INTERNATIONAL TELECOMMUNICATION UNION

RADIOCOMMUNICATION STUDY GROUPS

Document 9/BL/8-E
11 October 2004

Source: Document 9/8

Radiocommunication Study Group 9

DRAFT MODIFICATION OF RECOMMENDATION ITU-R F.763-4

Data transmission over HF circuits using phase shift keying or quadrature amplitude modulation

(Question ITU-R 145/9)

(1992-1994-1995-1997-1999)

1 Summary

This Recommendation provides data transmission systems using PSK and quadrature amplitude modulation over HF channels. Information is contained in Annex 6 for data rates from 3 200 to 12 800 bit/s.

2 Draft modification

Add new *recommends* as follows:

6 that for data transmissions at binary rates from 3 200 to 12 800 bit/s using serial transmission modems, the preferred system characteristics are described in Annex 6.

Add new Annex as contained in the Attachment.

Attachment: 1

ATTACHMENT

Annex 6

High data rate waveforms 3 200/4 800/6 400/8 000/9 600/12 800 bit/s using a serial transmission modem over HF circuits

1 Introduction

This Annex provides a detailed description of modem waveforms to ensure operation within HF radio networks. This family of waveforms is also known as STANAG 4539. A family of self-identifying waveforms is described for coded operation from 3 200 bit/s to 9 600 bit/s (with optional uncoded operation at 12 800 bit/s). The self-identifying feature[1] of this family of waveforms enables rapid adaptation of the modulation to respond to changing channel conditions. The key features of this waveform are:

a) Ability to track an HF channel with 3-5 ms of multipath fading.

b) Ability to correct for errors caused by fading, multipath and noise.

c) The equipment passband bandwidth requirement is 300 to 3 050 Hz.

d) Automatic data rate and interleaver detection.

e) Able to tolerate a shift of ±75 Hz between the transmission and reception HF carriers.

1.1 Overview

This section presents a modem waveform and coding for data rates of 3 200, 4 800, 6 400, 8 000, 9 600 and uncoded optional operation at 12 800 bit/s.

A block interleaver is used to obtain 6 interleaving lengths ranging from 0.12 s to 8.64 s. A single coding option, a constraint length 7, rate 1/2 convolutional code, punctured to rate 3/4, is used for all data rates. The full-tail-biting approach is used to produce block codes from this convolutional code that are the same length as the interleaver.

Both the data rate and interleaver settings are explicitly transmitted as a part of the waveform, both as part of the initial preamble and then periodically as both a reinserted preamble and in the periodic known symbol blocks. This self-identifying feature is important in developing efficient (ARQ) protocols for high frequency (HF) channels. The receive modem is able to deduce the data rate and interleaver setting either from the preamble or from the subsequent data portion of the waveform.

1.2 Modulation

The symbol rate for all symbols is 2 400 symbols-per-second, which should be accurate to a minimum of ±0.024 symbols-per-second (10 ppm) when the transmit data clock is generated by the modem and not provided by the data terminal equipment (DTE). Phase-shift keying (PSK) and quadrature amplitude modulation (QAM) modulation techniques are used. The sub-carrier (or pair of quadrature sub-carriers in the case of QAM) is centred at 1 800 Hz accurate to 0.018 Hz (10 ppm). The phase of the Quadrature sub-carrier relative to the In-phase carrier is 90 degrees. The power spectral density of the modulator output signal is constrained to be at least 20 dB below the

---

[1] Symbols sent in the preamble and channel probe phases specify data rate and interleaver depth.

signal level measured at 1 800 Hz, when tested outside of the band from 200 Hz to 3 400 Hz. The filter employed should introduce a ripple of no more than ±2 dB in the range from 800 Hz to 2 800 Hz. The filter used is a square root Nyquist filter with alpha = 0.35.

1.2.1 Known symbols

For all known symbols, the modulation used is PSK, with the symbol mapping shown in Table 1 and Fig. 1. No scrambling is applied to the known symbols.

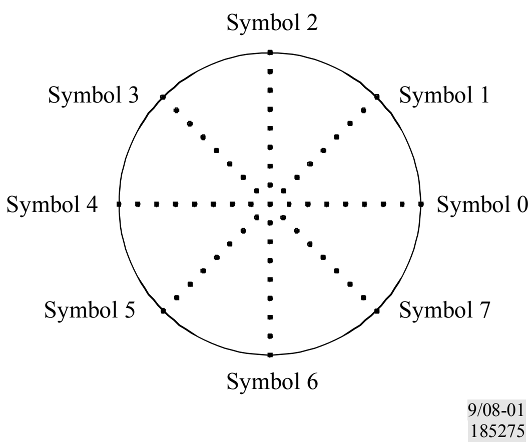
TABLE 1

8-PSK symbol mapping

Symbol Number	Phase	In-Phase	Quadrature
0	0	1.000000	0.000000
1	π/4	0.707107	0.707107
2	π/2	0.000000	1.000000
3	3π/4	–0.707107	0.707107
4	π	–1.000000	0.000000
5	5π/4	–0.707107	–0.707107
6	3π/2	0.0000000	–1.000000
7	7π/4	0.707107	–0.707107

FIGURE 1

8-PSK signal constellation and symbol mapping



1.2.2 Data symbols

For data symbols, the modulation used will depend upon the data rate. Table 2 describes the modulation that is used with each data rate.

TABLE 2

Modulation used to obtain each data rate

Data Rate (bit/s)	Modulation
3 200	QPSK
4 800	8-PSK
6 400	16-QAM
8 000	32-QAM
9 600	64-QAM
12 800	64-QAM

Both the 16-QAM and 32-QAM constellations employ multiple PSK rings to maintain good peak-to-average ratios, and the 64-QAM constellation is a variation of the standard square QAM constellation, which has been modified to improve the peak-to-average ratio.

1.2.2.1 PSK data symbols

For the PSK constellations, a distinction is made between the data bits and the symbol number for the purposes of scrambling the QPSK modulation to appear as 8-PSK, on-air. Scrambling is applied as a modulo 8 addition of a scrambling sequence to the 8-PSK symbol number. Transcoding is an operation which links a symbol to be transmitted to a group of data bits.

1.2.2.1.1 QPSK symbol mapping

For the 3 200 bit/s user data rate, transcoding is achieved by linking one of the symbols specified in Table 1 to a set of two consecutive data bits (dibit) as shown in Table 3. In this Table, the leftmost bit of the dibit is the older bit; i.e. fetched from the interleaver before the rightmost bit.

TABLE 3

Transcoding for 3 200 bit/s

Dibit	Symbol
00	0
01	2
11	4
10	6

1.2.2.1.2 8-PSK symbol mapping

For the 4 800 bit/s user data rate, transcoding is achieved by linking one symbol to a set of three consecutive data bits (tribit) as shown in Table 4. In this Table, the leftmost bit of the tribit is the oldest bit; i.e. fetched from the interleaver before the other two, and the rightmost bit is the most recent bit.

TABLE 4

Transcoding for 4 800 bit/s

Tribit	Symbol
000	1
001	0
010	2
011	3
100	6
101	7
110	5
111	4

1.2.2.1.3 QAM data symbols

For the QAM constellations, no distinction is made between the number formed directly from the data bits and the symbol number. Each set of 4 bits (16-QAM), 5 bits (32-QAM) or 6 bits (64-QAM) is mapped directly to a QAM symbol. For example, the four bit grouping 0111 would map to symbol 7 in the 16-QAM constellation while the 6 bit grouping 100011 would map to symbol 35 in the 64-QAM constellation. Again, in each case the leftmost bit is the oldest bit, i.e. fetched from the interleaver before the other bits, and the rightmost bit is the most recent bit.

The mapping of bits to symbols for the QAM constellations has been selected to minimize the number of bit errors incurred when errors involve adjacent signalling points in the constellation.

1.2.2.1.4 The 16-QAM constellation

The constellation points, which are for 16-QAM, are shown in Fig. 2 and described in terms of their In-phase and Quadrature components in Table 5. As can be seen in the Figure, the 16-QAM constellation comprises two PSK rings: 4-PSK inner and 12-PSK outer symbols.

FIGURE 2

16-QAM signalling constellation

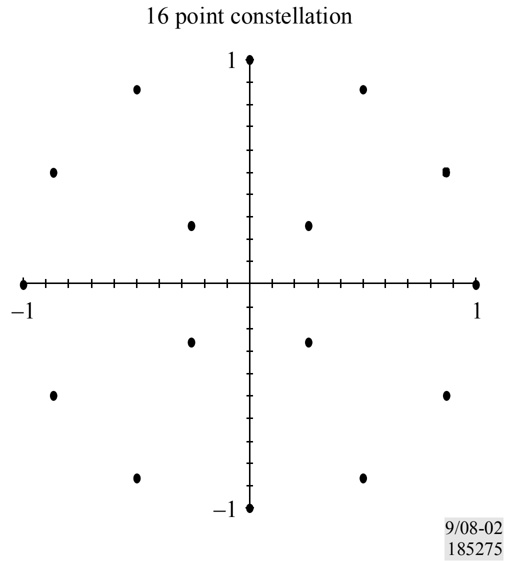


TABLE 5

In-phase and quadrature components of each 16-QAM symbol

Symbol Number	In-Phase	Quadrature
0	0.866025	0.500000
1	0.500000	0.866025
2	1.000000	0.000000
3	0.258819	0.258819
4	–0.500000	0.866025
5	0.000000	1.000000
6	–0.866025	0.500000
7	–0.258819	0.258819
8	0.500000	–0.866025
9	0.000000	–1.000000
10	0.866025	–0.500000
11	0.258819	–0.258819
12	–0.866025	–0.500000
13	–0.500000	–0.866025
14	–1.000000	0.000000
15	–0.258819	–0.258819

1.2.2.1.5 The 32-QAM constellation

The constellation points, which are used for 32-QAM, are shown in Fig. 3 and specified in terms of their In-phase and Quadrature components in Table 6. This constellation contains an outer ring of 16 symbols and an inner square of 16 symbols.

FIGURE 3

32-QAM signalling constellation

32 point constellation

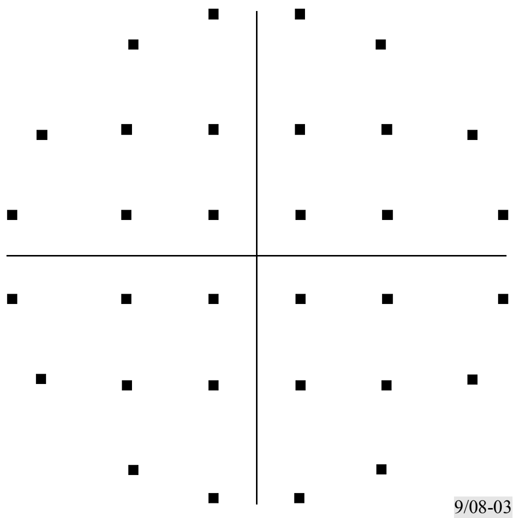


TABLE 6

In-phase and Quadrature components of each 32-QAM symbol

Symbol Number	In-Phase	Quadrature	Symbol Number	In-Phase	Quadrature
0	0.866380	0.499386	16	0.866380	–0.499386
1	0.984849	0.173415	17	0.984849	–0.173415
2	0.499386	0.866380	18	0.499386	–0.866380
3	0.173415	0.984849	19	0.173415	–0.984849
4	0.520246	0.520246	20	0.520246	–0.520246
5	0.520246	0.173415	21	0.520246	–0.173415
6	0.173415	0.520246	22	0.173415	–0.520246
7	0.173415	0.173415	23	0.173415	–0.173415
8	–0.866380	0.499386	24	–0.866380	–0.499386
9	–0.984849	0.173415	25	–0.984849	–0.173415
10	–0.499386	0.866380	26	–0.499386	–0.866380
11	–0.173415	0.984849	27	–0.173415	–0.984849
12	–0.520246	0.520246	28	–0.520246	–0.520246
13	–0.520246	0.173415	29	–0.520246	–0.173415
14	–0.173415	0.520246	30	–0.173415	–0.520246
15	–0.173415	0.173415	31	–0.173415	–0.173415

1.2.2.1.6 The 64-QAM constellation

The constellation points which are used for the 64-QAM modulation are shown in Fig. 4 and described in terms of their In-phase and Quadrature components in Table 7. This constellation is a variation on the standard 8 × 8 square constellation, which achieves a better peak-to-average ratio without sacrificing the very good pseudo-Gray code properties of the square constellation.

FIGURE 4

64-QAM signalling constellation

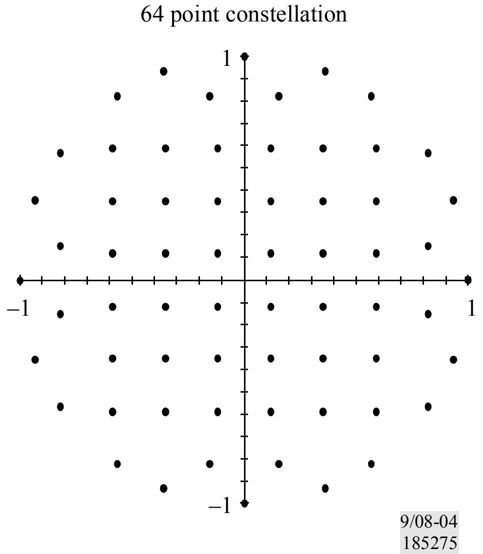


TABLE 7

In-phase and quadrature components of each 64-QAM symbol

Symbol Number	In-Phase	Quadrature	Symbol Number	In-Phase	Quadrature
0	1.000000	0.000000	32	0.000000	1.000000
1	0.822878	0.568218	33	–0.822878	0.568218
2	0.821137	0.152996	34	–0.821137	0.152996
3	0.932897	0.360142	35	–0.932897	0.360142
4	0.000000	–1.000000	36	–1.000000	0.000000
5	0.822878	–0.568218	37	–0.822878	–0.568218
6	0.821137	–0.152996	38	–0.821137	–0.152996
7	0.932897	–0.360142	39	–0.932897	–0.360142
8	0.568218	0.822878	40	–0.568218	0.822878

TABLE 7 (*continued*)

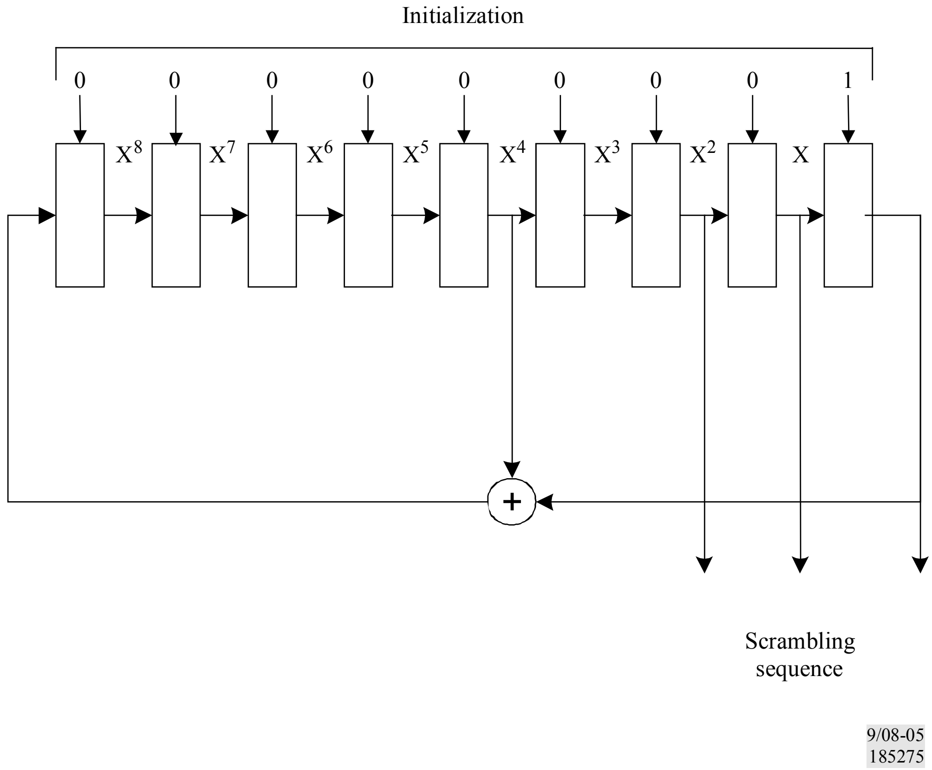
Symbol Number	**In-Phase**	**Quadrature**	**Symbol Number**	**In-Phase**	**Quadrature**
9	0.588429	0.588429	41	–0.588429	0.588429
10	0.588429	0.117686	42	–0.588429	0.117686
11	0.588429	0.353057	43	–0.588429	0.353057
12	0.568218	–0.822878	44	–0.568218	–0.822878
13	0.588429	–0.588429	45	–0.588429	–0.588429
14	0.588429	–0.117686	46	–0.588429	–0.117686
15	0.588429	–0.353057	47	–0.588429	–0.353057
16	0.152996	0.821137	48	–0.152996	0.821137
17	0.117686	0.588429	49	–0.117686	0.588429
18	0.117686	0.117686	50	–0.117686	0.117686
19	0.117686	0.353057	51	–0.117686	0.353057
20	0.152996	–0.821137	52	–0.152996	–0.821137
21	0.117686	–0.588429	53	–0.117686	–0.588429
22	0.117686	–0.117686	54	–0.117686	–0.117686
23	0.117686	–0.353057	55	–0.117686	–0.353057
24	0.360142	0.932897	56	–0.360142	0.932897
25	0.353057	0.588429	57	–0.353057	0.588429
26	0.353057	0.117686	58	–0.353057	0.117686
27	0.353057	0.353057	59	–0.353057	0.353057
28	0.360142	–0.932897	60	–0.360142	–0.932897
29	0.353057	–0.588429	61	–0.353057	–0.588429
30	0.353057	–0.117686	62	–0.353057	–0.117686
31	0.353057	–0.353057	63	–0.353057	–0.353057

1.2.3 Data scrambling

Data symbols for the 8-PSK symbol constellation (3 200 bit/s, 4 800 bit/s) are scrambled by modulo 8 addition with a scrambling sequence. The data symbols for the 16-QAM, 32-QAM, and 64-QAM constellations are scrambled by using an exclusive or (XOR) operation. Sequentially, the data bits forming each symbol (4 for 16-QAM, 5 for 32-QAM, and 6 for 64-QAM) are XOR'd with an equal number of bits from the scrambling sequence. In all cases, the scrambling sequence generator polynomial is $\times^9 + \times^4 + 1$ and the generator is initialized to 1 at the start of each data frame. A block diagram of the scrambling sequence generator is shown in Fig. 5.

FIGURE 5

Scrambling sequence generator illustrating scrambling generator for 8-PSK symbols



For 8-PSK symbols (3 200 bit/s and 4 800 bit/s), the scrambling is carried out taking the modulo 8 sum of the numerical value of the binary triplet consisting of the last (rightmost) three bits in the shift register, and the symbol number (transcoded value). For example, if the last three bits in the scrambling sequence shift register were 010 which has a numerical value equal 2, and the symbol number before scrambling was 6, symbol 0 would be transmitted since: (6 + 2) Modulo 8 = 0. For 16-QAM symbols, scrambling is carried out by XORing the 4 bit number consisting of the last (rightmost) four bits in the shift register with the symbol number. For example, if the last 4 bits in the scrambling sequence shift register were 0101 and the 16-QAM symbol number before scrambling was 3 (i.e. 0011), symbol 6 (0110) would be transmitted. For 32-QAM symbols, scrambling is carried out by XORing the 5 bit number formed by the last (rightmost) five bits in the shift register with the symbol number. For 64-QAM symbols, scrambling is carried out by XORing the 6 bit number formed by the last (rightmost) six bits in the shift register with the symbol number.

After each data symbol is scrambled, the generator is iterated (shifted) the required number of times to produce all new bits for use in scrambling the next symbol (i.e. 3 iterations for 8-PSK, 4 iterations for 16-QAM, 5 iterations for 32-QAM and 6 iterations for 64-QAM). Since the generator is iterated after the bits are used, the first data symbol of every data frame should be scrambled by the appropriate number of bits from the initialization value of 00000001.

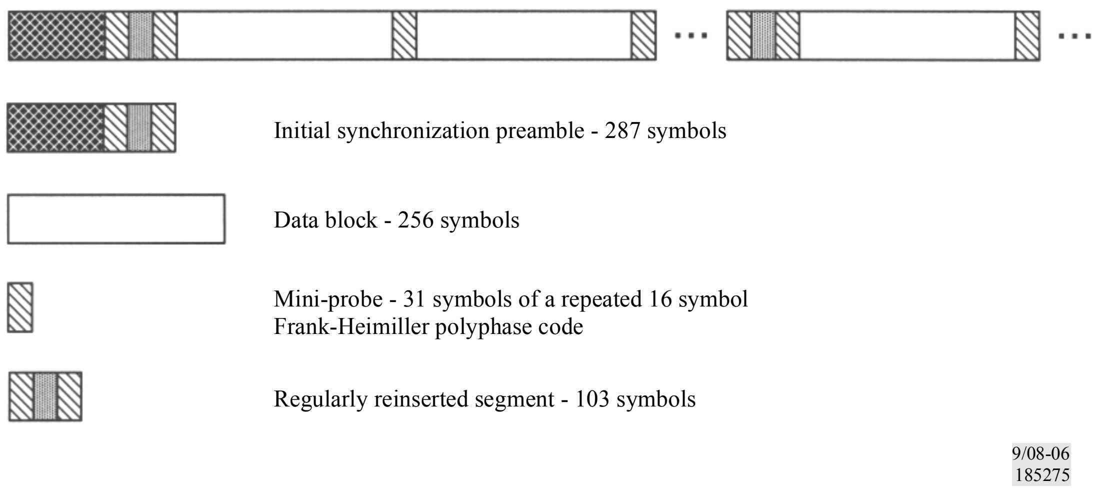
The length of the scrambling sequence is 511 bits. For a 256 symbol data block with 6 bits per symbol, this means that the scrambling sequence will be repeated just slightly more than three times, although in terms of symbols, there will be no repetition.

1.3 Frame structure

The frame structure that is used for the waveforms of this Annex is shown in Fig. 6. An initial 287 symbol preamble is followed by 72 frames of alternating data and known symbols. Each data frame has a data block consisting of 256 data symbols, followed by a mini-probe of 31 symbols of known data. After 72 data frames, a 72 symbol subset of the initial preamble is reinserted to facilitate late acquisition, Doppler shift removal, and synchronization adjustment. The total length of known data in this segment is actually 103 symbols: the 72 reinserted preamble symbols plus the preceding 31 symbol mini-probe segment which follows the last 256 symbol data block.

FIGURE 6

Frame structure for all waveforms



1.3.1 Synchronization and reinserted preambles

The synchronization preamble is used for rapid initial synchronization. The reinserted preamble is used to facilitate acquisition of an ongoing transmission (acquisition on data).

1.3.1.1 Synchronization preamble

The synchronization preamble consists of two parts. The first part consists of at least N blocks of 184 8-PSK symbols to be used exclusively for radio and modem AGC. The value of N is configurable to range from values of 0 to 7 (for N = 0 this first section is not sent at all). These 184 symbols are formed by taking the complex conjugate of the first 184 symbols of the sequence specified below for the second section.

The second section consists of 287 symbols. The first 184 symbols are intended exclusively for synchronization and Doppler offset removal purposes while the final 103 symbols, which are common with the reinserted preamble, also carry information regarding the data rate and interleaver settings. Expressed as a sequence of 8-PSK symbols, using the symbol numbers given in Table 1 the second section of the synchronization preamble is as follows:

1, 5, 1, 3, 6, 1, 3, 1, 1, 6, 3, 7, 7, 3, 5, 4, 3, 6, 6, 4, 5, 4, 0,

2, 2, 2, 6, 0, 7, 5, 7, 4, 0, 7, 5, 7, 1, 6, 1, 0, 5, 2, 2, 6, 2, 3,

6, 0, 0, 5, 1, 4, 2, 2, 2, 3, 4, 0, 6, 2, 7, 4, 3, 3, 7, 2, 0, 2, 6,

4, 4, 1, 7, 6, 2, 0, 6, 2, 3, 6, 7, 4, 3, 6, 1, 3, 7, 4, 6, 5, 7, 2,

0, 1, 1, 1, 4, 4, 0, 0, 5, 7, 7, 4, 7, 3, 5, 4, 1, 6, 5, 6, 6, 4, 6,

3, 4, 3, 0, 7, 1, 3, 4, 7, 0, 1, 4, 3, 3, 3, 5, 1, 1, 1, 4, 6, 1, 0,

6, 0, 1, 3, 1, 4, 1, 7, 7, 6, 3, 0, 0, 7, 2, 7, 2, 0, 2, 6, 1, 1, 1,

2, 7, 7, 5, 3, 3, 6, 0, 5, 3, 3, 1, 0, 7, 1, 1, 0, 3, 0, 4, 0, 7, 3,

0, 0, 0, 0, 0, 2, 4, 6, 0, 4, 0, 4, 0, 6, 4, 2, 0, 0, 0, 0, 0, 2, 4, 6, 0, 4, 0, 4, 0, 6, 4,

2,

(D0, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0 + 0, 4, 0, 4, 0, 0, 4, 4, 0, 0, 0, 0, 0) Modulo 8

(D1, D1, D1, D1, D1, D1, D1, D1, D1, D1, D1, D1, D1 + 0, 4, 0, 4, 0, 0, 4, 4, 0, 0, 0, 0, 0) Modulo 8

(D2, D2, D2, D2, D2, D2, D2, D2, D2, D2, D2, D2, D2 + 0, 4, 0, 4, 0, 0, 4, 4, 0, 0, 0, 0, 0) Modulo 8

6,

4, 4, 4, 4, 4, 6, 0, 2, 4, 0, 4, 0, 4, 2, 0, 6, 4, 4, 4, 4, 4, 6, 0, 2, 4, 0, 4, 0, 4, 2, 0.

where the data symbols D_0, D_1 and D_2 take one of 30 sets of values chosen from Table 8 to indicate the data rate and interleaver settings. The Modulo operations are meant to signify that each of the D values are used to shift the phase of a length 13 bit Barker code (0101001100000) by performing modulo 8 addition of the D value with each of the Barker code 13 phase values (0 or 4). This operation can encode 6 bits of information using QPSK modulation of the 13 bit (chip) Barker codes. Since the three Barker code sequences only occupy 39 symbols, the 31 symbol mini-probes are lengthened to 32 symbols each to provide the additional two symbols required to pad the three 13 symbol Barker codes up to a total of 41 symbols.

TABLE 8

D0, D1, D2 8-PSK symbol values as a function of data rate and interleaver length

Data Rate (bit/s)	Interleaver length in frames (256 symbol data blocks)					
	1	3	9	18	36	72
3 200	0,0,4	0,2,6	0,2,4	2,0,6	2,0,4	2,2,6
4 800	0,6,2	0,4,0	0,4,2	2,6,0	2,6,2	2,4,0
6 400	0,6,4	0,4,6	0,4,4	2,6,6	2,6,4	2,4,6
8 000	6,0,2	6,2,0	6,2,2	4,0,0	4,0,2	4,2,0
9 600	6,0,4	6,2,6	6,2,4	4,0,6	4,0,4	4,2,6
12 800	6,6,2*	N/A	N/A	N/A	N/A	N/A

* For 12 800 bit/s 1 frame interleaver interpreted as no interleaving.

The mapping chosen to create Table 8 uses 3 bits each to specify the data rate and interleaver length. The 3 data rate bits are the three most significant bits (MSB) of 3 dibit symbols and the interleaver length bits are the least significant bits (LSB). The phase of the Barker code is determined from the 3 resulting dibit words using Table 3, the dibit transcoding table. The 3 bit data rate and interleaver length mappings are shown in Table 9. Note that the transcoding has the effect of placing the 3 interleaver length bits in quadrature with the 3 data rate bits.

TABLE 9

Bit patterns for specifying data rate and interleaver length

Data Rate	3 Bit mapping
3 200	001
4 800	010
6 400	011
8 000	100
9 600	101
12 800	110

Interleaver length	3 Bit mapping	Name
1 Frame	001	Ultra Short (US)
3 Frames	010	Very Short (VS)
9 Frames	011	Short (S)
18 Frames	100	Medium (M)
36 Frames	101	Long (L)
72 Frames	110	Very Long (VL)

Since the Barker code is unbalanced in terms of the number of 0s and 1s, these 3-bit patterns have been chosen to avoid the 000 or 111 patterns in order to minimize the unbalance in the combined three symbols. More specifically, one of the three repeats of the Barker code that appears on each of the quadrature components is always phase shifted by 180 degrees with respect to the other two. This results in a net imbalance in each quadrature component of the 39 symbols that is always 17 to 22, rather than ever being 12 to 27.

1.3.1.2 Reinserted preamble

The reinserted preamble is identical to the final 72 symbols of the synchronization preamble. In fact, the final 103 symbols are common between the synchronization preamble and the contiguous block consisting of the reinserted preamble and the mini-probe which immediately precedes it. The 103 symbols of known data (including the 31 mini-probe symbols of the preceding data frame) are thus:

0, 0, 0, 0, 0, 2, 4, 6, 0, 4, 0, 4, 0, 6, 4, 2, 0, 0, 0, 0, 0, 2, 4, 6, 0, 4, 0, 4, 0, 6, 4, 2,
(D0, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0, D0 + 0, 4, 0, 4, 0, 0, 4, 4, 0, 0, 0, 0, 0) Modulo 8
(D1, D1, D1, D1, D1, D1, D1, D1, D1, D1, D1, D1, D1 + 0, 4, 0, 4, 0, 0, 4, 4, 0, 0, 0, 0, 0) Modulo 8
(D2, D2, D2, D2, D2, D2, D2, D2, D2, D2, D2, D2, D2 + 0, 4, 0, 4, 0, 0, 4, 4, 0, 0, 0, 0, 0) Modulo 8
6, 4, 4, 4, 4, 4, 6, 0, 2, 4, 0, 4, 0, 4, 2, 0, 6, 4, 4, 4, 4, 4, 6, 0, 2, 4, 0, 4, 0, 4, 2, 0.

where the data symbols D_0, D_1, and D_2 again take one of 30 sets of values chosen from Table 8 to indicate the data rate and interleaver settings as described in the Synchronization preamble section above. Note that the first 31 of these symbols are the immediately preceding mini-probe, which follows the last of the 72 data blocks.

1.3.2 Mini-probes

Mini-probes 31 symbols in length are inserted following every 256 symbol data block and at the end of each preamble (where they are considered to be part of the preamble). Using the 8-PSK symbol mapping, each mini-probe is based on the repeated Frank-Heimiller sequence. The sequence that is used, specified in terms of the 8-PSK symbol numbers, is given by:

0, 0, 0, 0, 0, 2, 4, 6, 0, 4, 0, 4, 0, 6, 4, 2, 0, 0, 0, 0, 0, 2, 4, 6, 0, 4, 0, 4, 0, 6, 4.

This mini-probe will be designated "+". The phase inverted version of this is:

4, 4, 4, 4, 4, 6, 0, 2, 4, 0, 4, 0, 4, 2, 0, 6, 4, 4, 4, 4, 4, 6, 0, 2, 4, 0, 4, 0, 4, 2, 0.

and mini-probes using this sequence will be designated "–", as the phase of each symbol has been rotated 180 degrees from the "+".

There are a total of 73 mini-probes for each set of 72 data blocks. For convenience, each mini-probe is sequentially numbered, with mini-probe 0 being defined as the last 31 symbols of the preceding (reinserted) preamble, mini-probe number 1 following the first data block after a (reinserted) preamble. Mini-probe 72 follows the 72nd data block, and is also the first 31 symbols of the next 103 symbol reinserted preamble. Mini-probes 0 and 72 have been defined as part of the reinsertion preamble to have the signs – and + respectively. The data rate and interleaver length information encoded into the synchronization and reinserted preambles are also be encoded into mini-probes

1 to 72. These 72 mini-probes are grouped into four sets of 18 consecutive mini-probes (1 to 18, 19 to 36, 37 to 54, and 55 to 72). Note that the 256 symbol data block that immediately follows the 18th mini-probe, in each of the first three sets, is also the 1st data block of an interleaver block with frame lengths of 1, 3, 9, and 18. The length 36 interleaver block begins after the second set, and a reinserted preamble begins after the fourth set. This structure permits data to begin to be demodulated as soon as the interleaver boundary becomes known.

Each 18 mini-probe sequence consists of seven – signs, a + sign, followed by six sign values that are dependent on the data rate and interleaver length, three sign values that specify which of the four sets of 18 mini-probes it is, and then finally a + sign. For the fourth set, this final + sign (mini-probe 72) is also the initial mini-probe of the next reinserted preamble (which uses the + phase).

Pictorially, this length 18 sequence is: – – – – – – – + S_0 S_1 S_2 S_3 S_4 S_5 S_6 S_7 S_8 +, where the first six S_i sign values are defined in Table 10. Note that these 6 bit patterns (+ is a 0) correspond to the concatenation of the 3 bit mappings from Table 9 for the data rate (S_0 S_1 S_2) and the interleaver length (S_3 S_4 S_5). The final three S_i sign values which specify the mini-probe set (count) are defined in Table 11.

TABLE 10

S0, S1, S2, S3, S4, S5 (sign) values as a function of data rate and interleaver setting

Data Rate (bit/s)	Interleaver length in frames (256 symbol data blocks)					
	1	3	9	18	36	72
3 200	+ + – + + –	+ + – + – +	+ + – + – –	+ + – – + +	+ + – – + –	+ + – – – +
4 800	+ – + + + –	+ – + + – +	+ – + + – –	+ – + – + +	+ – + – + –	+ – + – – +
6 400	+ – – + + –	+ – – + – +	+ – – + – –	+ – – – + +	+ – – – + –	+ – – – – +
8 000	– + + + + –	– + + + – +	– + + + – –	– + + – + +	– + + – + –	– + + – – +
9 600	– + – + + –	– + – + – +	– + – + – –	– + – – + +	– + – – + –	– + – – – +
12 800	– – + + + –	N/A	N/A	N/A	N/A	N/A

TABLE 11

S6, S7, S8 (sign) values as a function of mini-probe set

Mini-probe set			
1 to 18	19 to 36	37 to 54	55 to 72
+ + –	+ – +	+ – –	– + +

The first eight mini-probes in each set (– – – – – – – +) uniquely locate the starting point for the following nine Si values. This is possible since the Si sequences used contain at most runs of four + or – phases. This makes it impossible for a sequence of seven mini-probes with the same phase followed by one with a phase reversal to occur anywhere else except at the beginning of one of the 18 mini-probe sequences. Once this fixed eight mini-probe pattern is located, the 0 or 180 degree phase ambiguity is also resolved so that the following 9 mini-probes can be properly matched to the data rate, interleaver length, and mini-probe set count. The entire mini-probe sequence is as follows:

[rp] – – – – – – – + S0 S1 S2 S3 S4 S5 S6 S7 S8 + – – – – – – + S0 S1 S2 S3 S4 S5 S6 S7 S8 + – – – – – – – + S0 S1 S2 S3 S4 S5 S6 S7 S8 + – – – – – – – + S0 S1 S2 S3 S4 S5 S6 S7 S8 [rp]

where the [rp] represents the 103 reinserted preamble symbols (includes mini-probes 72 and 0).

1.4 Coding and interleaving

The interleaver is a block interleaver. Each block of input data is also encoded using a block encoding technique with a code block size equal to the size of the block interleaver. Thus, the input data bits will be sent as successive blocks of bits that span the duration of the interleaver length selected. Table 12 shows the number of input data bits per block as function of both data rate and interleaver length. Note that an "input data block" should not be confused with the 256 symbol data block that is part of a data frame in the waveform format. The bits from an input data block will be mapped through the coding and interleaving to the number of data frames, and thus 256 symbol data blocks, that define the interleaver length.

TABLE 12

Input data block size in bits as a function of data rate and interleaver length

Data Rate (bit/s)	**Interleaver length in frames**					
	1	**3**	**9**	**18**	**36**	**72**
	Number of input data bits per block					
3 200	384	1 152	3 456	6 912	13 824	27 648
4 800	576	1 728	5 184	10 368	20 736	41 472
6 400	768	2 304	6 912	13 824	27 648	55 296
8 000	960	2 880	8 640	17 280	34 560	69 120
9 600	1 152	3 456	10 368	20 736	41 472	82 944

1.4.1 Block boundary alignment

Each code block is interleaved within a single interleaver block of the same size. The boundaries of these blocks is aligned such that the beginning of the first data frame following each reinserted preamble should coincide with an interleaver boundary. Thus for an interleaver length of three frames, the first three data frames following a reinserted preamble will contain all of the encoded bits for a single input data block. The first data symbol from the first data frame in each interleaver set will have as its MSB the first bit fetched from the interleaver.

1.4.2 Block encoding

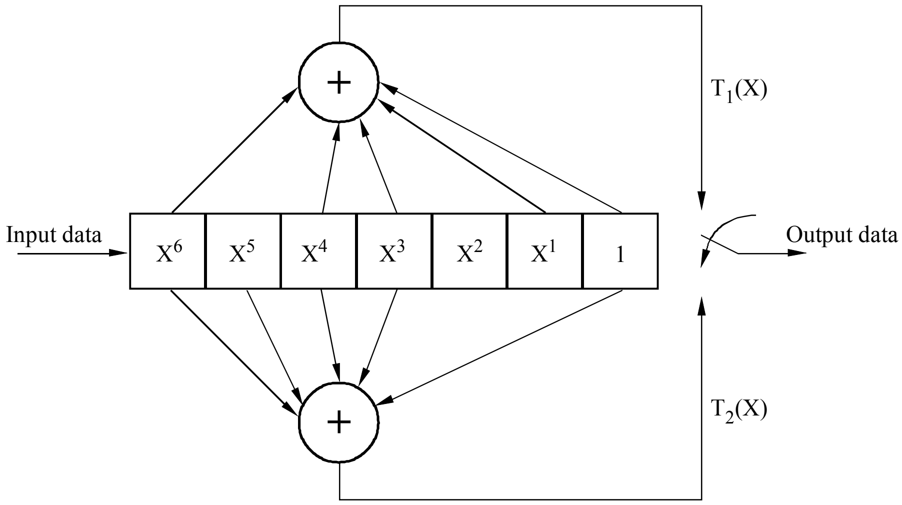
The full-tail-biting and puncturing techniques is used with a rate 1/2 convolutional code to produce a rate 3/4 block code that is the same length as the interleaver.

1.4.3 Rate 1/2 convolutional code

A constraint length 7, rate 1/2 convolutional code is used prior to puncturing. Figure 7 is a pictorial representation of the encoder. The two generator polynomials used are:

FIGURE 7

Constraint length 7, rate 1/2 convolutional encoder



$T_1 = X^6 + X^4 + X^3 + X + 1$

$T_2 = X^6 + X^5 + X^4 + X^3 + 1$

9/08-07
185275

The two summing nodes in the Figure represent modulo 2 addition. For each bit input to the encoder, two bits are taken from the encoder, with the upper output bit, T_1 (x), taken first.

1.4.3.1 Full-tail-biting encoding

To begin encoding each block of input data, the encoder is preloaded by shifting in the first six input data bits without taking any output bits. These six input bits is temporarily saved so that they can be used to "flush" the encoder. The first two coded output bits are taken after the seventh bit has been shifted in, and are defined to be the first two bits of the resulting block code. After the last input data bit has been encoded, the first six "saved" data bits are encoded. Note that the encoder shift register should not be changed before encoding these saved bits; i.e. it should be filled with the last seven input data bits. The six "saved" data bits are encoded by shifting them into the encoder one at a time, beginning with the earliest of the six. The encoding thus continues by taking the two resulting coded output bits as each of the saved six bits is shifted in. These encoded bits are the final bits of the resulting (unpunctured) block code. Prior to puncturing, the resulting block code will have exactly twice as many bits as the input information bits. Puncturing of the rate 1/2 code to the required rate 3/4 is done prior to sending bits to the interleaver.

1.4.3.2 Puncturing to rate 3/4

In order to obtain a rate 1/2 code from the rate 3/4 code used, the output of the encoder must be punctured by not transmitting 1 bit out of every 3. Puncturing is performed by using a puncturing mask of 1 1 1 0 0 1, applied to the bits output from the encoder. In this notation a 1 indicates that the bit is retained and a 0 indicates that the bit is not transmitted. For an encoder generated sequence of:

$T_1(k), T_2(k), T_1(k+1), T_2(k+1), T_1(k+2), T_2(k+2) ...$

the transmitted sequence would be:

$T_1(k), T_2(k), T_1(k+1), T_1(k+2) ...$

Defining $T_1(0)$, $T_2(0)$ to be the first two bits of the block code generated as defined in section 1.4.2, then the value of k in the above sequences is an integral multiple of 3. The block code is punctured in this manner before being input to the interleaver.

1.4.4 Block interleaver structure

The block interleaver used is designed to separate neighbouring bits in the punctured block code as far as possible over the span of the interleaver with the largest separations resulting for the bits that were originally closest to each other. Because of the 30 different combinations of data rates and interleaver lengths, a flexible interleaver structure is needed.

1.4.4.1 Interleaver size in bits

The interleaver consists of a single dimension array, numbered from 0 to its size in bits –1. The array size depends on both the data rate and interleaver length selected as shown in Table 13.

TABLE 13

Interleaver size in bits as a function of data rate and interleaver length

Data rate (bit/s)	**Interleaver length in frames**					
	1	**3**	**9**	**18**	**36**	**72**
	Interleaver size in bits					
3 200	512	1 536	4 608	9 216	18 432	36 864
4 800	768	2 304	6 912	13 824	27 648	55 296
6 400	1 024	3 072	9 216	18 432	36864	73 728
8 000	1 280	3 840	11 520	23 040	46 080	92 160
9 600	1 536	4 608	13 824	27 648	55 296	110 592

1.4.4.2 Interleaver load

The punctured block code bits are loaded into the interleaver array beginning with location 0. The location for loading each successive bit is obtained from the previous location by incrementing by the "Interleaver increment value" specified in Table 14, modulo the "Interleaver size in bits."

Defining the first punctured block code bit to be B(0), then the load location for B(n) is given by:

Load Location = (n * Interleaver Increment Value) Modulo (Interleaver Size in Bits)

Thus for 3 200 bit/s, with a one frame interleaver (512 bit size with an increment of 97), the first 8 interleaver load locations are: 0, 97, 194, 291, 388, 485, 70, and 167.

TABLE 14

Interleaver increment value as a function of data rate and interleaver length

Data rate (bit/s)	Interleaver length in frames					
	1	3	9	18	36	72
	Interleaver increment value					
3 200	97	229	805	1 393	3 281	6 985
4 800	145	361	1045	2 089	5 137	10 273
6 400	189	481	1393	3 281	6 985	11 141
8 000	201	601	1741	3 481	8 561	14 441
9 600	229	805	2 089	5 137	10 273	17 329

These increment values have been chosen to ensure that the combined cycles of puncturing and assignment of bit positions in each symbol for the specific constellation being used is the same as if there had been no interleaving. This is important, because each symbol of a constellation contains “strong” and “weak” bit positions, except for the lowest data rate. Bit position refers to the location of the bit, ranging from MSB to LSB, in the symbol mapping. A strong bit position is one that has a large average distance between all the constellation points where the bit is a 0 and the closest point where it is a 1. Typically, the MSB is a strong bit and the LSB a weak bit. An interleaving strategy that did not evenly distribute these bits in the way they occur without interleaving could degrade performance.

1.4.4.3 Interleaver fetch

The fetching sequence for all data rates and interleaver lengths start with location 0 of the interleaver array and increment the fetch location by 1. This is a simple linear fetch from beginning to end of the interleaver array.

1.5 Operational features and message protocols

The format of this high data rate waveform has been designed to permit it to work well with most of the protocols used and planned for use with HF. The reinserted preamble facilitates acquisition (or reacquisition) of an ongoing broadcast transmission. The short length of the synchronization preamble, wide range of interleaving lengths, and the use of full-tail-biting coding is intended to provide efficient operation with ARQ protocols. To further enhance the operation with these protocols, the following operational features are included in the HF modem.

1.5.1 Onset of transmission

The modem begins a transmission no later than 100 ms after it has received an entire input data block (enough bits to fill a coded and interleaved block), or upon receipt of the last input data bit, whichever occurs first. The latter would only occur when the message is shorter than one interleaver block. A transmission is defined as beginning with the keying of the radio, followed by the output of the preamble waveform after the configured pre-key delay, if any.

The delay between when the modem receives the first input data bit and the onset of transmission will be highly dependent on the means for delivery of the input data bits to the modem. A synchronous serial interface at the user data rate will have the greatest delay. For this reason it is advisable that a high-speed asynchronous interface (serial or Ethernet port) with flow-control be used if this delay is of concern for the particular application.

1.5.2 End of message

The use of an end-of-message (EOM) in the transmit waveform is a configurable option. When the use of an EOM has been selected, a 32-bit EOM pattern is appended after the last input data bit of the message. The EOM, expressed in hexadecimal notation is 4B65A5B2, where the left most bit is sent first. If the last bit of the EOM does not fill out an input data block, the remaining bits in the input data block is set to zero before encoding and interleaving the block.

If the use of an EOM has been inhibited, and the last input data bit does not fill out an input data block, the remaining bits in the input data block is set to zero before encoding and interleaving the block. It is anticipated that the use of an EOM will only be inhibited when an ARQ data protocol uses ARQ blocks which completely fill (or nearly so) the selected input data block size (interleaver block). Without this feature, the use of an EOM would require the transmission of an additional interleaver block under these circumstances.

1.5.3 Termination of a transmission

The modem should terminate a transmission only after the transmission of the final data frame, including a mini-probe, associated with the final interleaver block. Note that a data frame consists of a 256 symbol data block followed by a mini-probe. Also any signal processing and/or filter delays in the modem and the HF transmitter must be accounted for (as part of the key line control timing) to ensure that the entire final mini-probe is transmitted before the transmitter power is turned off.

1.5.4 Termination of receive data processing

There are a number of events, which can cause the HF modem to cease processing the received signal to recover data, and return to the acquisition mode. These are necessary because a modem is not able to acquire a new transmission while it is attempting to demodulate and decode data.

1.5.4.1 Detection of EOM

The HF modem should always scan all of the decoded bits for the 32-bit EOM pattern defined in section 1.5.2. Upon detection of the EOM, the modem will return to the acquisition mode. The modem should continue to deliver decoded bits to the user (DTE) until the final bit immediately preceding the EOM has been delivered.

1.5.4.2 Receipt of a specified number of data blocks

The maximum message duration measured in number of Input Data Blocks (interleaver blocks) is a configurable parameter. Setting this parameter to zero will specify that an unlimited number may be received. Once the modem has decoded and delivered to the user (DTE), the number of bits corresponding to the configured maximum message duration, the HF modem should return to the acquisition mode and terminate the delivery of decoded bits to the user (DTE). Operation with a specified number of input data blocks may be used by an ARQ protocol where the size of the ARQ packet is fixed, or occasionally changed to accommodate changing propagation conditions. In this case, it is anticipated that this parameter (maximum message duration) should be sent to the receiving end of the link as part of the ARQ protocol. It would then be sent to the receiving modem through the remote control interface since it is not embedded in the waveform itself as the data rate and interleaver length parameters are.

1.6 Performance capabilities

The performance capabilities for the high data rate mode are presented in this section. These test results demonstrate that the modem operates reliably over HF circuits for the channel impairments tested.

1.6.1 Simulator characteristics

The high data rate mode were tested using a baseband HF simulator patterned after the Watterson Model in accordance with Recommendation ITU-R F.1487. Additive white Gaussian noise (AWGN) was used as the noise source. Both signal and noise power was measured in a 3 kHz bandwidth.

1.6.2 Radio filters

Finite impulse response (FIR) filters that reflect radio passband requirements were used. The filter is an N = 63 FIR filter with the following coefficients (read across, then down) and has a sample rate of 16 000 samples per second:

3.4793306E-04	–4.6615634E-05	3.6863006E-05	6.8983925E-04
1.2186785E-03	7.1322870E-04	–6.2685051E-04	–1.1305640E-03
3.8082659E-04	2.2257954E-03	1.0150929E-03	–3.6258003E-03
–6.9094691E-03	–4.2534569E-03	1.1371180E-03	–1.0868903E-04
–1.1312117E-02	–2.2036370E-02	–1.8856425E-02	–4.9115933E-03
–1.3025356E-03	–2.1579735E-02	–4.8379221E-02	–4.8040411E-02
–1.4815010E-02	9.8565688E-03	–2.0275153E-02	–9.0223589E-02
–1.1587973E-01	–2.2672007E-02	1.6315786E-01	3.1537800E-01
3.1537800E-01	1.6315786E-01	–2.2672007E-02	–1.1587973E-01
–9.0223589E-02	–2.0275153E-02	9.8565688E-03	–1.4815010E-02
–4.8040411E-02	–4.8379221E-02	–2.1579735E-02	–1.3025356E-03
–4.9115933E-03	–1.8856425E-02	–2.2036370E-02	–1.1312117E-02
–1.0868903E-04	1.1371180E-03	–4.2534569E-03	–6.9094691E-03
–3.6258003E-03	1.0150929E-03	2.2257954E-03	3.8082659E-04
–1.1305640E-03	–6.2685051E-04	7.1322870E-04	1.2186785E-03
6.8983925E-04	3.6863006E-05	–4.6615634E-05	3.4793306E-04

1.6.3 BER performance

BER performance was measured using radio filters, with the HF channel simulator programmed to simulate the following channels for a BER of 10^{-4}:

- The AWGN channel consists of a single, non-fading path. Each condition was measured for 15 minutes.
- The Rician channel consists of two independent but equal average power paths, with a fixed 2 ms delay between paths. The first path was non-fading. The second was a Rayleigh fading path with a two sigma fading BW of 2 Hz. Each condition was measured for 2 hours.
- The Recommendation ITU-R F.1487 "mid latitudes disturbed" condition (Poor channel) consists of two independent but equal average power Rayleigh fading paths, with a fixed 2 ms delay between paths, and with a two sigma fading BW of 1 Hz. Each condition was measured for two hours.

The measured performance, employing the maximum interleaving period (the 72-frame "Very Long" interleaver), under each of the conditions listed for coded 10^{-4} BER is shown in Table 15.

TABLE 15

High data rate mode performance tests for 10^{-4} BER

User data rate (bit/s)	Average SNR (dB) for BER not to exceed 10^{-4}		
	AWGN channel	Rician channel	Poor channel
12 800*	27	–	–
9 600	21	30	30
8 000	19	25	26
6 400	16	21	23
4 800	13	17	20
3 200	9	12	14

* Optional data rate

1.6.4 Doppler shift performance

During Doppler shift performance test modem acquired and maintained synchronization for at least 5 minutes with a test signal having the following characteristics: 9 600 bit/s/very long interleaver, ±75 Hz frequency offset, 2 ms delay spread, a fading BW of 1 Hz, and an average SNR of 30 dB.

1.7 Associated communications equipment

The QAM constellations described in this Annex are more sensitive to equipment variations than the PSK constellations described elsewhere in this Recommendation. Because of this sensitivity, radio filters will have a significant impact on the performance of modems implementing the high data rate waveform. In addition, because of the level sensitive nature of the QAM constellations, turn-on transients, AGC, and ALC can cause significant performance degradation.

____