ESA PSS-04-103 Issue 1
September 1989

Telemetry channel coding standard

Prepared by the
Standards Approval Board (STAB)
for Space Data Communications

Approved by:
The Inspector General, ESA

european space agency / agence spatiale européenne
8-10, rue Mario-Nikis, 75738 PARIS CEDEX, France

Published by ESA Publications Division,
ESTEC, Noordwijk, The Netherlands.

Printed in the Netherlands.

ESA Price code: E0
ISSN 0379 - 4059

SPACE DATA COMMUNICATIONS PROCEDURES, SPECIFICATIONS & STANDARDS

Space Data Communications is the subject of the PSS-04 branch of the ESA Procedures, Specifications & Standards (PSS) series. This branch is further divided into two subbranches:

- the **Space Link Standards and Protocols** subbranch (document reference nos.: ESA PSS-04-1XX);

- the **Spacecraft Data Interfaces and Protocols** subbranch (document reference nos.: ESA PSS-04-2XX).

The Space Data Communications PSS documents have the purpose of ensuring the compatibility of spacecraft TT&C subsystems with the relevant ESA infrastructure (i.e. the ESA (ESOC) tracking and data-communication network and the ESA (ESTEC) satellite check-out facilities).

DOCUMENT CHANGE RECORD

Issue number and date	Sections affected	Remarks
Issue No. 1 September 1989	All	– Directly derived from the **Recommendation for Space Data System Standards: Telemetry Channel Coding,** CCSDS 101.0-B-2 'Blue Book', January 1987.(*) – This document supersedes the previous version, ref.: TTC-A-03 Issue No.1, December 1979.

NOTE (*): ERRATUM

The following errors have been found in CCSDS 101.0-B-2, January 1987:

— on page 4-6, 5th line: this statement is erroneous, as discussed in Appendix B of this Standard.

— on page B-3: the second table is erroneously a repeat of the first table. The correct table can be found at the top of page B-4.

REFERENCES

[1] **Telemetry Channel Coding**, Recommendation **CCSDS 101.0-B-2,** Issue 2, 'Blue Book', Consultative Committee for Space Data Systems, January 1987.

[2] **Packet Telemetry Standard (ESA PSS-04-106),** Issue 1, January 1988, European Space Agency.

[3] **Radio Frequency and Modulation Standard (ESA PSS-04-105),** Issue 1, December 1989, European Space Agency.

PAGE INTENTIONALLY LEFT BLANK

CONTENTS

ILLUSTRATIONS

TABLES

APPENDICES

1. PURPOSE AND SCOPE

1.1 PURPOSE

The purpose of this Standard is to establish uniform requirements for the implementation of spacecraft telemetry channel coding systems by ESA. To this end the Standard:

- establishes a common framework and provides a common basis for the coding structures of ESA spacecraft telemetry streams.
- allows ESA spacecraft projects to proceed coherently with the implementation of space-borne telemetry systems which are compatible with the ESA ground systems.
- potentially provides a high level of commonality between both ESA spaceborne and ground-based telemetry systems and the ground systems of other CCSDS[(1)] Agencies.
- provides information on the performance of the various telemetry channel coding schemes specified by this Standard.

1.2 SCOPE

This Standard belongs to the Space Link Standard & Protocols sub-branch of the ESA PSS branch for Space Data Communications. It supersedes the **Telemetry Coding Standard (TTC-A-03)**.

The Telemetry Channel Coding Standard does not require that coding be used on all missions, since the selection of a coding scheme is governed by the requirements of the telemetry channel performance. However, for those projects for which it is planned to use coding, this Standard defines the requirements for digital telemetry channel coding techniques to be used on board the spacecraft, whenever the ESA ground network standard facilities are to be used. This does not prohibit projects from adhering to future ESA channel coding standards, e.g. those that will be used to specify the Data Relay Satellite (DRS) support systems. The ESA ground network standard facilities support, at present:

NOTE (1): CCSDS: Consultative Committee for Space Data Systems. This Standard is directly derived from a technical recommendation on Telemetry Channel Coding issued by the CCSDS (Reference [1]).

(a) Rate 1/2 Convolutional Code (Non-multiplexed)

This code, which can be used alone, is particularly suited to channels with predominantly Gaussian noise. The bandwidth expansion, when compared with the uncoded channel, is 100%. The on-board encoder is simple to implement, and substantial coding gains are obtained with decoders using soft decisions. Decoding failures are not signalled and produce error bursts.

(b) Reed-Solomon Code

The Reed-Solomon code, which can be used alone, is a powerful burst error correcting block code. The bandwidth expansion, when compared with the uncoded channel, is 14%. The on-board implementation is more complex than for the convolutional code, whereas the coding gain is comparable at low error probabilities. However, the decoder signals its failure to output an error-free block of data.

(c) Concatenated Codes

When better performance is needed than can be provided by either the convolutional code or the Reed-Solomon code used alone, a concatenation of the two codes can be used. The Reed-Solomon code is used as the 'outer code' (first encoding) concatenated with the convolutional code as the 'inner code' (second encoding). The bandwidth expansion, when compared with the uncoded channel, is 128%. The outer decoder signals its failure to output error free data.

It should be noted that these codes, and in particular the Reed-Solomon code, are fully compatible with the data structures of Reference [2]. Finally, a standard **pseudo-randomising** technique is specified. This technique allows a minimum symbol transition density to be (in practice) guaranteed on the link.

2. APPLICABILITY

2.1 GENERAL

This Standard applies to all spacecraft using the ESA ground network standard facilities and services.

The capabilities provided within the constraints of this Standard will accommodate the requirements of a great variety of applications for telemetry subsystems, thus providing a universal basis for cost-effective and technically compatible development of telemetry systems in different projects.

It should be noted that this Standard is compatible with the modulation techniques specified in Reference [3]. However, when used with QPSK modulation, certain combinations of encoding and types of QPSK are preferred, in order to achieve the performance defined in this Standard. Such cases should be checked with the supporting network (e.g. ESOC).

2.2 EXCEPTIONS

In exceptional cases, owing to mission-specific requirements, some deviations from this Standard may be warranted. Waivers to any requirements set forth in this Standard may be obtained only after:

- the technical and/or operational advantages of such deviations have been demonstrated, and/or
- it has been shown that the intended change is in line with the existing systems.

Requests for waivers should be addressed by the Project Manager to the ESA Standards Approval Board (STAB) for Space Data Communications. Such requests should be submitted as early as possible, preferably during the study phase of the project.

PAGE INTENTIONALLY LEFT BLANK

3. SYSTEM OVERVIEW

3.1 BIT NUMBERING CONVENTION AND NOMENCLATURE

The following **CAUTION** should be observed when interpreting the bit numbering convention which is used throughout this standard:

CAUTION

In this document, the following convention is used to identify each bit in a forward-ordered 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 called 'Bit 1', and so on up to 'Bit N-1'. When the field is used to express a binary value (such as an integer), the Most Significant Bit (MSB) shall be the first transmitted bit of the field (i.e. 'Bit 0').

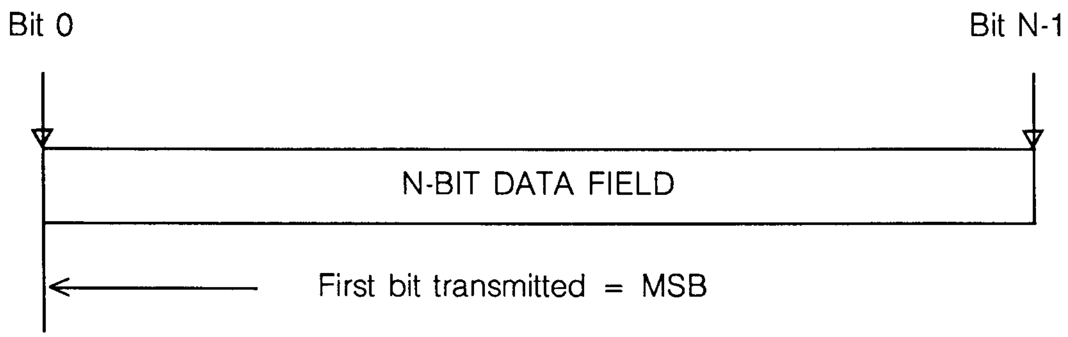


Figure 3.1 BIT NUMBERING

Spacecraft data fields are often grouped into 8-bit 'words' which conform to the above convention. The following nomenclature is used throughout this Standard to describe this grouping:

OCTET = 8-bit word

In this document, the above-defined numbering convention for identifying a bit is also used for identifying each octet in a forward-ordered N-octet string.

3.2 CHANNEL ACCESS DATA UNIT STRUCTURES AND CHANNEL CONFIGURATIONS

The Channel Access Data Unit (CADU) structures and channel configurations for four different schemes are given below. The telemetry Transfer Frame and the Attached Synchronisation Marker shown in the figures are defined in Reference [2]. Frame synchronisation may in principle take place after convolutional decoding, but the configuration shown in Figure 3.5 (concatenated coding scheme) gives optimum performance. For implementation reasons, the same synchronisation position is used for all coding schemes.

The **Randomiser** is a function which provides adequate symbol transitions, independent of the Transfer Frame contents, to obtain steady symbol synchronisation on ground. It should not be confused with periodic channel interleaving which converts burst errors into random bit errors.

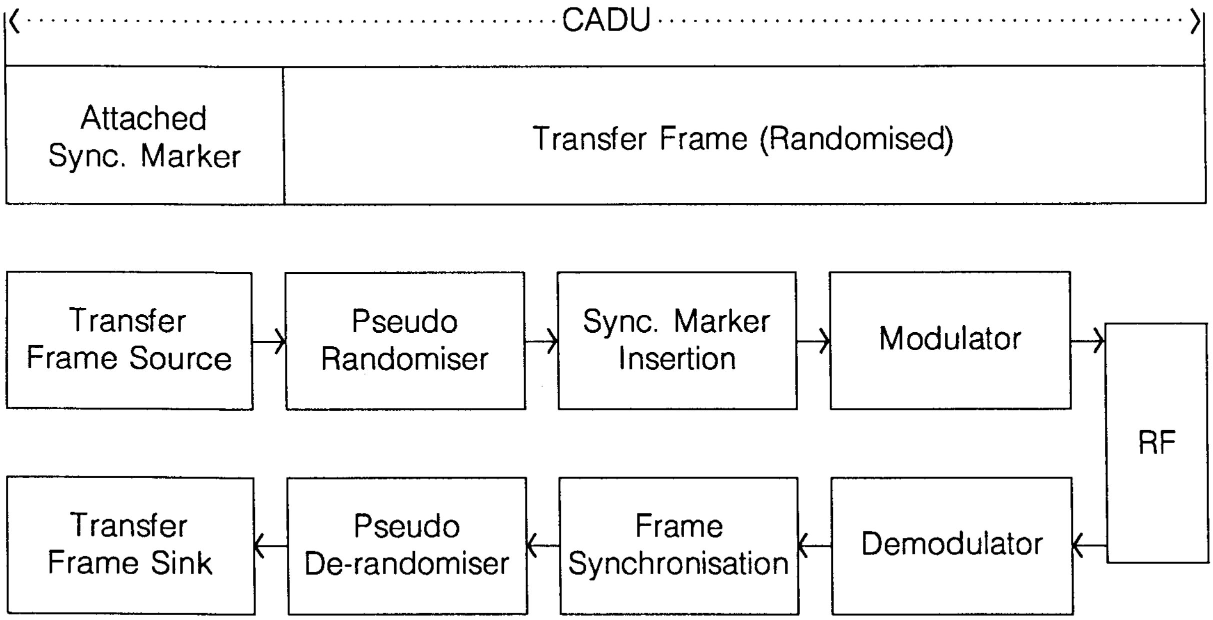


Figure 3.2 CHANNEL WITH NO CODING

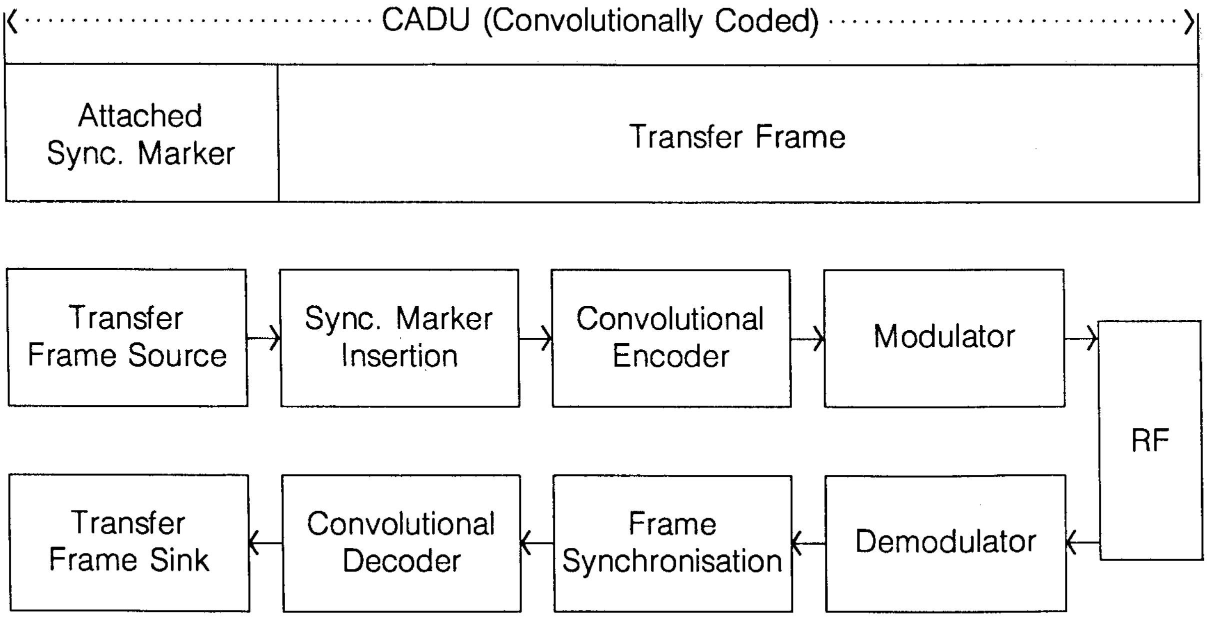


Figure 3.3 CHANNEL WITH CONVOLUTIONAL CODE

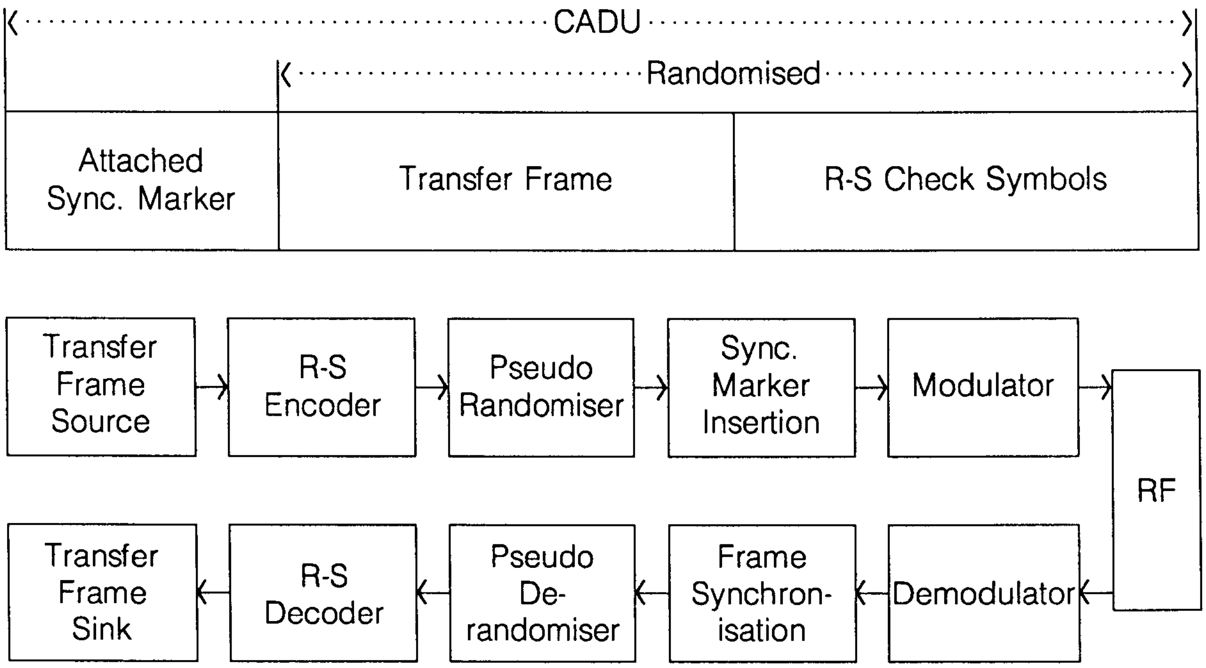


Figure 3.4 CHANNEL WITH REED-SOLOMON CODE

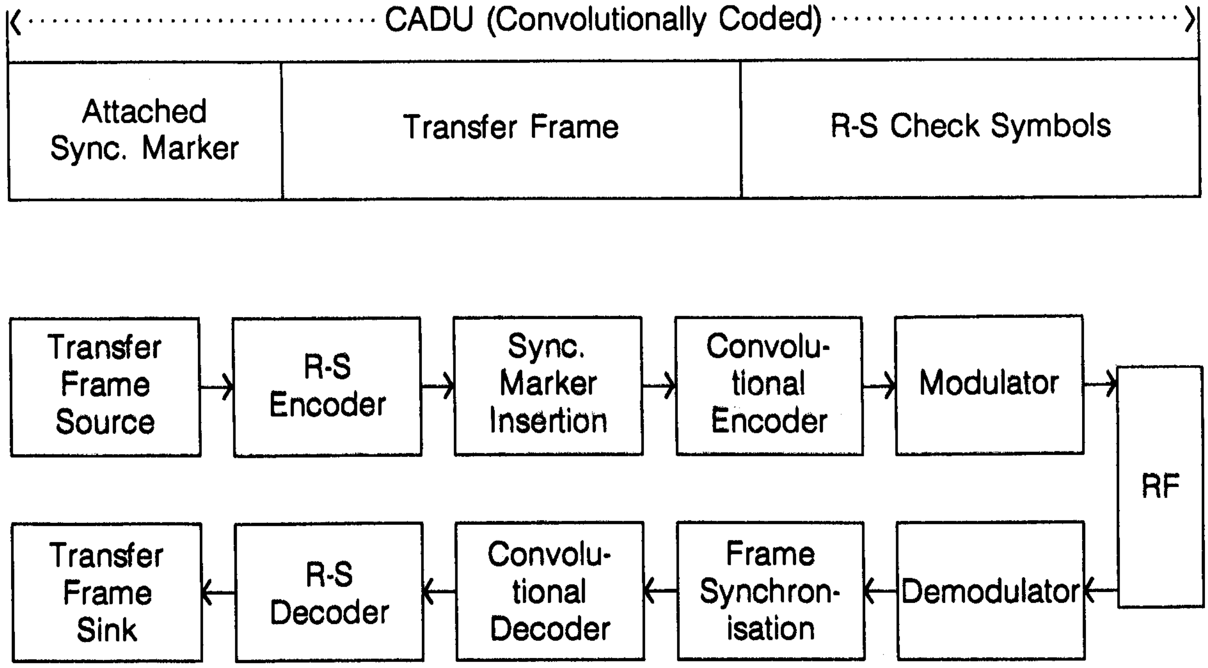


Figure 3.5 CHANNEL WITH CONCATENATED CODES

4. CONVOLUTIONAL CODING STANDARD

On channels with predominantly Gaussian noise, a rate 1/2, constraint length 7 convolutional code can be used. This code is particularly efficient when used with soft decisions. The standard convolutional code specified here may be used alone or as the inner code in a concatenated system. In a concatenated system, the Reed-Solomon code described in Section 5 shall be used as the outer code.

This Standard specifies only a non-multiplexed convolutional code. The code is transparent and has the following characteristics:

(1) Nomenclature	: Convolutional code
(2) Code rate	: 1/2 bit per symbol
(3) Constraint length	: 7 bits
(4) Connection vectors	: G1 = 1111001 G2 = 1011011
(5) Phase relationship	: G1 is associated with first symbol
(6) Symbol inversion	: On output path of G2

The encoding rule specified in this Standard permits the generation of a non-systematic code.[(2)] This encoding rule is defined by the following equations:

$$s1(t) = i(t) + i(t-1) + i(t-2) + i(t-3) + i(t-6) \quad \text{modulo 2}$$

$$s2(t) = i(t) + i(t-2) + i(t-3) + i(t-5) + i(t-6) + 1 \quad \text{modulo 2}$$

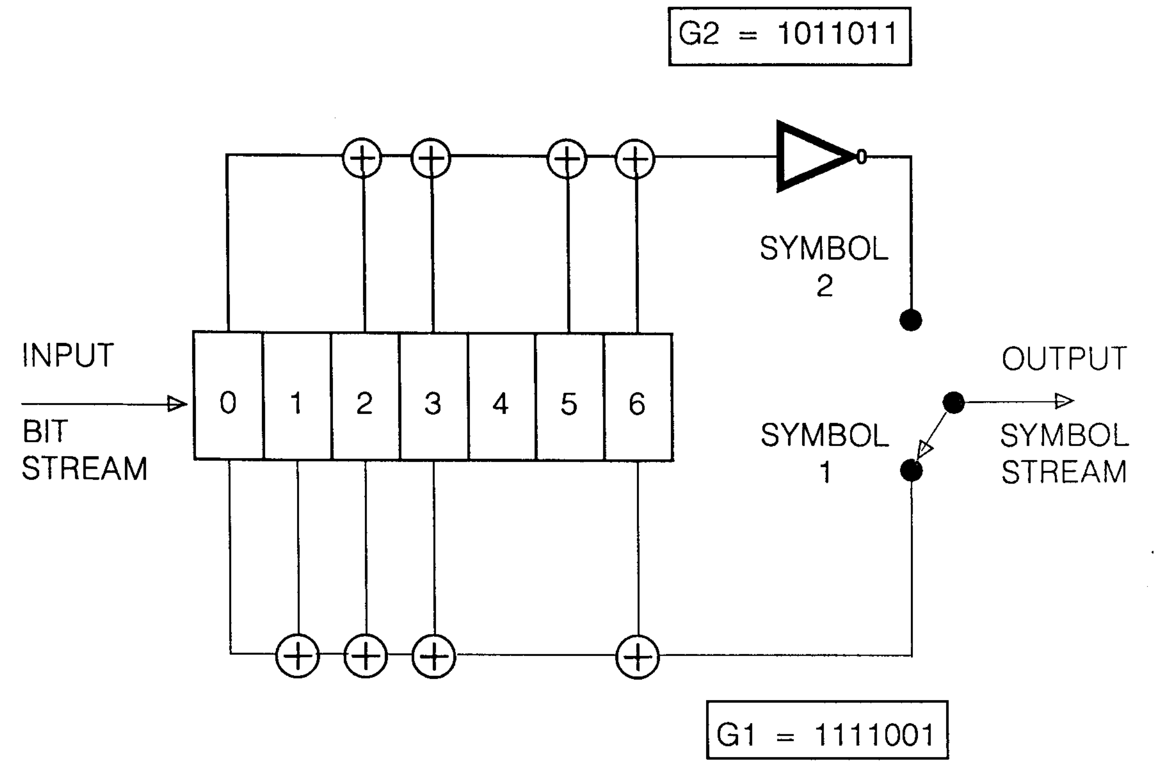
where $s1$ is the first output symbol, $s2$ the second output symbol and $i(t)$ is the input information at time, t. The sign + represents the modulo 2 addition.

A possible implementation of the encoder is shown in Figure 4.1.

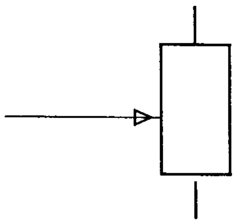
NOTE (2): A slightly different convention was specified in TTC-A-03 of December 1979, which concerns G1 and G2 . The new convention is fully compatible with Reference [1]. ESA will, in the future, suspend the support of the convention laid down in TTC-A-03.

When BPSK modulation systems are used, NRZ-M or NRZ-L may be used as a modulating waveform. If the user contemplates conversion of his modulating waveform from NRZ-L to NRZ-M (differential encoding), such conversion should be performed on board at the input to the convolutional encoder. Correspondingly, the conversion on the ground from NRZ-M to NRZ-L (differential decoding) should be performed at the output of the convolutional decoder. This avoids unnecessary link performance loss.

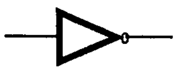
CAUTION: When a fixed pattern in the symbol stream (e.g. Attached Sync. Marker) is used to provide node synchronisation for the Viterbi decoder, care must be taken to account for any translation of the pattern due to the modulating waveform conversion.



NOTES

(1)  = single bit delay

(2)  = modulo 2 adder

(3)  = inverter

(4) For each bit at the input of the encoder, the symbol switch samples both symbol outputs (G1 symbol first).

(5) The output of G2 is complemented to ensure that there are sufficient symbol transitions in the case of a steady state (all zeros or all ones) at the input of the encoder.

Figure 4.1 POSSIBLE CONVOLUTIONAL ENCODER IMPLEMENTATION

PAGE INTENTIONALLY LEFT BLANK

5. REED-SOLOMON CODING STANDARD

5.1 INTRODUCTION

The purpose of the Reed-Solomon (R-S) code is to provide the virtually error-free channel required to support efficient automated ground handling of space mission data. The R-S code has a powerful burst error correcting capability and is consequently well suited for concatenation (as outer code) with the convolutional code (as inner code) defined in Section 4. The R-S code may, however, also be used alone or, in the future, with other inner codes.[(3)]

The encoding of the R-S code is merged with the segmenting of the data into Transfer Frames. Each frame consists of I interleaved R-S code words. Each Transfer Frame is preceded by an Attached Synchronisation Marker, which in addition to frame synchronisation, also provides synchronisation for the R-S decoder. The Attached Synchronisation Marker does not participate in the encoding process.

5.2 CODE SPECIFICATION

The parameters of the selected R-S code are as follows:

(1) $J = 8$ bits per R-S symbol

(2) $N = 2^J - 1 = 255$ R-S symbols per R-S codeword

(3) The encoding is systematic, the first $K = 223$ R-S symbols containing the information symbols (input data) and the last $N - K = 32$ symbols constituting the check symbols.

NOTE (3): Given the symbol (octet) basis of the R-S code, it is important that channel errors be concentrated in as few symbols as possible. When the R-S code is used alone in a burst-error environment (e.g. with RFI), the use of some types of channel interleaver (such as the Periodic Channel Interleaver (PCI) used in NASA's TDRSS) will degrade performance by increasing the number of R-S symbols in error. When the R-S code is concatenated with a random-bit-error-correcting inner code, the use of such an interleaver may be necessary to protect the inner decoder. The choice of interleaver is dependent on a variety of operational factors, and potential users should contact appropriate experts in the Agency for advice.

(4) The decoder shall be able to correct all combinations of $E = (N - K)/2 = 16$ symbol errors.

(5) The generator polynomial of the code is:

$$g(z) = \prod_{j=112}^{143} (z + b^j)$$

where b is a root of the irreducible polynomial:

$$x^8 + x^6 + x^4 + x^3 + x^2 + x + 1$$

over the binary field. The highest power of z is transmitted first. Since the generator polynomial is symmetric, this convention has no effect on the encoding.

(6) The field GF (256) is represented as a linear vector space. The first eight powers of b form the basis. The representation of the field is defined by the octets representing 1, b, b^2, . . ___ , b^7 (in dual basis). These octets are:

	$z^0 \ldots\ldots z^7$
1	01111011
b	01111001
b^2	00101011
b^3	00111111
b^4	00001001
b^5	10000111
b^6	01011111
b^7	00110111

z_0 being transmitted first.

Further discussion on the symbol representation is contained in Appendix B: **Dual Basis Representation of Reed-Solomon Symbols**.

5.3 CHANNEL ACCESS DATA UNIT STRUCTURE

5.3.1 General Specification

The Transfer Frame has a maximum length of $223 * I$ information symbols, where I is the interleaving depth. When the frame is R-S encoded, the information octets are encoded as $I *$ R-S codewords, where symbol numbers $i + j * I$ ($0 \leq i < I$ and $0 \leq j < 255$) belong to codeword i. The result of the encoding process is a Codeblock in which the encoded Transfer Frame appears unchanged as the first $223 * I$ symbols, followed by $32 * I$ check symbols.

The interleaving depth, I, may be selected from the six values given in Table 5.1, which provides the corresponding standard lengths for Transfer Frame and Codeblock.

Interleaving Depth I	Transfer Frame length	Codeblock length
$I = 1$	223 octets	255 octets
$I = 2$	446 octets	510 octets
$I = 3$	669 octets	765 octets
$I = 4$	892 octets	1020 octets
$I = 5$	1115 octets	1275 octets
$I = 8$	1784 octets	2040 octets

Table 5.1 STANDARD INTERLEAVING DEPTHS AND CODEBLOCK LENGTHS

The Channel Access Data Unit (CADU) structure for a standard Transfer Frame is illustrated in Figure 5.1.

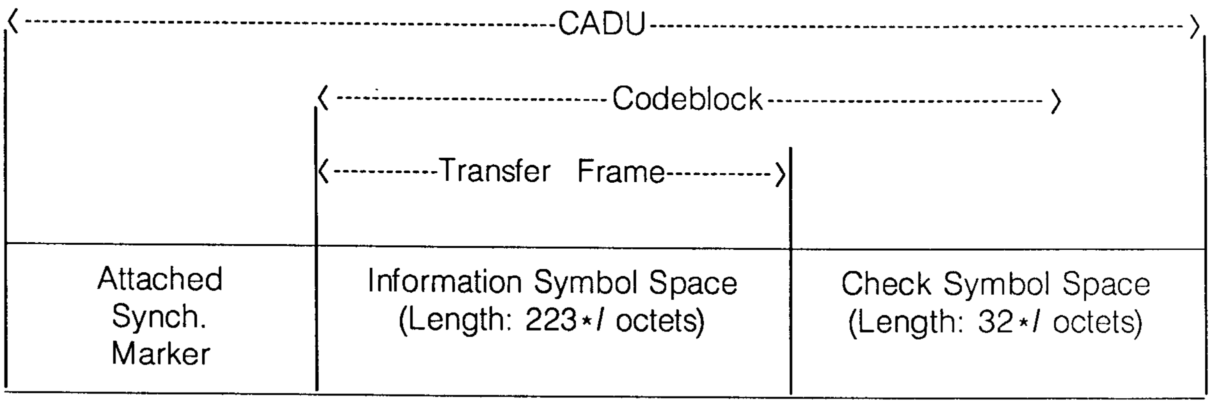


Figure 5.1 TRANSMITTED CHANNEL ACCESS DATA UNIT STRUCTURE

A more detailed picture of the interleaving process is presented in Figure 5.2. The row number corresponds to the symbol number within a codeword. The column number is the number of the codeword within the Codeblock. The number in the matrix is the symbol number in the Codeblock. All counts start with 0 and symbol number 0, and bit number 0 is transmitted first.

	$i=0$	$i=1$	$i=2$	.....	$i=(I-1)$
$J=0$	0	1	2	.....	$I-1$
$J=1$	I	$I+1$	$I+2$	.....	$2*I-1$
:	:	:	:	.....	:
$J=m$	$m*I$	$m*I+1$	$m*I+2$	.....	$(m+1)*I-1$
$J=222$	$222*I$	$222*I+1$	$222*I+2$	.....	$223*I-1$
$J=223$	$223*I$	$223*I+1$	$223*I+2$	.....	$224*I-1$
$J=254$	$254*I$	$254*I+1$	$254*I+2$	.....	$255*I-1$

Figure 5.2 INTERLEAVING PROCESS

The rows $J=0$ up to and including $J=222$ contain the information symbols; the last 32 rows contain the check symbols.

5.3.2 Accommodation of Shorter Transfer Frame Lengths

Transfer Frame lengths shorter than the standard lengths given in Table 5.1 may be accommodated by shortening the Codeblock length accordingly. To achieve this, the necessary number of leading information octets are 'suppressed' from the Information Symbols Space (see Figure 5.1, for example). However, since the R-S code is a block code, both encoder and decoder must always operate on a full Codeblock basis. To achieve this, the encoder assumes that the 'suppressed' leading information octets are all zeros, but does not transmit them. At the receiving end, the decoder assumes that the same 'suppressed' leading information octets have been received as all zeros.

5.3.3 Synchronisation

Codeblock synchronisation of the R-S decoder is achieved by means of the Attached Synchronisation Marker associated with each Codeblock. The standard 32-bit marker pattern shall be as follows:

0001 1010 1100 1111 1111 1100 0001 1101

First transmitted bit (bit 0) — Last transmitted bit (bit 31)

This pattern may be represented in hexadecimal notation as:

1 A C F F C 1 D

This bit pattern, which is also that specified for the Transfer Frame (see Reference [2] for details), was chosen to provide good synchronisation properties with a low false alarm probability in a noisy channel under the following conditions: True as well as complemented data sense, forward as well as reverse time ordering, and synchronisation directly in the bit domain as well as the symbol domain (as translated by the specified convolutional code of Section 4).

The marker repeats at every transmitted codeblock length plus 32 bits. For $I = 4$ and full-length codeblock, the marker repeats every 1024 octets.

5.3.4 Ambiguity Resolution

The ambiguity between true and complemented data must be resolved so that only true data are provided to the R-S decoder. Data in NRZ-L form are normally resolved by means of the 32-bit Attached Synchronisation Marker while NRZ-M data are self-resolving.

6. SYMBOL TRANSITION DENSITY

6.1 GENERAL

To ensure recovery of the symbol clock by the ground demodulators, the symbol transition density of the coded channel must exceed a value specified in Reference [3]. This subject is also partially addressed in Reference [2] (case of the 'IDLE' state of the Virtual Channel, with absence of data), which places the responsibility of ensuring adequate symbol synchronisation with the Project. Various solutions are proposed by Reference [2], such as a specific 'filler pattern', split-phase modulation or NRZ-L with convolutional coding. This Standard specifies two solutions to the symbol transition density problem which are independent of the telemetry data contents. These two solutions are:

(a) to convolutionally encode the channel with the code specified in Section 4, and

(b) to 'pseudo-randomise' the channel symbol stream with the symbol transition generator described in the next subsection.

When BPSK is used, only one of the two solutions may be selected. Simultaneous use of convolutional coding and 'pseudo-randomising' is not supported.

When QPSK is used (whether convolutional coding is used or not), 'pseudo-randomising' may be mandatory for certain I and Q channel multiplexing schemes.

6.2 SYMBOL TRANSITION GENERATOR (PSEUDO-RANDOMISER)

CAUTION: The symbols referred to hereafter are one bit wide.

To ensure adequate symbol transition density when Transfer Frames or R-S Codeblocks are modulated onto the physical channel without convolutional coding, a pseudo-random sequence shall be synchronously 'exclusively OR'ed' with each symbol of the Transfer Frames or R-S Codeblocks. The random sequence shall be generated with the aid of the polynomial given below:

$$h(x) = x^8 + x^7 + x^5 + x^3 + 1$$

This sequence repeats after 255 symbols and the generator is re-initialised to an all-ones state during each Attached Synchronisation Marker period. The first 40 symbols of the pseudo-random sequence are shown below. The left-most symbol is the first symbol of the sequence and is 'exclusively OR'ed' with the first symbol of the Transfer Frame or R-S Codeblock. (Note: The Attached Synchronisation Marker is not part of the Transfer Frame or the R-S Codeblock).

1111 1111 0100 1000 0000 1110 1100 0000 1001 1010

This sequence may be represented in hexadecimal notation as:

FF480EC09A

7. PERFORMANCE OF VARIOUS CODING SCHEMES

The performance curves presented in this section are theoretical curves referring to the input of the decoder. They do not include any technological losses introduced by down-converters or demodulators. It is assumed that the synchronisation is perfect for the convolutional decoder and the R-S decoder. The convolutional decoder is assumed to be based on the Maximum Likelihood Decoding (Viterbi Decoding or VD) soft-decision algorithm and to have an eight-level soft bit quantisation correctly adjusted. The convolutional code is nonmultiplexed.

Four sets of performance curves are provided. Each set shows the performance of the following schemes:

PSK	Coherent biphase PSK (no coding)
R-S	PSK and standard Reed-Solomon coding
VD	PSK and standard convolutional coding
VD + R-S	PSK and convolutional coding concatenated with R-S coding

The performance curves are given for an ideal coherent bi-phase PSK modulated on a channel with additive white Gaussian noise. The error probabilities are plotted against the E_b/N_o measured in dB, where E_b is the energy per decoded information bit and N_o is the one-sided noise spectral density. The curves have been simulated down to a frame loss probability of 10^{-4}. The low error rate part of the curves has been calculated on the basis of an analytical model.

The user should note that performance presented in terms of 'Probability of Bit Error' does not make much sense in the era of packetised telemetry, as it is only for uncoded PSK that a simple relationship between Probability of Bit Error and Probability of Frame Error (i.e. Frame Loss) exists. It is the Probability of Frame Loss which is of interest when packet loss has to be calculated (see Reference [2]). The curves of Figure 7.1 are therefore essentially given for completeness of information. The curve for the concatenated codes (VD + R-S) has been drawn with an assumed interleaving depth of $I = 8$.

For convolutionally encoded data, it is still possible to calculate the Probability of Frame Loss from the Probability of Bit Error with a reasonable accuracy, as the the average burst error length and density can be predicted for a certain E_b/N_o (the burst characteristics of the the Viterbi decoder have been taken into account in the computation of the VD curves). For both the PSK and VD schemes, the presence or absence of bit errors in a Transfer Frame must be detected by the Frame Check Sequence as specified in Reference [2]. The presence of one (or more) bit error signifies that the frame is lost.

When Reed-Solomon coding is involved, the Probability of Bit Error becomes useless in practice, as the decoding process determines whether the number of erroneous symbols is correctable (less than 17 errors) or in excess of the decoder correction capabilities (more than 16 errors, with variations to more than 1000 bit errors per R-S Codeblock, depending on the distribution of the errors). Status information concerning the quality of the decoded data is provided by the decoding process when Reed-Solomon coding (alone or concatenated with convolutional coding) is used.

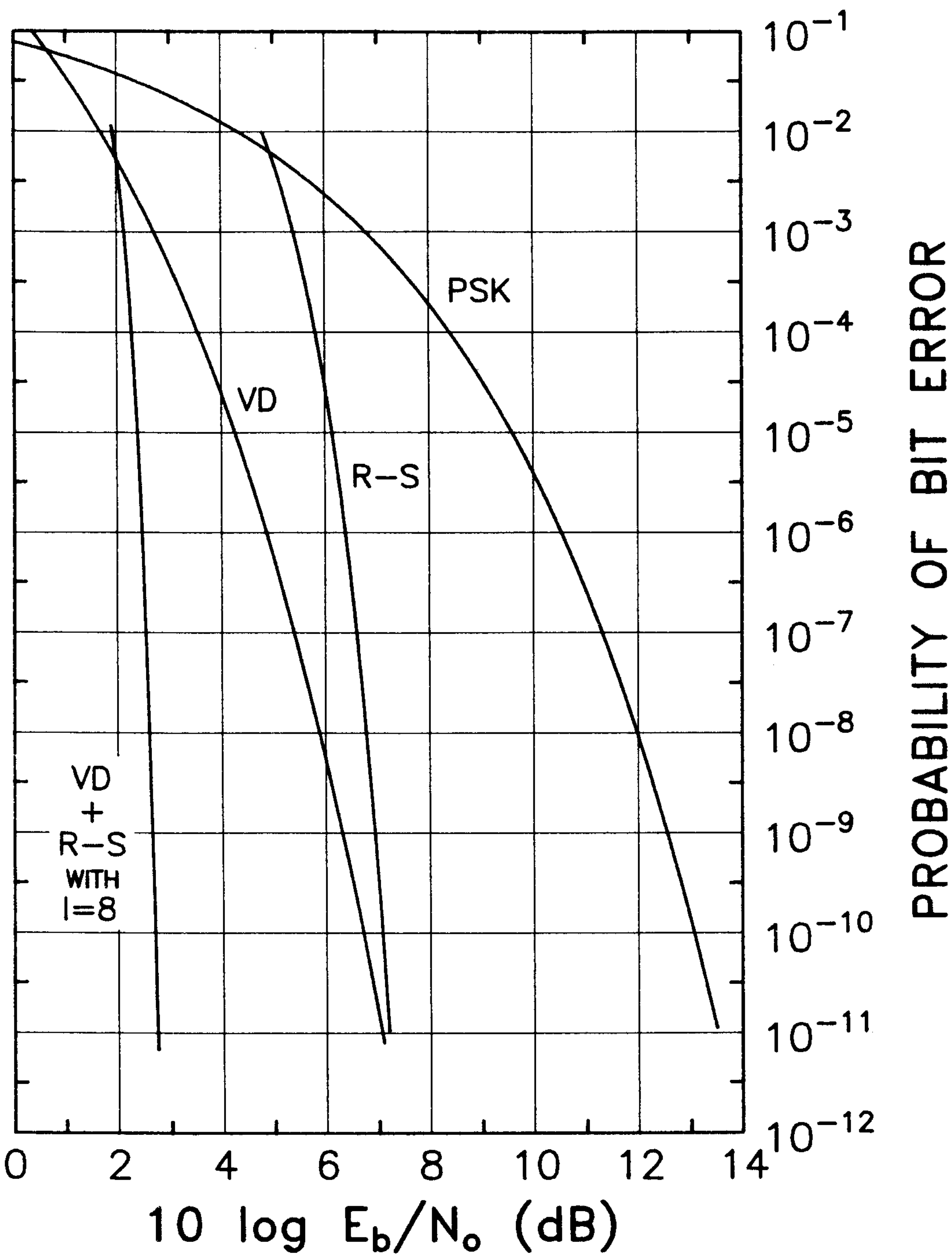


Figure 7.1 PROBABILITY OF BIT ERROR VERSUS E_b/N_o FOR VARIOUS CODING SCHEMES

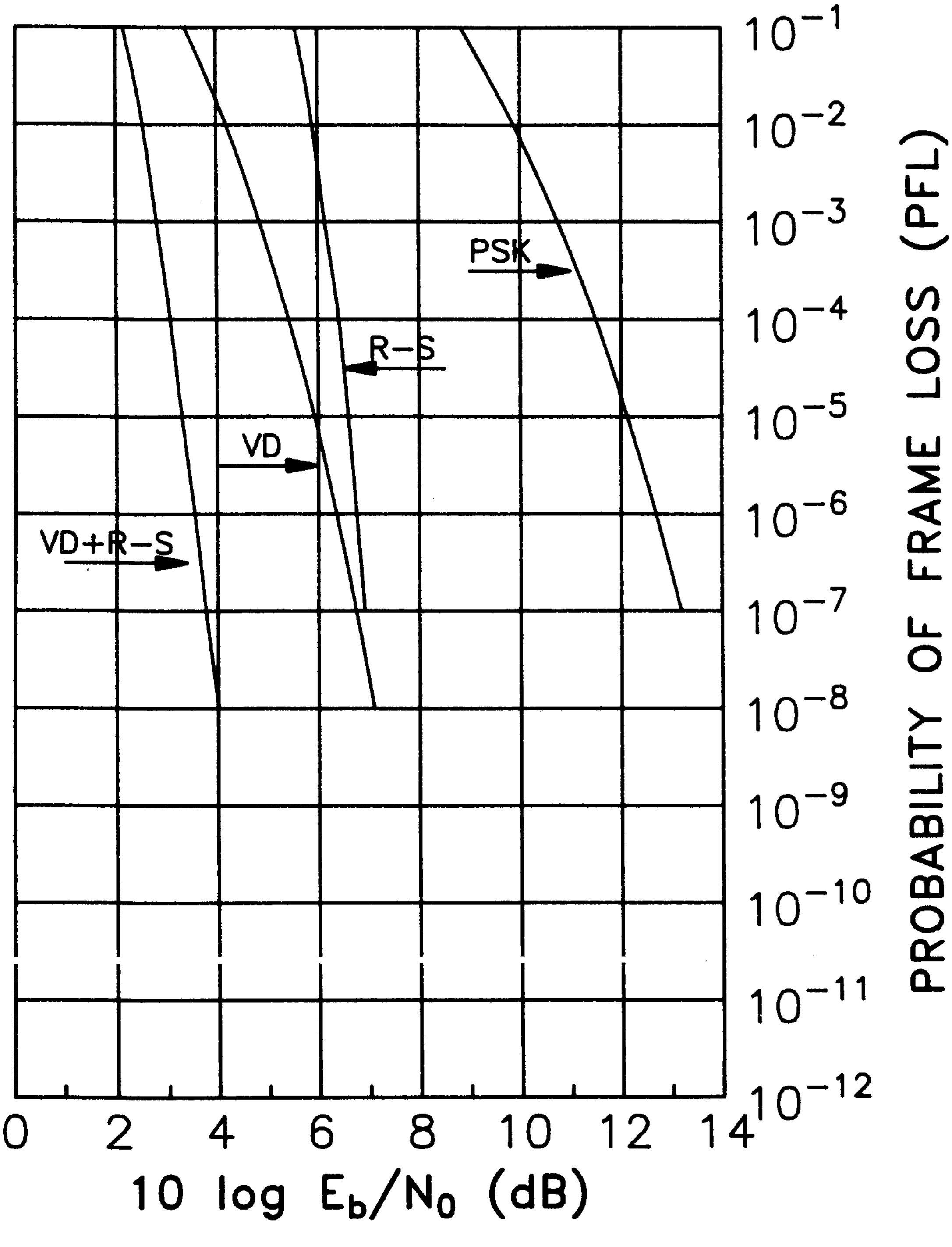


Figure 7.2 PROBABILITY OF FRAME LOSS (PFL) VERSUS E_b/N_0 FOR VARIOUS CODING SCHEMES. INTERLEAVING DEPTH I = 1 (CORRESPONDING TO FRAME LENGTH OF 255 OCTETS) FOR R-S AND VD + R-S SCHEMES.

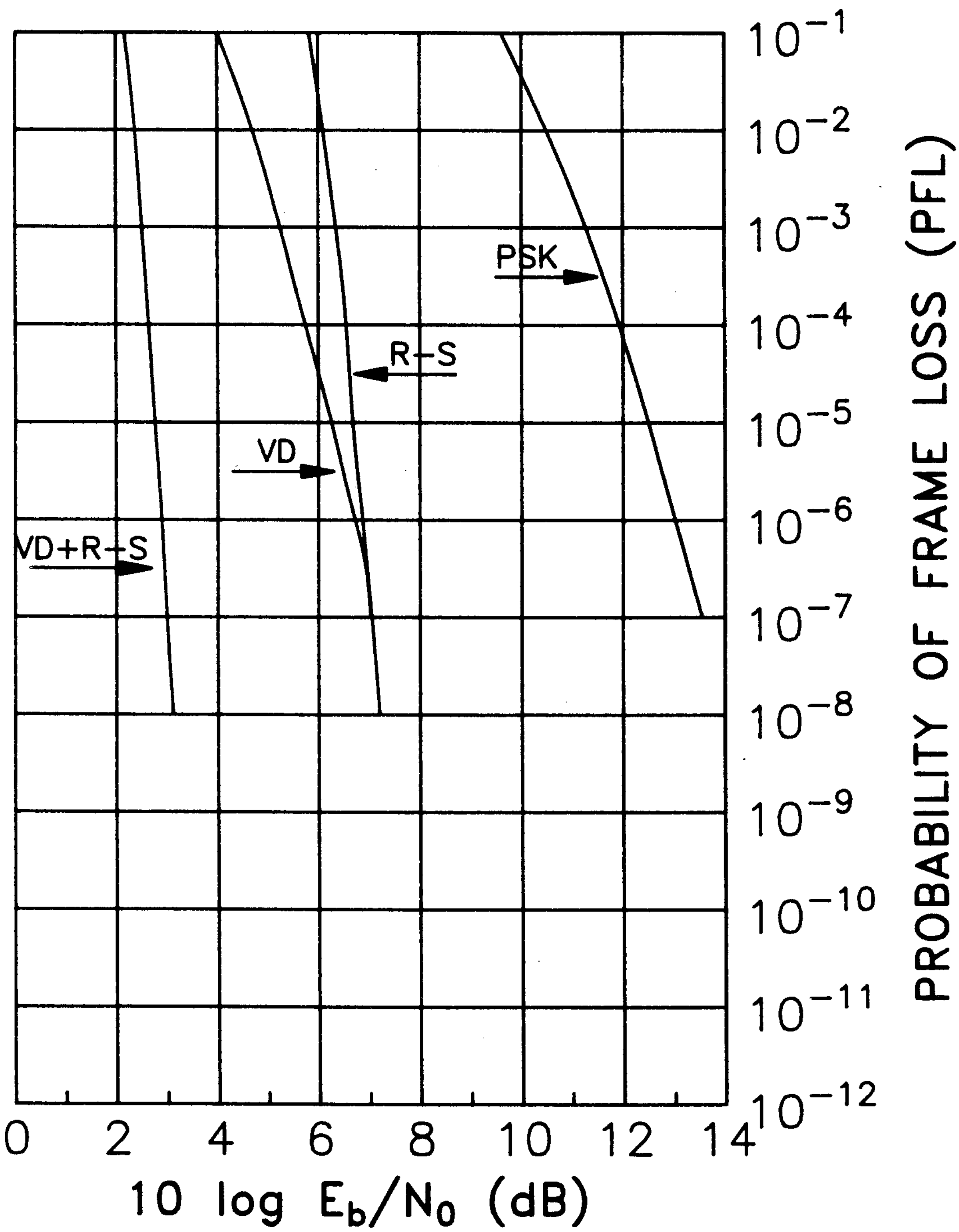


Figure 7.3 PROBABILITY OF FRAME LOSS (PFL) VERSUS E_b/N_0 FOR VARIOUS CODING SCHEMES. INTERLEAVING DEPTH I = 5 (CORRESPONDING TO FRAME LENGTH OF 1275 OCTETS) FOR R-S AND VD + R-S SCHEMES.

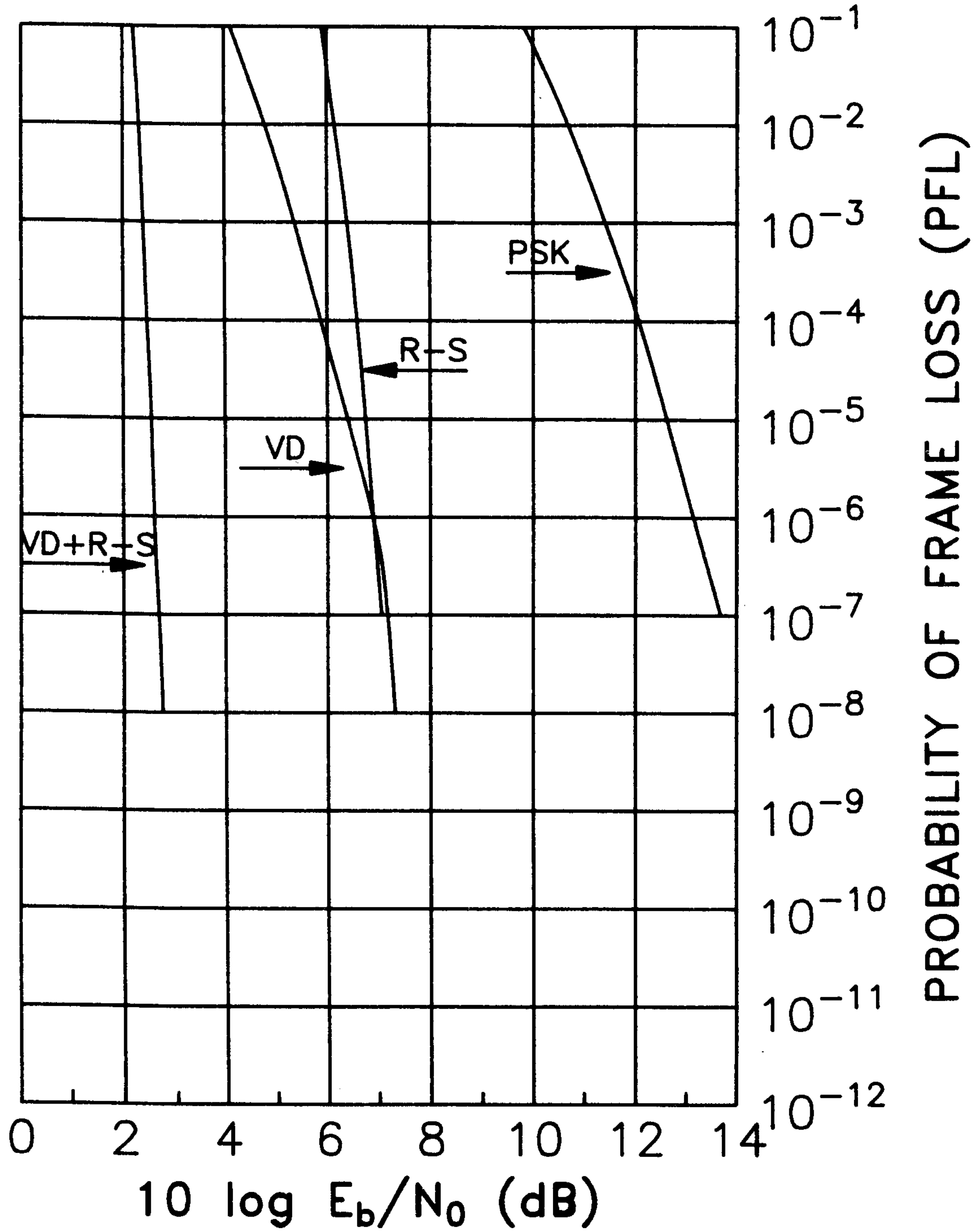


Figure 7.4 PROBABILITY OF FRAME LOSS (PFL) VERSUS E_b5N_o FOR VARIOUS CODING SCHEMES. INTERLEAVING DEPTH I = 8 (CORRESPONDING TO FRAME LENGTH OF 2040 OCTETS) FOR R-S AND VD + R-S SCHEMES.

8. COMPATIBILITY WITH OTHER GROUND NETWORKS

This Standard is an enhanced subset of Reference [1].
Unlike Reference [1], this Standard:

- does not cover issues specifically related to the NASA Tracking and Data Relay Satellite System (TDRSS) operations and, in particular, the subject of convolutional coding with interleaving;
- permits the use of Reed-Solomon coding alone; i.e., without the convolutional (inner) code;
- permits interleaving depths of 1, 2, 3, 4, 5 and 8 (as opposed to 1 and 5 for Reference [1]);
- defines a symbol translation generation technique which ensures proper demodulator synchronisation when convolutional coding is not used.

PAGE INTENTIONALLY LEFT BLANK

APPENDIX A
GLOSSARY OF 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, $N > K$.

CHANNEL SYMBOL:
The unit of output of the innermost encoder.

CODEBLOCK:
A codeblock of an (N, K) block code is a sequence of N channel symbols that have been produced as a unit by encoding a sequence of K information symbols and that will be decoded as a unit.

CODE RATE:
The average ratio of the number of binary digits at the input of an encoder to the number of binary digits at its output.

CODEWORD:
In a block code, one of the sequences in the range of the one-to-one transformation (see **Block Encoding**).

CONCATENATION:
The use of two or more codes to process data sequentially, the output of one encoder being used as the input of the next.

CONNECTION VECTOR:
In convolutional coding, a device used to specify one of the parity checks to be performed on the shift register in the encoder. For a binary constraint length k convolutional code, a connection vector is a k-bit binary number. A 'one' in position i (counted from the left) indicates that the output of the i th stage of the shift register is to be used in computing that parity check.

CONSTRAINT LENGTH:
In convolutional coding, the number of consecutive input bits that are needed to determine the value of the output symbols at any time.

CONVOLUTIONAL CODE:
As used in this document, a code in which a number of output symbols are produced for each input information bit. Each output symbol is a linear combination of the current input bit as well as some or all of the previous $k-1$ bits where k is the constraint length of the code.

GF(256):

The Galois Field consisting of exactly '256' elements.

INNER CODE:

In a concatenated coding system, the last encoding algorithm that is applied to the data stream. The data stream here consists of the codewords generated by the outer decoder(s).

MODULATING WAVEFORM:

A way of representing data bits ('1' and '0') by a particular waveform.

NRZ-L:

A modulating waveform in which a data 'one' is represented by one of two levels, and a data 'zero' is represented by the other level.

NRZ-M:

A modulating waveform in which a data 'one' is represented by a change in level and a data 'zero' is represented by no change in level.

OUTER CODE:

In a concatenated coding system, the first encoding algorithm that is applied to the data stream.

REED-SOLOMON (R-S) SYMBOL:

A set of J bits that represents an element in GF (2^J), the code alphabet of a J-bit Reed-Solomon code.

SYSTEMATIC CODE:

A code in which the input information sequence appears in unaltered form as part of the output codeword.

TRANSPARENT CODE:

A code that has the property that complementing the input of the encoder or decoder results in complementing the output.

APPENDIX B
DUAL BASIS REPRESENTATION OF REED-SOLOMON SYMBOLS

The representation of Reed-Solomon symbols, as specified in Section 5 of this Standard, permits the implementation of an on-board Reed-Solomon encoder which, from a compatibility standpoint, is identical to Berlekamp's Reed-Solomon encoder as described in Reference [1] (CCSDS Recommendation, from which this Standard is directly derived). Berlekamp's R-S encoder's essential feature consists of a serial multiplier which, in some early on-board implementations, minimised the number of discrete components to be used.

This Berlekamp multiplier uses a dual basis representation to simplify the multiplication process, where the multiplication of two field elements x and y is given by:

(Dual x) $*$ y = Dual z

where (Dual x) is the dual basis representation of an element x.

In this approach, two different representations are used by the Berlekamp multiplier process, namely:

- **Dual Basis Representation**
  This representation is used for the symbols of the R-S codeword, i.e. the 8-bit symbols at the input and output of the encoder are represented in the dual basis of a ('a' is expressed as 'β' in Reference [1]).

- **Conventional Basis Representation**(4)
  This representation is used for the coefficients of the generator polynomial of the code, $g(z)$. In the case of the multiplier, the coefficients are represented in co-ordinates relative to the basis 1, a, a^2, . . ., a^7.

It should be noted that it is not necessary to use a Berlekamp serial multiplier in order to generate the R-S code specified in Section 5.2. An R-S encoder/decoder system may use any internal representation of the field, as long as the encoder output and the decoder input symbols are consistent with the dual basis representation specified in Section 5.2.

NOTE (4): In Reference [1], pages 4-6, the conventional basis representation is that specified in co-ordinates relative to the basis (1, α, . . ., α^7). However, Reference [1] erroneously states that this basis (1, α, . . ., α^7) is dual to the (l_0, l_1, . . ., l_7) basis.

The definition and relationships between the different representations are as follows:

(a) The relationship between the basis b, defined in Section 5.2, and basis a is expressed by:

$$a = b^{57}$$

(b) The relationship between these two representations and the conventional basis representation specified in Reference [1] (in which the representation of symbols associated with the conventional encoder is given in terms of polynomials in 'α') is expressed by:

$$b = \alpha^{11}$$

$$a = b^{57} = \alpha^{(11 * 57) \text{ modulo } 255}$$

$$a = \alpha^{117} \text{ (= } \beta \text{, in Reference [1])}$$

where α is a root of the primitive polynomial:

$$x^8 + x^7 + x^2 + x + 1$$

(c) The dual basis representation of a finite field element, z, is given by:

$$z = z_0 l_0 + z_1 l_1 + \; .\;.\;.\;. + z_7 l_7$$

where z_i is either a zero or a one and l_i is given by:

$$l_7 = b^{22}$$

$$l_6 = b^{122}$$

$$l_5 = b^{179}$$

$$l_4 = b^{236}$$

$$l_3 = b^{038}$$

$$l_2 = b^{206}$$

$$l_1 = b^{008}$$

$$l_0 = b^{220}$$

This dual basis representation has the property that:

$$\mathrm{Tr}\,(l_i\, a^j) = \begin{cases} 1, \text{ if } i = j \\ 0, \text{ otherwise} \end{cases}$$

for each $j = 0, 1, \ldots, 7$. The function Tr (z), called the 'trace', is defined by:

$$\mathrm{Tr}\,(z) = \sum_{K=0}^{7} z^{2^K}$$

for each element of GF(256). The eight-bit string representing z is transmitted with z_0 first, followed by $z_1, z_2, \ldots, z_7$.

(d) For the basis α, any element of GF(256) has a representation of the form:

$$u_7\,\alpha^7 + u_6\,\alpha^6 + \ldots + u_0\,\alpha^0$$

The relationship between the representation in basis α and the representation in dual basis (as defined in the preceding paragraph) can be found in Table B.1 which, in addition, gives the relationship between other representations for the 8-bit symbols of GF (256).

POWER OF α	POWER OF b	CONVENTIONAL REPRESENTATION USING POWERS OF *b*	CONVENTIONAL REPRESENTATION USING POWERS OF *a*	DUAL BASIS REPRESENTATION	CONVENTIONAL REPRESENTATION USING POWERS OF α
000	000	00000001	00000001	01111011	00000001
011	001	00000010	10101110	01111001	10101101
022	002	00000100	00001010	00101011	10111110
033	003	00001000	10011101	00111111	00111010
044	004	00010000	01000100	00001001	00111100
055	005	00100000	11101000	10000111	11011100
066	006	01000000	00110101	01011111	01010110
077	007	10000000	10111100	00110111	11001010
000	000	00000001	00000001	01111011	00000001
117	057	00110010	00000010	11110111	01001101
234	114	01111000	00000100	11101110	10001100
096	171	00110110	00001000	11011100	11110011
213	228	10110000	00010000	10111001	00101010
075	030	10001001	00100000	01110011	11110001
192	087	01101000	01000000	11100111	10110000
054	144	11110111	10000000	11001111	01101110
242	022	10011000	01100101	00000001	11001100
067	122	00000011	10101111	00000010	10101100
184	179	01010110	11010101	00000100	01111001
046	236	10001000	00100001	00001000	11110000
163	038	01011010	01000010	00010000	11111101
226	206	00010111	11100001	00100000	00101110
088	008	01011111	01001001	01000000	01000010
125	220	11101101	11110111	10000000	11000101
000	000	00000001	00000001	01111011	00000001
001	116	10111111	00101000	10101111	00000010
002	232	10100111	11110001	10011001	00000100
003	093	01110100	10010011	11111010	00001000
004	209	10111000	10001101	10000110	00010000
005	070	01111011	10101011	11101100	00100000
006	186	11100000	01100001	11101111	01000000
007	047	10101011	01100110	10001101	10000000

Table B.1 RELATIONSHIP BETWEEN DIFFERENT REPRESENTATIONS OF GF (256)