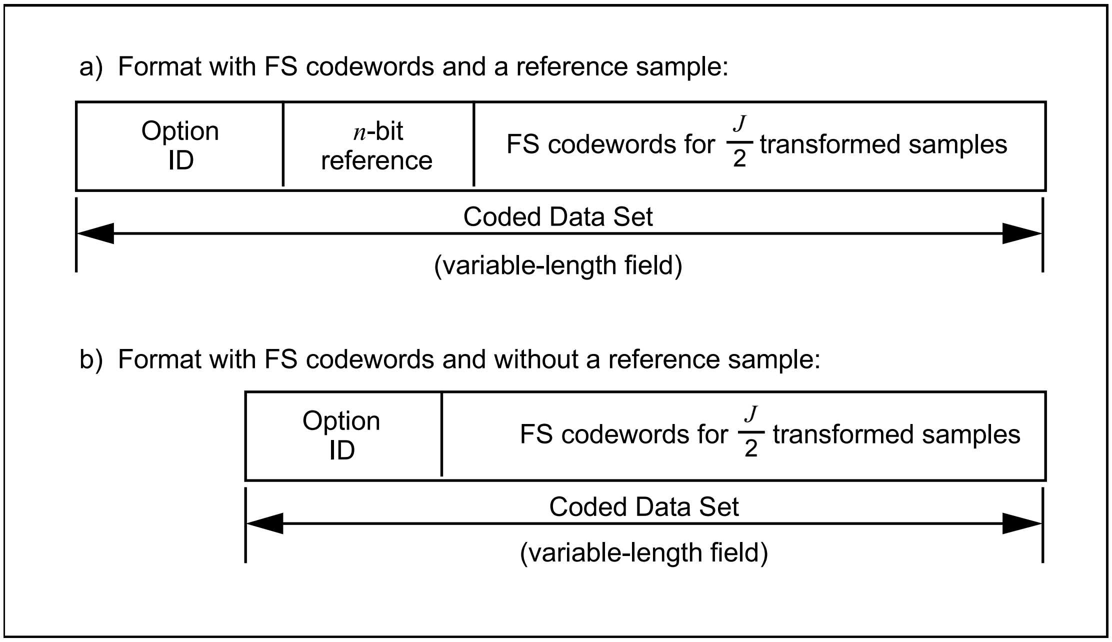


Figure 5-4: CDS Format When the Second-Extension Option Is Selected

5.2 PACKET FORMAT

5.2.1 LOSSLESS PACKET FORMAT

When the CCSDS space packet structure (reference [2]) is used to transport the CDSes, the lossless data compression packets shall be formatted as shown in figure 5-5 (see reference [4]). The packet formatter uses the parameter provided by the source data coder to form one or more CDSes to determine the packet size in bytes. Fill bits of zero value may be needed to force the packet to end on a byte boundary.

Packet Primary Header	Secondary Header (optional)	CDS #1 (with n-bit reference)	. . .	CDS #l	Fill Bits

Figure 5-5: Packet Format for *l* CDSes

5.2.2 PACKET REQUIREMENTS

5.2.2.1 A Source Packet Data Field must meet the following requirements:

– a CDS within a packet must meet the format requirements defined in 5.1;

– when the reference sample is used, the Source Packet Data Field shall begin with a CDS that contains this reference, followed by one or more CDSes; when the reference sample is not required in the preprocessor, or the preprocessor is absent, a reference sample shall not be inserted in the first CDS in the Source Packet Data Field;

– several CDSes can be put in sequence within a source packet;

– fill bits are allowed only at the end of the Source Packet Data field, not within the body of compressed data;

– each packet must end on a byte boundary.

NOTE – Some implementations may require an adequate number of fill bits be added in order to end a packet on an even-numbered byte boundary.

5.2.2.2 Unless the option to use the CIP is chosen (see section 6), in order to decode packets that may include fill bits, several pieces of information must be communicated to the decoder a priori. This information will be mission specific and fixed for a given Application Process Identifier (APID) per mission:

– l, the number of CDSes that are in a packet;

– r, the reference sample interval~~, equaling the number of CDSes counted from one CDS containing a reference sample up to but not including the next consecutive CDS containing a reference sample~~;

– n, the resolution;

– J, the number of samples per block;

– whether the Basic or Restricted set of code options is used (when $n \leq 4$);

– N, number of samples of the input sequence.

5.2.2.3 A Packet Secondary Header is optional and can be used, for example, to relate observation time and position information to the user (see reference [4]).

5.2.2.4 The use of the Sequence Flags in the Packet Sequence Control Field is optional and can be used, for example, to signal a group of compressed data packets. Their use is governed by reference [4].

5.3 FILE FORMAT

When the compressed data is formatted as a file, it shall consist of a File Header, followed by a File Body. The File Header contains the necessary fields to decode the CDSes contained in the File Body.

NOTE – Figure 5-6 depicts the structure of a compressed file.

File Header	File Body

Figure 5-6: File Format

The format for the File Header and the File Body is defined in section 7.

NOTE – CCSDS File Deliver Protocol (CFDP) (see reference [6]) is the preferred CCSDS solution for file transfer, but other private solutions are not forbidden.

6.3.3.3 Compression Technique Identification Field

6.3.3.3.1 The Compression Technique Identification (CTI) field shall signal the compression technique in use for the group of source packets identified by the CIP.

6.3.3.3.2 When no compression technique for the current group is used, the CTI field shall be set to all zeros.

6.3.3.3.3 Only the Lossless data compression technique is currently defined, and is signaled by the value ‘00000001’ in the CTI field. Other values are reserved for future use by CCSDS and are not permitted.

6.3.3.4 Reference Sample Interval Field

~~**6.3.3.4.1** The reference sample interval, *r*, equals the number of CDSes counted from one CDS containing a reference sample up to but not including the next consecutive CDS containing a reference sample. When the preprocessor is absent, or it does not require a reference sample, the reference sample shall not be inserted in the CDS; nevertheless, parameter *r* serves to define the interval of input data sample blocks for the zero-block option as described in 4.3.~~

~~**6.3.3.4.2**~~ The 8-bit Reference Sample Interval field shall contain a binary number equal to $(r–1) \bmod 256$. That is, this field encodes the modulus of $(r–1)$ with respect to divisor 256.

6.3.3.5 Source Configuration

6.3.3.5.1 Subfield Partitions

6.3.3.5.1.1 The Source Configuration field shall be partitioned into four subfields, which should appear in the following order: Preprocessor, Entropy Coder, Extended Parameters, and Instrument Configuration (see figure 6-2). The Preprocessor and Entropy Coder subfields are required, whereas the Instrument Configuration subfield is optional. The Extended Parameters subfield is required whenever any of the following conditions hold:

a) the block length J satisfies $J>16$;

b) the reference sample interval r satisfies $r>256$;

c) the Restricted set of code options is used (see 5.1.2.1);

If none of the above conditions holds, then the Extended Parameters subfield shall not be included.

7 FILE FORMAT

7.1 OVERVIEW

When compressed data are stored or transmitted as a file, the File Format is an optional format that provides information about compression options and defines a structure to store the sequence of CDSes resulting from compressing N input samples.

The File Header contains information which allows decompression of the CDSes stored in the File Body, without *a priori* information.

The File Body contains the compressed CDSes and fill bits.

7.2 FILE STRUCTURE

7.2.1 GENERAL

7.2.1.1 The File Format shall consist on a header specified in 7.2.2 followed by a body specified in 7.2.3.

7.2.1.2 The user-selected output word size, measured in bytes, shall be an integer B in the range $1 \leq B \leq 8$.

7.2.2 HEADER

The File Header shall have the structure defined in table 7-1.

Table 7-1: File Header Structure

Field	Width (bits)	Description	Reference
Reserved	1	This field shall have value '0'.	
Output Word Size (B)	3	The value *B-1* encoded as a 3-bit unsigned binary integer.	7.2
Preprocessor Status	1	'0': Preprocessor absent '1': Preprocessor present	4.1
Predictor Type	3	'000': bypass predictor or preprocessor absent '001': unit delay predictor '111': application-specific predictor All other codes are reserved by CCSDS for future preprocessing options.	4.2

Field	Width (bits)	Description	Reference
Mapper Type	2	'00': Prediction Error mapper or preprocessor absent '01': reserved '10': reserved '11': application-specific mapper	4.4
Data Sense	1	'0': two's complement '1': positive (mandatory if preprocessor is bypassed or preprocessor absent)	4.4
Reserved	8	This field shall have the value '00000000'.	
Input Data Resolution	5	This field shall contain the value *n-1* encoded as a 5-bit unsigned binary integer.	4.4
Reserved	1	This field shall have the value '0'.	
Block Size	2	'00': J=8 '01': J=16 '10': J=32 '11': J=64	3.1
Restricted Code Option	1	'0': Basic set of code options are used; '1': Restricted set of code options are used.	5.1.2.1
Reference Sample Interval	12	This field shall contain a binary number equal to r–1, encoded as a 12-bit unsigned binary integer.	4.3
Reserved	8	This field shall have the value '00000000'.	
Number of Samples (N)	48	This field shall be set to the total number of compressed input samples that are contained in the file, encoded as the 48-bit unsigned binary integer representation of *N-1*.	

7.2.3 BODY

7.2.3.1 The File Body shall consist of the concatenation of the Coded Data Sets (CDSes, as defined in defined in subsection 5.1.4) resulting from compressing N input samples.

7.2.3.2 Following the last CDS in the compressed file, fill bits shall be appended as needed to reach the next output word boundary, so that the compressed file size is a multiple of the output word size (B). Fill bits shall be all 'zeros'.

A3 PATENT CONSIDERATIONS

~~Implementers should be aware that the method for coding low entropy data specified in this Recommended Standard is covered by U.S. Patent 5448642. Potential user agencies should direct their requests for licenses to the U.S. Patent 5448642 patent rights holder, whose contact information is:~~

~~Bryan A. Geurts~~
~~Department Office of Patent Counsel~~
~~NASA Goddard Space Flight Center~~
~~Code 140.1~~
~~8800 Greenbelt Road, Greenbelt, MD 20771~~
~~Phone: +1 301 286 7352~~
~~Fax: +1 301 286 9502~~
~~E-mail: bryan.a.geurts@nasa.gov~~

At time of publication, the specifications of this Recommended Standard are not known to be the subject of patent rights.There is currently no known active patent for this standard.[2]

[2] The United States Patent and Trademark Office shows the status of a previously applicable patent to be 'Expired' at time of publication of the current issue of this Recommended Standard.