The Consultative Committee for Space Data Systems

Draft Recommendation for
Space Data System Standards

HIGH PHOTON EFFICIENCY OPTICAL COMMUNICATIONS

CODING & MODULATION

PROPOSED DRAFT RECOMMENDED STANDARD

CCSDS XXX.0-B-0.2

WHITE BOOK
September 2015

AUTHORITY

Issue:	White Book, Draft 2
Date:	September 2015
Location:	Not Applicable

(WHEN THIS RECOMMENDED STANDARD IS FINALIZED, IT WILL CONTAIN THE FOLLOWING STATEMENT OF AUTHORITY:)

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and represents the consensus technical agreement of the participating CCSDS Member Agencies. The procedure for review and authorization of CCSDS documents is detailed in *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4), and the record of Agency participation in the authorization of this document can be obtained from the CCSDS Secretariat at the address below.

This document is published and maintained by:

CCSDS Secretariat
Space Communications and Navigation Office, 7L70
Space Operations Mission Directorate
NASA Headquarters
Washington, DC 20546-0001, USA

FOREWORD

This document is a CCSDS recommended standard for the channel coding and modulation of signals to be used in optical communications systems of space missions. It was contributed to CCSDS by NASA. The channel coding and modulation concepts described herein are intended for missions that are cross-supported between Agencies of the CCSDS.

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Experimental Specification is therefore subject to CCSDS document management and change control procedures, which are defined in the *Procedures Manual for the Consultative Committee for Space Data Systems*. Current versions of CCSDS documents are maintained at the CCSDS Web site:

http://www.ccsds.org/

Questions relating to the contents or status of this document should be addressed to the CCSDS Secretariat at the address indicated on page i.

At time of publication, the active Member and Observer Agencies of the CCSDS were:

Member Agencies

- Agenzia Spaziale Italiana (ASI)/Italy.
- Canadian Space Agency (CSA)/Canada.
- Centre National d'Etudes Spatiales (CNES)/France.
- China National Space Administration (CNSA)/People's Republic of China.
- Deutsches Zentrum für Luft- und Raumfahrt (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Federal Space Agency (FSA)/Russian Federation.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- Japan Aerospace Exploration Agency (JAXA)/Japan.
- National Aeronautics and Space Administration (NASA)/USA.
- UK Space Agency/United Kingdom.

Observer Agencies

- Austrian Space Agency (ASA)/Austria.
- Belgian Federal Science Policy Office (BFSPO)/Belgium.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- China Satellite Launch and Tracking Control General, Beijing Institute of Tracking and Telecommunications Technology (CLTC/BITTT)/China.
- Chinese Academy of Sciences (CAS)/China.
- Chinese Academy of Space Technology (CAST)/China.
- Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
- Danish National Space Center (DNSC)/Denmark.
- Departamento de Ciência e Tecnologia Aeroespacial (DCTA)/Brazil.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Geo-Informatics and Space Technology Development Agency (GISTDA)/Thailand.
- Hellenic National Space Committee (HNSC)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Institute of Space Research (IKI)/Russian Federation.
- KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- Korea Aerospace Research Institute (KARI)/Korea.
- Ministry of Communications (MOC)/Israel.
- National Institute of Information and Communications Technology (NICT)/Japan.
- National Oceanic and Atmospheric Administration (NOAA)/USA.
- National Space Agency of the Republic of Kazakhstan (NSARK)/Kazakhstan.
- National Space Organization (NSPO)/Chinese Taipei.
- Naval Center for Space Technology (NCST)/USA.
- Scientific and Technological Research Council of Turkey (TUBITAK)/Turkey.
- South African National Space Agency (SANSA)/Republic of South Africa.
- Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
- Swedish Space Corporation (SSC)/Sweden.
- Swiss Space Office (SSO)/Switzerland.
- United States Geological Survey (USGS)/USA.

PREFACE

This document is a draft CCSDS Recommended Standard. Its 'White Book' status indicates that its contents are not stable, and several iterations resulting in substantial technical changes are likely to occur before it is considered to be sufficiently mature to be released for review by the CCSDS Agencies.

Implementers are cautioned **not** to fabricate any final equipment in accordance with this document's technical content.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

DOCUMENT CONTROL

Document	Title	Date	Status
CCSDS XXX.0-B-0.1	Optical Communications – Coding & Modulation, Recommended Standard, Draft 1	December 2014	Superseded
CCSDS XXX.0-B-0.2	Optical Communications – Coding & Modulation, Recommended Standard, Draft 2	March 2015	Superseded
CCSDS XXX.0-B-0.3	Optical Communications – Coding & Modulation, Recommended Standard, Draft 3	September 2015	Current draft

CONTENTS

Section Page

Figure

Table

No table of contents entries found.

1 INTRODUCTION

1.1 PURPOSE AND SCOPE

The purpose of this Recommended Standard is to specify the signaling format for optical communications systems used by space missions, including downlink (space-to-ground) and uplink (ground-to-space). When provided with a set of CCSDS Transfer Frames produced by the Data Link Protocol Sublayer [2], [3], this specification allows one to determine the sequence of pulsed and non-pulsed slots to be transmitted.

1.2 NOMENCLATURE

1.2.1 NORMATIVE TEXT

The following conventions apply throughout this Specification:

a) the words ‘shall’ and ‘must’ imply a binding and verifiable specification;

b) the word ‘should’ implies an optional, but desirable, specification;

c) the word ‘may’ implies an optional specification;

d) the words ‘is’, ‘are’, and ‘will’ imply statements of fact.

1.2.2 INFORMATIVE TEXT

In the normative sections of this document, informative text is set off from the normative specifications either in notes or under one of the following subsection headings:

– Overview;

– Background;

– Rationale;

– Discussion.

1.3 DEFINITIONS

1.3.1 DEFINITIONS FROM THE OPEN SYSTEM INTERCONNECTION (OSI) BASIC REFERENCE MODEL

This Recommended Standard makes use of a number of terms defined in reference [3]. The use of those terms in this Recommended Standard shall be understood in a generic sense, i.e., in the sense that those terms are generally applicable to any of a variety of technologies that provide for the exchange of information between real systems. Those terms are:

a) Data Link Layer;

b) b) Physical Layer;

c) c) service;

d) d) service data unit.

1.3.2 DEFINITION OF CADU

In this Recommended Standard, the data unit that consists of the Attached Synchronization Marker (ASM) and the Transfer Frame is called the Channel Access Data Unit (CADU).

1.4 CONVENTIONS

In this document, the following convention is used to identify each bit in an *N*-bit field. The first bit in the field 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*-1'. When the field is used to express a binary value (such as a counter), the Most Significant Bit (MSB) shall be the first transmitted bit of the field, i.e., 'Bit 0' (see Figure 1-1).

Figure 1-1: Bit Numbering Convention

The convention for matrices differs from that for bit fields. Matrices are indexed beginning with the number '1'.

In accordance with standard data-communications practice, data fields are often grouped into 8-bit 'words' which conform to the above convention. Throughout this Specification, such an 8-bit word is called an 'octet'. The numbering for octets within a data structure starts with '0'.

NOTE: Throughout this document, "bit" refers to the contents of the transfer frames. A bit is a unit of information transferred between the data link protocol sublayer and the coding sublayers. Other symbols, whether binary or nonbinary, will be referred to by other names, such as "binary digits." It should be understood that the ordering conventions described above shall apply equally to other types of symbols.

1.5 REFERENCES

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

[1] CCSDS 131.0-B-2, "TM Synchronization and Channel Coding," Blue Book. Issue 2. August 2011.
[2] CCSDS 132.0-B-1, "TM Space Data Link Protocol. Recommendation for Space Data System Standards," Blue Book. Issue 1. September 2003.
[3] CCSDS 732.0-B-2, "AOS Space Data Link Protocol. Recommendation for Space Data System Standards," Blue Book. Issue 2. July 2006.
[4] CCSDS 231.0-B-2, "TC Synchronization and Channel Coding," Blue Book. Issue 2. September 2010.
[5] CCSDS 231.0-B-XX, "Short Blocklength LDPC Codes for TC Synchronization and Channel Coding. Proposed New Chapter for 231.0-B-2," Pink Sheets. October 2014.

2 OVERVIEW

2.1 ARCHITECTURE

Figure 2-1 illustrates the relationship of this Recommended Standard to the Open Systems Interconnection reference model (reference [A2]). Two sublayers of the Data Link Layer are defined for CCSDS space link protocols. The TM and AOS Space Data Link Protocols specified in Ref.s [2] and [3], respectively, correspond to the Data Link Protocol Sublayer, and provide functions for transferring data using the protocol data unit called the Transfer Frame. The Optical Coding and Modulation protocol specified in this Recommendation provides the functions of the synchronization and channel coding sublayer of the data link layer, for transferring Transfer Frames over an optical space link. In addition, this Recommendation specifies the modulations to be used on the physical layer of an optical space link.

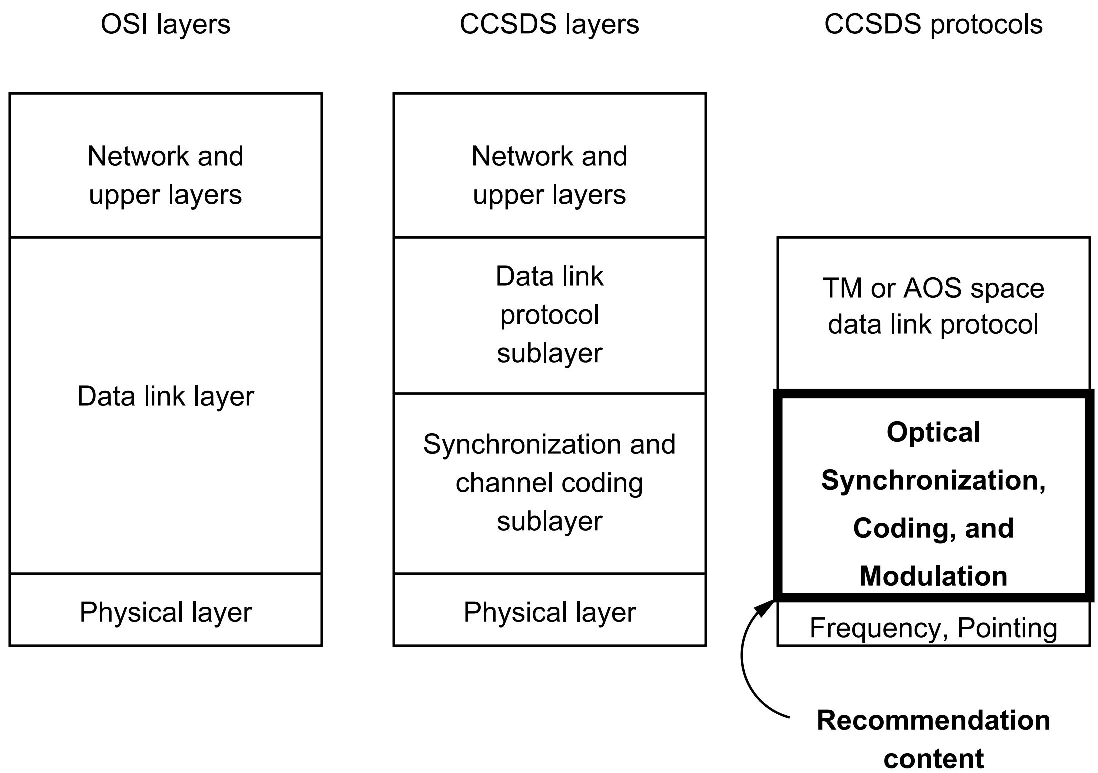

Figure 2-1: Relationship with OSI layers

2.2 SUMMARY OF FUNCTIONS

The Optical Synchronization, Coding, and Modulation sublayer provides an optical signaling specification for the downlink, and a separate optical signaling specification for the uplink.

The downlink specification defines the relationship between input CCSDS Transfer Frames and output pulsed slots. This includes specification of a CCSDS Transfer Frame slicer, CRC, channel coding, modulation, channel interleaver, codeblock synchronization marker, pseudo-noise (PN) spreader, slot mapper, and guard slot insertion.

The uplink specification defines the relationship between input frames and output pulsed slots. The uplink signaling functions are to: 1) provide a reference beacon; 2) aid synchronization; 3) support a command capability for near-real-time link control; and 5) support ranging.

3 DOWNLINK SIGNALING

The Recommended Standard operates, consistent with Ref.s [2] and [3], by taking CCSDS Transfer Frames as input, and producing a binary vector indicating the positions of pulsed slots at output. The functional blocks of the architecture are shown in Figure 3-1, along with the notation used in the following sections that define these functions mathematically. It should be understood that the functions need not be implemented explicitly as defined here; any implementation producing the proper pattern of pulsed slots complies with the standard.

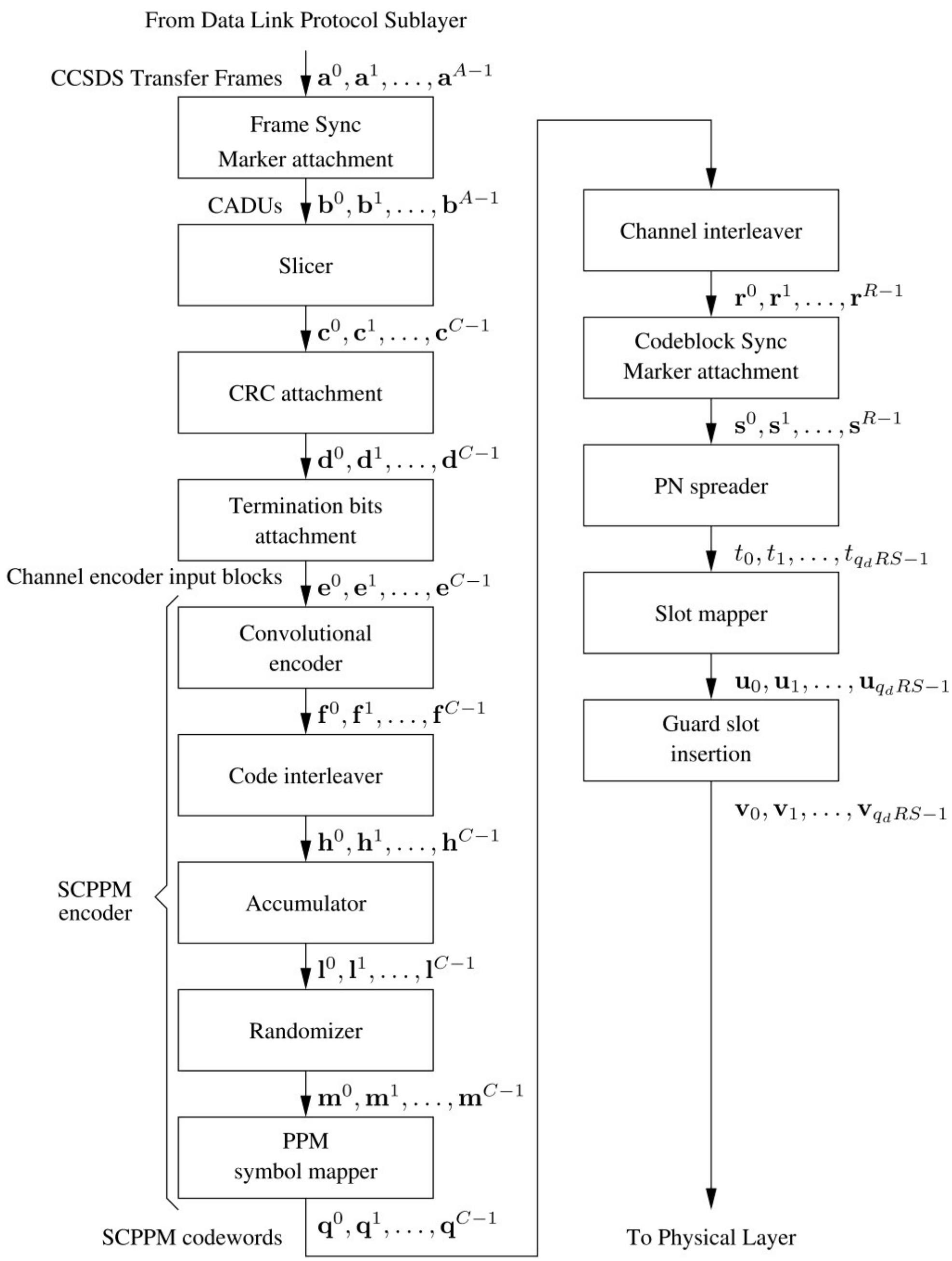

Figure 3-1: Functional diagram for downlink signaling

3.1 CCSDS TRANSFER FRAMES

The input to the coding and synchronization sublayer shall be a sequence of A CCSDS Transfer Frames

$$\boldsymbol{a}^0, \boldsymbol{a}^1, \dots, \boldsymbol{a}^{A-1}$$

each containing T bits. The structure of the Transfer Frames is defined in Ref.s [2] and [3]. For $i \in \{0, 1, \dots, A-1\}$, the i^{th} Transfer Frame is denoted

$$\boldsymbol{a}^i = a_0^i, a_1^i, \dots, a_{T-1}^i$$

where $a_j^i \in \{0,1\}$ is the j^{th} bit of the i^{th} frame.

3.2 FRAME SYNCHRONIZATION MARKER (FSM)

A Frame Synchronization Marker (FSM) shall be prepended to each Transfer Frame. The FSM shall be the sequence $\boldsymbol{s} = s_0, s_1, \dots, s_{63}$, represented in hexadecimal notation as

$$\boldsymbol{s} = \text{0x034776C7272895B0}$$

A 64-binary-digit FSM together with an T-binary-digit Transfer Frame forms a Channel Access Data Unit (CADU) of length $B = T + 64$. For $i \in \{0, 1, \dots, A-1\}$, the i^{th} CADU shall be

$$\boldsymbol{b}^i = b_0^i, b_1^i, \dots, b_{B-1}^i$$

where

$$b_j^i = \begin{cases} s_j, & j < 64 \\ a_{j-64}^i, & 64 \le j < B \end{cases}$$

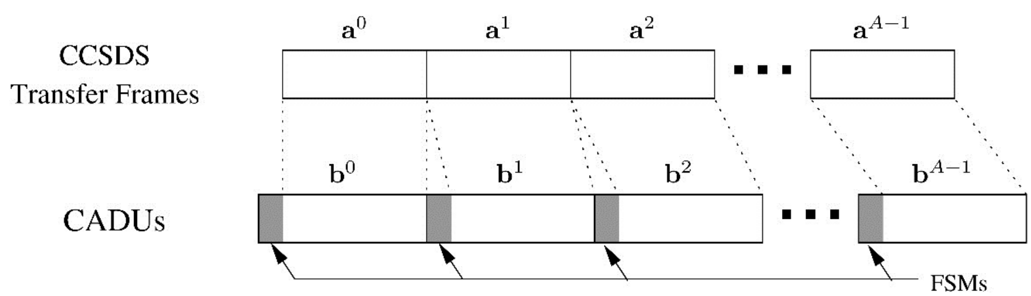

Figure 3-2: FSM attachment

3.3 SLICER

The sequence of CADUs

$$\boldsymbol{b}^0, \boldsymbol{b}^1, \dots, \boldsymbol{b}^{A-1}$$

is a vector of vectors that can be viewed as a single vector

$$\widehat{\boldsymbol{b}} = \hat{b}_0, \hat{b}_1, \dots, \hat{b}_{AB-1}$$

where for $i \in \{0, 1, \ldots, A-1\}$ and $j \in \{0, 1, \ldots, B-1\}$,

$$\hat{b}_{iB+j} = b_j^i$$

The sequence $\hat{b}$ shall be padded at its end with the minimum number of zeroes so that its length is a multiple of k given in Table 3-1.

$$\tilde{\boldsymbol{b}} = \hat{b}_0, \hat{b}_1, \ldots, \hat{b}_{AB-1}, \underbrace{0, 0, \ldots, 0}_{P}$$

where

$$\tilde{b}_j = \begin{cases} \hat{b}_j, & j < AB \\ 0, & AB \le j < AB + P \end{cases}$$

and

$$P = \min\{p : k \mid AB + p\}$$

The sequence $\tilde{b}$ shall be sliced into information blocks of length k, as shown in Figure 3-3. The slicer re-indexes $\tilde{b}$ into $C = (AB + P)/k$ blocks each of length $\hat{k}$:

$$\boldsymbol{c} = \boldsymbol{c}^0, \boldsymbol{c}^1, \ldots, \boldsymbol{c}^{C-1}$$

where for $i \in \{0, 1, \ldots, C-1\}$ the i^{th} block is denoted $\boldsymbol{c}^i = c_0^i, c_1^i, \ldots, c_{k-1}^i$, and for $j \in \{0, 1, \ldots, k-1\}$,

$$c_j^i = \tilde{b}_{ki+j}$$

Table 3-1: Information block sizes

Code Rate	Information block size k	Length of information blocks with CRC-32 and 2-binary-digit termination $\hat{k}$
1/3	5006	5040
1/2	7526	7560
2/3	10046	10080

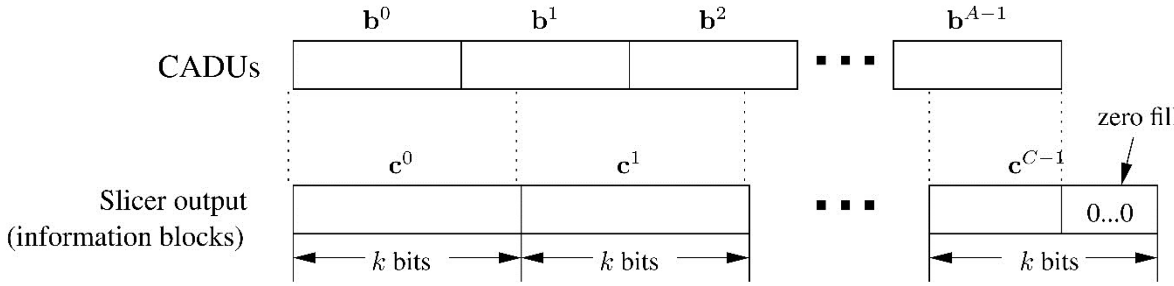

Figure 3-3: Slicer

3.4 CRC ATTACHMENT

Thirty-two cyclic redundancy check CRC binary digits shall be appended to the end of each randomized information block. The CRC parity binary digits are computed as follows. The i^{th} randomized information block $\boldsymbol{c}^i$ is padded with 32 zeroes and expressed in polynomial notation as

$$c^i(D) = c_0^i D^{k+31} + c_1^i D^{k+30} + \cdots + c_{k-1}^i D^{32}$$

This polynomial is divided by the CRC generator polynomial over GF(2) [see Koopman2002 – this polynomial, 0x90022004, maximizes min. dist. For inputs of 8 to 32,738]

$$g(D) = D^{32} + D^{29} + D^{18} + D^{14} + D^3 + 1$$

resulting in the remainder

$$p(D) = p_0 D^{31} + p_1 D^{30} + \cdots + p_{30} D + p_{31}$$

The i^{th} randomized information block with the attached CRC is given by

$$\boldsymbol{d}^i = c_0^i, c_1^i, \ldots, c_{k-1}^i, p_0, p_1, \ldots, p_{31}$$

Note: $g(D)$ as defined here is a common 32-bit CRC also used for HDLC, ANSI X3.66, ITU-T V.42, Ethernet, Serial ATA, etc.

3.5 TERMINATION BINARY DIGITS ATTACHMENT

Two zeroes are appended to the randomized information block with attached CRC, to produce a block of the length $\hat{k} = k + 34$ binary digits, called the SCPPM encoder input block. The i^{th} SCPPM encoder input block is $\boldsymbol{e}^i = e_0^i, e_1^i, \ldots, e_{\hat{k}-1}^i$, where

$$e_j^i = \begin{cases} d_j^i, & 0 \le j < \hat{k} - 2 \\ 0, & j = \hat{k} - 2 \text{ or } j = \hat{k} - 1 \end{cases}$$

The CRC and termination binary digits are shown in Figure 3-4.

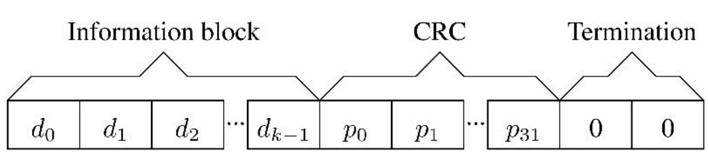

Figure 3-4: CRC and termination

3.6 SCPPM ENCODER

The SCPPM encoder has the structure shown in

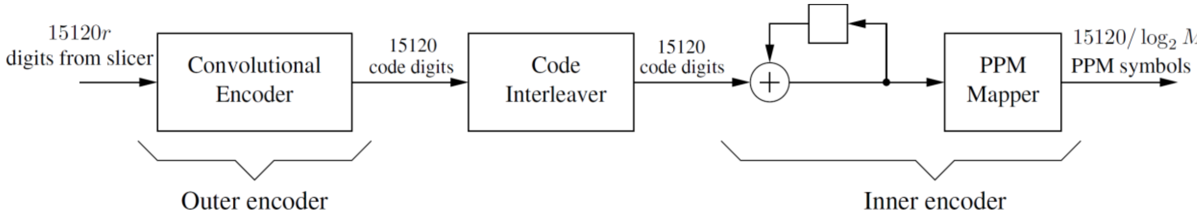

Figure 3-5. The i^{th} SCPPM encoder takes an input block of length $\hat{k} = 15120r$ (see Table 3-1), and produces 15120 convolutionally coded binary symbols which are interleaved, accumulated, and mapped to PPM symbols. The individual SCPPM encoder components are described in the following subsections.

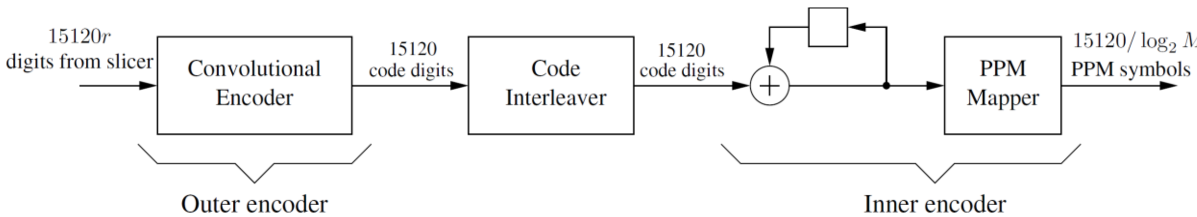

Figure 3-5: SCPPM Encoder

3.6.1 CONVOLUTIONAL ENCODER

The SCPPM outer code shall be a constraint-length three convolutional code defined by the generator polynomials

$$g^{(1)}(D) = 1 + D^2$$
$$g^{(2)}(D) = 1 + D + D^2$$
$$g^{(3)}(D) = 1 + D + D^2$$

or [5, 7, 7] in octal notation. The encoder for this rate 1/3 mother code is shown in Figure 3-6. The encoder shall be initialized to the all-zeroes state prior to encoding each input block.

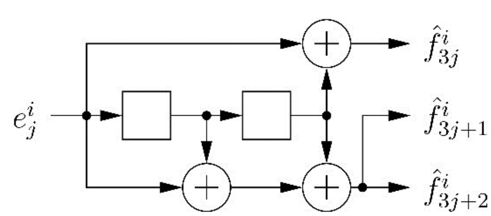

Figure 3-6: Encoder for rate 1/3 mother convolutional encoder

The j^{th} binary digit of the i^{th} SCPPM encoder input block, e_j^i, enters the convolutional encoder, which in response produces the three code symbols $\hat{f}_{3j}^i, \hat{f}_{3j+1}^i, \hat{f}_{3j+2}^i$, corresponding to the polynomials $g^{(1)}(D), g^{(2)}(D), g^{(3)}(D)$, respectively. After all $\hat{k}$ binary digits of $\boldsymbol{e}^i$ enter the encoder, the encoder has produced the convolutional codeword

$$\hat{\boldsymbol{f}}^i = \hat{f}^{\,i}_0, \hat{f}^i_1, \dots, \hat{f}^i_{3\hat{k}-1}$$

The rate 1/3 code may be punctured, resulting in a rate 1/2 or rate 2/3 code, using the puncture patterns given in

Table 3-2. The puncturing is accomplished with the following procedure:

$j \leftarrow 0$
for $m \leftarrow 0$ **to** $3\hat{k} - 1$
  if $(p_{m \bmod 6} \equiv 1)$
    $f^i_j \leftarrow \hat{f}^i_m$
    $j \leftarrow j + 1$
  endif
endfor

The rate 1/2 code punctures every 3rd code symbol, $\hat{f}^i_{3\hat{k}+2}$. The rate 2/3 code additionally punctures every other first code symbol, $\hat{f}^i_{3\hat{k}}$. The resulting i^{th} convolutional codeword is denoted

$$\boldsymbol{f}^i = f^i_0, f^i_1, \dots, f^i_{15119}$$

NOTE: $\hat{k}$ is defined so that, regardless of rate, each convolutional codeword has length 15120.

Table 3-2: Convolutional encoder puncture patterns

Rate	p_0	p_1	p_2	p_3	p_4	p_5
1/3	1	1	1	1	1	1
1/2	1	1	0	1	1	0
2/3	1	1	0	0	1	0

3.6.2 CODE INTERLEAVER

The binary symbols of each 15120-symbol convolutional codeword shall be permuted by a 15120-binary-symbol block interleaver. The j^{th}code symbol of the i^{th} interleaved codeword shall be

$$h^i_j = f^i_{\pi(j)}$$

where

$$\pi(j) = (11j + 210j^2) \mod 15120$$

The interleaver may be implemented by writing code symbols sequentially to positions 1 through 15120, and reading code symbols in interleaved order from positions $\pi(1)$ through $\pi(15120)$.

NOTE: The code interleaver is part of the SCPPM encoder, and is not related to the channel interleaver.

3.6.3 ACCUMULATOR

Interleaved convolutional codewords shall enter an accumulator, a rate 1 code with transfer function $1/(1+D)$, as shown in Figure 3-7.

The accumulator shall be initialized to the zero-state prior to encoding each interleaved convolutional codeword. For the i^{th} codeword, the j^{th} output of the accumulator is given by

$$l_j^i = \begin{cases} h_j^i, & j = 0 \\ l_{j-1}^i \oplus h_j^i, & 1 \le j < 15120 \end{cases}$$

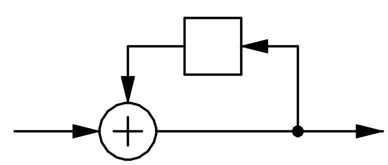

Figure 3-7: The accumulator

3.6.4 RANDOMIZER

Each output block from the accumulator shall be randomized by performing the digit-wise modulo-2 addition of it with a pseudo-noise (PN) sequence produced by a linear feedback shift register. The shift register is initialized to the all-one state at the beginning of each accumulated codeblock of 15120 binary symbols.

The generator polynomial of the PN sequence is given by:

$$g(D) = D^8 + D^7 + D^5 + D^3 + 1$$

The PN sequence is periodic with period 255. The first 40 binary digits of the PN sequence are

$$p_0, \dots, p_{39} = 1111\ 1111\ 0100\ 1000\ 0000\ 1110\ 1100\ 0000\ 1001\ 1010$$

The i^{th} randomized block is given by

$$\boldsymbol{m}^i = l_0^i \oplus p_0, l_1^i \oplus p_1, \dots, l_{15119}^i \oplus p_{k-1}$$

where ⊕ represents modulo-2 addition. One way to implement the randomizer is with the shift register structure shown in Figure 3-8.

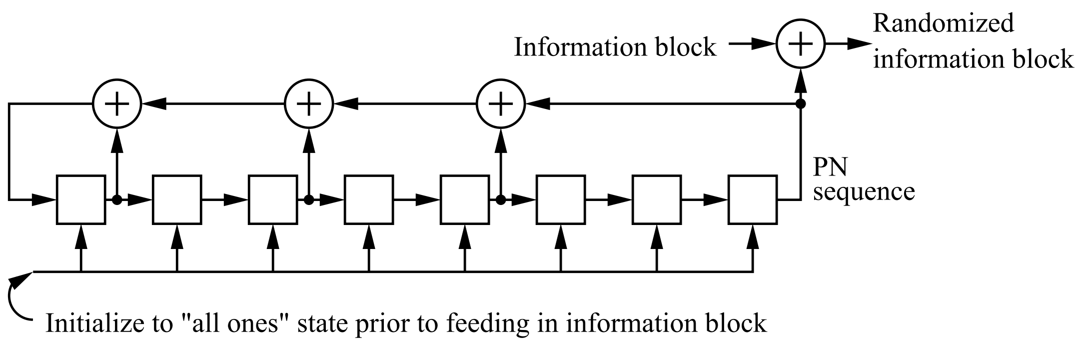

Figure 3-8: Randomizer data flow

3.6.5 PPM SYMBOL MAPPER

The output code symbols from the accumulator shall be mapped to M-ary PPM symbols, $M \in \{4, 8, 16, 32, 64, 128,\ 256\}$. Each $m = \log_2 M$ binary code symbols are grouped to form one PPM symbol, which is an integer in $\{0, 1, \dots, M-1\}$. The j^{th} PPM symbol of the i^{th} SCPPM codeword shall be

$$q_j^i = \sum_{a=0}^{m-1} 2^a \cdot l_{mj+a}^i$$

In this way, the PPM symbol is simply the integer value corresponding to each grouping of m binary code symbols. For example, when $M = 16$ an accumulator output of

$$l_0^i, l_1^i, l_3^i, l_4^i, l_5^i, l_6^i, l_7^i, \dots = \underbrace{1, 1, 0, 1}_{13}, \underbrace{0, 1, 0, 1}_{5}, \dots$$

would correspond to PPM symbols $q_0^i, q_1^i, \dots = 13, 5, \dots$.

The output is the sequence of SCPPM codewords $\boldsymbol{q}^0, \boldsymbol{q}^1, \dots, \boldsymbol{q}^{C-1}$. Each codeword has

$$S = \frac{15120}{\log_2 M}$$

PPM symbols.

NOTE: Since 15120 is a multiple of $\log_2 M$, there are no leftover code symbols in the groupings.

3.7 CHANNEL INTERLEAVER

The sequence of SCPPM codewords $\boldsymbol{q} = \boldsymbol{q}^0, \boldsymbol{q}^1, \dots, \boldsymbol{q}^{C-1}$ is a vector of vectors that can be viewed as a single vector of PPM symbols

$$\hat{\boldsymbol{q}} = \hat{q}_0, \hat{q}_1, \dots, \hat{q}_{CS-1}$$

where for $i \in \{0, 1, \dots, C-1\}$ and $j \in \{0, 1, \dots, S-1\}$,

$$\hat{q}_{iS+j} = q_j^i$$

The sequence of PPM symbols $\hat{\boldsymbol{q}}$ shall be channel interleaved with a convolutional interleaver. The interleaver has N rows, with the i^{th} row containing a shift register of length iB, meaning that it holds iB PPM symbols. The structure of the channel interleaver is shown in Figure 3-9.

Prior to interleaving, the shift registers may be in any state. The input PPM symbols $\hat{\boldsymbol{q}}$ are de-multiplexed into the N rows, sequentially and in circular fashion, beginning with row 0. The outputs of the N shift registers are multiplexed, sequentially and in circular fashion, beginning with row 0. In this way, the de-multiplexer arm is always positioned at the same row as the multiplexer arm.

The j^{th} interleaver output is

$$\hat{r}_j = \hat{q}_{\pi[j]}$$

where $\pi[j]$ is defined recursively by

$$\pi[j] = \begin{cases} j, & \text{if } j = 0 \mod N \\ \pi[j-1] - NB + 1, & \text{otherwise} \end{cases}$$

Negative values of $\pi[j]$ refer to initial interleaver register contents, and $\hat{r}_j$ in these cases may be any value.

For example, when $N = 4$ and $B = 1$, the input $\hat{q}_0, \hat{q}_1, \hat{q}_2, \dots$ will produce an interleaver output of

$$\hat{r}_0, \hat{r}_1, \hat{r}_2, \hat{r}_3, \hat{r}_4, \hat{r}_5, \hat{r}_6, \hat{r}_7, \hat{r}_8, \hat{r}_9, \hat{r}_{10}, \dots = \hat{q}_0, \hat{q}_{-3}, \hat{q}_{-6}, \hat{q}_{-9}, \hat{q}_4, \hat{q}_1, \hat{q}_{-2}, \hat{q}_{-5}, \hat{q}_8, \hat{q}_5, \hat{q}_2, \dots$$

During the operation of the interleaver, the entire initial contents of the interleaver are produced at the output. This is $N(N-1)B/2$ symbols. After the last symbol, $\hat{q}_{CS-1}$, is input, the interleaver must be operated another $N(N-1)B/2$ steps before it appears at the output. Thus, the output contains $N(N-1)B$ more symbols than the input. This output is

$$\hat{\boldsymbol{r}} = \hat{r}_0, \hat{r}_1, \dots, \hat{r}_{SC+N(N-1)B-1}$$

The sequence $\hat{\boldsymbol{r}}$ can be re-indexed into $R = C + N(N-1)B/S$ blocks each containing S symbols:

$$\boldsymbol{r} = \boldsymbol{r}^0, \boldsymbol{r}^1, \dots, \boldsymbol{r}^{R-1}$$

where for $i \in \{0, 1, \ldots, R-1\}$ the i^{th} block is denoted $\boldsymbol{r}^i = r_0^i, r_1^i, \ldots, r_{S-1}^i$, and for $j \in \{0, 1, \ldots, S-1\}$,

$$r_j^i = \hat{r}_{iS+j}$$

Each $\boldsymbol{r}^i$ is called an interleaved codeword (notwithstanding the fact that it contains symbols from many different SCPPM codewords), because it contains S symbols.

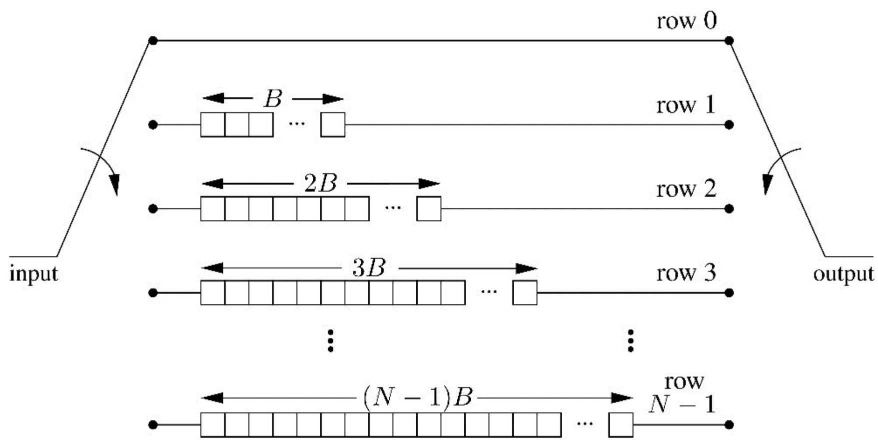

Figure 3-9: Convolutional channel interleaver

3.8 CODEBLOCK SYNCHRONIZATION MARKER

A codeblock synchronization marker (CSM) of W M-ary PPM symbols shall be prepended to each interleaved SCPPM codeword. The CSM is denoted

$$\boldsymbol{w} = w_0, w_1, \ldots, w_{W-1} = 0x\text{[TBD]}$$

After CSM attachment, the j^{th} PPM symbol of the i^{th} interleaved codeword is:

$$\hat{s}_j^i = \begin{cases} w_j, & \text{if } j < W \\ r_{j-W}^i, & \text{if } W \le j < S + W \end{cases}$$

The sequence of CSM+interleaved-codewords is $\hat{\boldsymbol{s}} = \hat{\boldsymbol{s}}^0, \hat{\boldsymbol{s}}^1, \ldots, \hat{\boldsymbol{s}}^{R-1}$. This sequence is re-indexed in PPM symbols as $\boldsymbol{s} = s_0, s_1, \ldots, s_{RS-1}$, where

$$s_{iS+j} = \hat{s}_j^i$$

3.9 PN SPREADER

Each PPM symbol shall be PN spread by a factor of q_d by repeating the symbol q_d times and adding an offset to each symbol according to a PN sequence. Each set of $m = \log_2 M$ binary digits from the periodic PN sequence defined in Section 3.6.4 is used to define an integer

$$\hat{p}_j = \sum_{i=0}^{m-1} 2^{m-i} p_{jm+i}$$

where $p_0, p_1, p_2, \ldots$, are the binary digits of the PN sequence, and $\hat{p}_j \in \{0, 1, \ldots, M-1\}$. For $j \in \{0, 1, \ldots, q_d RS - 1\}$, the j^{th}symbol at the output of the PN spreader is:

$$t_j = (s_{[j/q_d]} + \hat{p}_j) \bmod M$$

where $[x]$ denotes the integer part of x.

3.10 SLOT MAPPER

Each PPM symbol $t_j \in \{0, 1, \ldots, M-1\}$ shall be mapped to a binary vector of length M,

$$\boldsymbol{u}_j = u_{j,0}, u_{j,1}, \ldots, u_{j,M-1}$$

where

$$u_{j,i} = \begin{cases} 1, & \text{if } i = t_j \\ 0, & \text{otherwise} \end{cases}$$

Each $\boldsymbol{u}_j$ contains $M-1$ zeroes and one one.

3.11 GUARD SLOT INSERTION

After each set of M slots $\boldsymbol{u}_j$, $M/4$ guard slots shall be inserted, as shown in Figure 3-10. The result is the slot sequence $\boldsymbol{v} = \boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_{q_d RS-1}$, where each $\boldsymbol{v}_j$ is a vector of length $5M/4$, and

$$v_{j,i} = \begin{cases} u_{j,i}, & i < M \\ 0, & M \le i < 5M/4 \end{cases}$$

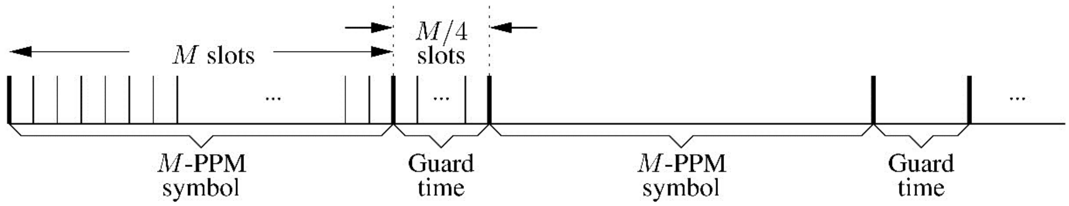

Figure 3-10: Guard slot insertion.

4 UPLINK SIGNALING

The Recommended Standard operates, consistent with Ref. [4], by taking CCSDS TC Transfer Frames as input, and producing a binary vector indicating the positions of pulsed slots at the output. The functional blocks of the architecture are shown in Figure 4-1, along with the notation used in the following sections that define these functions mathematically. It should be understood that the functions need not be implemented explicitly as defined here; any implementation producing the proper pattern of pulsed slots complies with the standard.

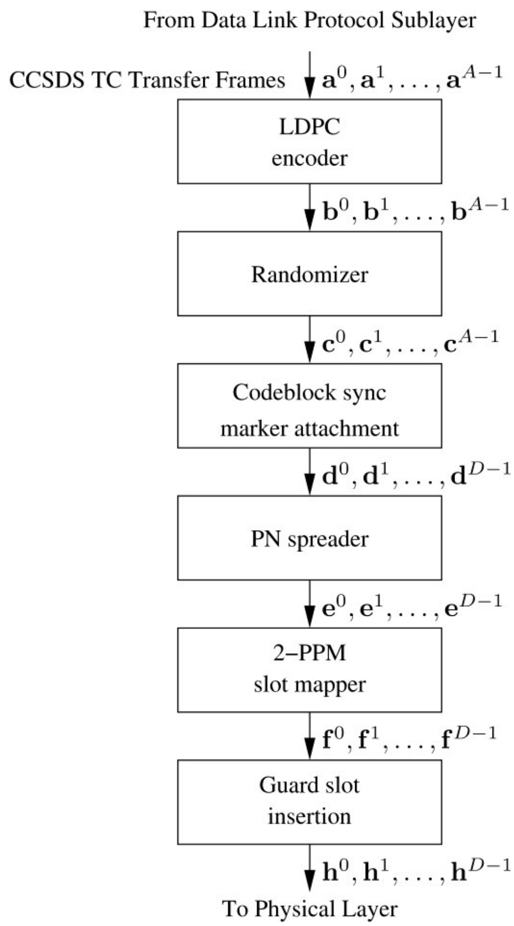

Figure 4-1: Functional diagram for uplink signaling.

4.1 CCSDS TC TRANSFER FRAMES

The input to the coding and synchronization sublayer shall be a sequence of A CCSDS TC Transfer Frames

$$\boldsymbol{a}^0, \boldsymbol{a}^1, \ldots, \boldsymbol{a}^{A-1}$$

The structure of the Transfer Frames shall be as defined in Ref. [4]. The Transfer Frames each shall be 64 binary digits in length. For $i \in \{0, 1, \ldots, A-1\}$, the i^{th} Transfer Frame is denoted

$$\boldsymbol{a}^i = a_0^i, a_1^i, \ldots, a_{63}^i$$

where $a_j^i \in \{0,1\}$ is the j^{th} bit of the i^{th} frame.

4.2 LDPC ENCODER

Each transfer frame $\boldsymbol{a}^i$ shall be encoded using the (128, 64) binary low-density parity-check (LDPC) code defined Section 2.2 of Ref. [5]. For $i \in \{0, 1, \ldots, A-1\}$, the 128 binary symbols of the i^{th} LDPC-encoded Transfer Frame are denoted

$$\boldsymbol{b}^i = b_0^i, b_1^i, \ldots, b_{127}^i$$

where $b_j^i \in \{0,1\}$ is the j^{th} binary digit of the i^{th} codeblock.

4.3 RANDOMIZER

Each codeblock $\boldsymbol{b}^i$ shall be randomized by performing the digit-wise modulo-2 addition of it with a pseudo-noise (PN) sequence produced by a linear feedback shift register. The shift register is initialized to the all-one state at the beginning of each codeblock. The generator polynomial of the PN sequence is as defined in Section 3.6.4.

The i^{th} randomized codeblock is given by

$$\boldsymbol{c}^i = b_0^i \oplus p_0, b_1^i \oplus p_1, \ldots, b_{15119}^i \oplus p_{127}$$

where $\oplus$ represents modulo-2 addition.

4.4 CODEBLOCK SYNCHRONIZATION MARKER

Multiple codeblocks shall be concatenated within one Communications Link Transmission Unit (CLTU), as described in Ref. [4]. Prepended at the beginning of each CLTU is a codeblock synchronization marker (CSM) of $W_u =$ [TBD] binary symbols. The CSM is denoted

$$\boldsymbol{w} = w_0, w_1, \ldots, w_{W_u-1} = 0x\text{[TBD]}$$

After CSM attachment, the j^{th} binary symbol of the i^{th} CLTU is:

$$d_j^i = \begin{cases} w_j, & \text{if } j < W_u \\ c_{j-W}^i, & \text{if } W_u \le j < S_i + W_u \end{cases}$$

Where S_i is the number of LDPC code symbols in the i^{th} CLTU. The sequence of CLTUs is denoted $\boldsymbol{d} = \boldsymbol{d}^0, \boldsymbol{d}^1, \ldots, \boldsymbol{d}^{D-1}$.

4.5 PN SPREADER

Each CLTU shall be PN spread by a factor of q_u by repeating each binary symbol q_u times and digit-wise adding it to a PN sequence. The i^{th} PN-spread CLTU is given by

$$\boldsymbol{e}^i = d_0^i \oplus p_0, d_1^i \oplus p_1, \dots, d_{S_i+W_u-1}^i \oplus p_{S_i+W_u-1}$$

where $p_0, p_1, p_2, \dots$, are the binary digits of the PN sequence defined in Section 3.6.4.

4.6 2-PPM SLOT MAPPER

Each PN spread CLTU is a sequence of binary symbols denoted

$$\boldsymbol{e}^i = (e_0^i, e_1^i, \dots, e_{S_i+W_u-1}^i)$$

To modulate with 2-PPM, each binary symbol e_j^i shall be mapped to a vector of length 2 by

$$\boldsymbol{f}_j^i = (f_{j,0}^i, f_{j,1}^i) = \begin{cases} (1,0), & \text{if } e_j^i = 0 \\ (0,1), & \text{otherwise} \end{cases}$$

and the sequence of slots for the i^{th} CLTU is denoted

$$\boldsymbol{f}^i = (f_{0,0}^i, f_{0,1}^i, f_{1,0}^i, f_{1,1}^i, \dots, f_{S_i+W_u-1,0}^i, f_{S_i+W_u-1,1}^i)$$

4.7 GUARD SLOT INSERTION

After each 2-PPM symbol comprising two slots, a guard time of two slots shall be inserted. The result for the the i^{th} CLTU is the slot sequence

$$\boldsymbol{h}^i = (f_{0,0}^i, f_{0,1}^i, 0, 0, f_{1,0}^i, f_{1,1}^i, 0, 0, \dots, f_{S_i+W_u-1,0}^i, f_{S_i+W_u-1,1}^i, 0, 0)$$

5 MANAGED PARAMETERS

5.1 OVERVIEW

In order to conserve bandwidth on the space link, some parameters associated with synchronization and channel coding are handled by management rather than by inline communications protocol. The managed parameters are those which tend to be static for long periods of time, and whose change generally signifies a major reconfiguration of the synchronization and channel coding systems associated with a particular mission. Through the use of a management system, management conveys the required information to the synchronization and channel coding systems.

In this section, the managed parameters used by synchronization and channel coding systems are listed. These parameters are defined in an abstract sense and are not intended to imply any particular implementation of a management system.

5.2 MANAGED PARAMETERS FOR DOWNLINK SIGNALING

Managed Parameter	Allowed Values
PPM order, M	4, 8, 16, 32, 64, 128, 256
Code rate, r	1/3, 1/2, 2/3
Number of rows, N	Must be a divisor of $15120/\log_2 M$
Shift register length increment, B	NB must be a divisor of $15120/\log_2 M$
Codeblock synchronization marker length, W	[TBD]
PN spreading factor, q_d	1, 2, 3, 4, 8, 16, 32

5.3 MANAGED PARAMETERS FOR UPLINK SIGNALING

Managed Parameter	Allowed Values
PN spreading factor, q_u	1, 2, 3, 4, 8, 16, 32, 64

ANNEX A

SERVICE

(NORMATIVE)

This annex provides service definition in the form of primitives, which present an abstract model of the logical exchange of data and control information between the service provider and the service user. The definitions of primitives are independent of specific implementation approaches.

The parameters of the primitives are specified in an abstract sense and specify the information to be made available to the user of the primitives. The way in which a specific implementation makes this information available is not constrained by this specification. In addition to the parameters specified in this annex, an implementation can provide other parameters to the service user (e.g., parameters for controlling the service, monitoring performance, facilitating diagnosis, and so on).

Overview of the Service

The TM Synchronization and Channel Coding provides unidirectional (one way) transfer of a sequence of fixed-length TM or AOS Transfer Frames at a constant frame rate over a Physical Channel across a space link, with optional error detection/correction.

Only one user can use this service on a Physical Channel, and Transfer Frames from different users are not multiplexed together within one Physical Channel.

A3 SERVICE PARAMETERS

A3.1 FRAME

A3.1.1 The Frame parameter is the service data unit of this service and shall be either a TM Transfer Frame defined in reference [2] or an AOS Transfer Frame defined in reference [3].

A3.1.2 The length of any Transfer Frame transferred on a Physical Channel must be the same, and is established by management.

CCSDS 131.0-B-2 Page A-1 August 2011

CCSDS RECOMMENDED STANDARD FOR TM SYNCHRONIZATION AND CHANNEL CODING

A3.2 QUALITY INDICATOR

The Quality Indicator parameter shall be used to notify the user at the receiving end of the service that there is an uncorrectable error in the received Transfer Frame.

A3.3 SEQUENCE INDICATOR

The Sequence Indicator parameter shall be used to notify the user at the receiving end of the service that one or more Transfer Frames of the Physical Channel have been lost as the result of a loss of frame synchronization.

A4 SERVICE PRIMITIVES

A4.1 GENERAL

A4.1.1 The service primitives associated with this service are: a) ChannelAccess.request; b) ChannelAccess.indication.

A4.1.2 The ChannelAccess.request primitive shall be passed from the service user at the sending end to the service provider to request that a Frame be transferred through the Physical Channel to the user at the receiving end.

A4.1.3 The ChannelAccess.indication shall be passed from the service provider to the service user at the receiving end to deliver a Frame.

A4.2 ChannelAccess.request A4.2.1 Function The ChannelAccess.request primitive is the service request primitive for this service.

A4.2.2 Semantics

The ChannelAccess.request primitive shall provide a parameter as follows: ChannelAccess.request (Frame)

A4.2.3 When Generated

The ChannelAccess.request primitive is passed to the service provider to request it to process and send the Frame.

A4.2.4 Effect On Receipt

Receipt of the ChannelAccess.request primitive causes the service provider to perform the functions described in 2.3.1 and to transfer the resulting channel symbols.

A4.3 ChannelAccess.indication A4.3.1 Function The ChannelAccess.indication primitive is the service indication primitive for this service.

A4.3.2 Semantics

The ChannelAccess.indication primitive shall provide parameters as follows:

ChannelAccess.indication (Frame, Quality Indicator,

Sequence Indicator)

A4.3.3 When Generated

The ChannelAccess.indication primitive is passed from the service provider to the service user at the receiving end to deliver a Frame.

A4.3.4 Effect On Receipt

The effect of receipt of the ChannelAccess.indication primitive by the service user is undefined.

ANNEX B

SECURITY, SANA, AND PATENT CONSIDERATIONS

B1 SECURITY CONSIDERATIONS

B1.1 SECURITY BACKGROUND

It is assumed that security is provided by encryption, authentication methods, and access control to be performed at higher layers (application and/or transport layers). Mission and service providers are expected to select from recommended security methods, suitable to the specific application profile. Specification of these security methods and other security provisions is outside the scope of this Recommended Standard. The coding layer has the objective of delivering data with the minimum possible amount of residual errors. An LDPC, Reed-Solomon, or other code with CRC code must be used to insure that residual errors are detected and the frame flagged. There is an extremely low probability of additional undetected errors that may escape this scrutiny. These errors may affect the encryption process in unpredictable ways, possibly affecting the decryption stage and producing data loss, but will not compromise the security of the data.

B1.2 SECURITY CONCERNS

Security concerns in the areas of data privacy, authentication, access control, availability of resources, and auditing are to be addressed in higher layers and are not related to this Recommended Standard.

B1.4 CONSEQUENCES OF NOT APPLYING SECURITY

There are no specific security measures prescribed for the coding layer. Therefore consequences of not applying security are only imputable to the lack of proper security measures in other layers. Residual undetected errors may produce additional data loss when the link carries encrypted data.

B2 SANA CONSIDERATIONS

The recommendations of this document do not require any action from SANA.

B3 PATENT CONSIDERATIONS

B3.1 POTENTIAL THREATS AND ATTACK SCENARIOS

ANNEX C

ACRONYMS AND TERMS

(INFORMATIVE)

This annex lists key acronyms and terms that are used throughout this Recommended Standard to describe synchronization and channel coding.

C1. ACRONYMS

PPM Pulse Position Modulation

SCPPM Serially-concatenated Convolutionally-coded Pulse Position Modulation

C2. TERMS

Block Encoding: A one-to-one transformation of sequences of length k of elements of a source alphabet to sequences of length n of elements of a code alphabet, $n>k$.

Synchronization and Channel Coding Sublayer: That sublayer of the Data Link Layer used by CCSDS space link protocols which uses a prescribed coding technique to reliably transfer Transfer Frames through the potentially noisy Physical Layer.

Channel Symbol: The unit of output of the innermost encoder.

ANNEX D

INFORMATIVE REFERENCES

[A1] “Coded Modulation for the Deep-Space Optical Channel: Serially Concatenated Pulse-Position Modulation”, B. Moision and J. Hamkins, IPN Progress Report 42-161 May 15, 2005.

[A2] Information Technology—Open Systems Interconnection—Basic Reference Model: The Basic Model. International Standard, ISO/IEC 7498-1. 2nd ed. Geneva: ISO,1994.

[A3] *Procedures Manual for the Consultative Committee for Space Data Systems*. CCSDS A00.0-Y-9. Yellow Book. Issue 9. Washington, D.C.: CCSDS, November 2003.

[A4] LLCD-ICD-002, “Lunar Lasercom Space Terminal to Lunar Lasercom Ground Terminal Interface Control Document,” Revision 3.0, July 5, 2011.

ANNEX E– BACKUP MATERIAL