The PACTOR-4 Protocol

A Technical Description

PACTOR-4

Protocol Details

1. General information

PACTOR-4 (P4) can generally be considered as an extension of the PACTOR-3 protocol towards a single carrier modulation method (except for speedlevel 1, "chirp mode", see below) with adaptive equalization and RAKE receivers for constructive path superposition. It makes use of a concatenated convolutional code for channel coding. Many mechanisms of the ARQ protocol have been adopted from PACTOR-2/3, as well as cycle times and source coding.

Depending on the channel quality, P4 uses 10 so-called speedlevels. Each speedlevel can be considered as a separate sub-protocol. Modulation type and / or channel coding differ (apart from special exceptions) from speedlevel to speedlevel. Speedlevels thus define their own "waveforms". In the following text speedlevels 1 to 10 will be abbreviated as SL 1 to SL 10.

For P4, as with PACTOR-3, short and long packets are used to ensure the best possible compromise between time quantization ("response time" with many changes of direction in the half-duplex channel) and overhead / idle times.

The "normal" ARQ cycle time for P4 is 1.25 s (short cycles) or 3 * 1.25 s (long cycles). In "Long Path Mode", these values are extended to 1.4 and 3 * 1.4 s. These times are identical to PACTOR-3.

Each packet consists of a header ("preamble") and the subsequent data (symbol) field. The net data field (bit layer) is, as usual with PACTOR, divided into user data area, status byte and two CRC bytes (CCITT CRC-16).

The header length (apart from the special cases "CS3" and "Chirp-Mode", see below) is always 20 symbols, which are spread by a factor of 8 or 16.

In the following text PACTOR is optionally referred to as "P", PACTOR-3 e.g. as P3

2. Data Processing

Generally, the data is processed from the raw data up to the transmission (on the transmission side) according to the following scheme:

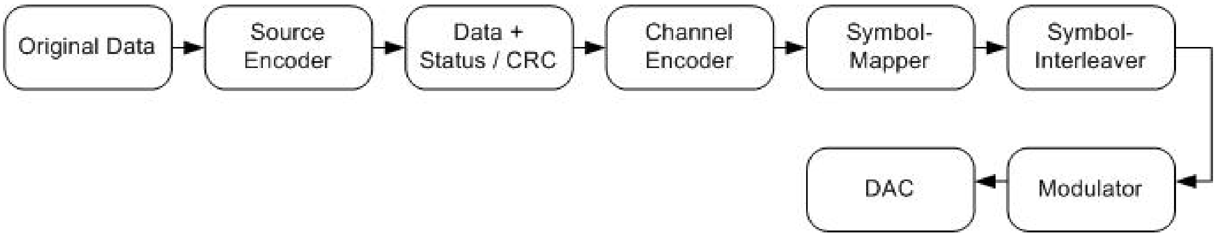


Figure 2.1: Transmission Data Processing

As raw data any data bytes (8 bits) are accepted. Like P3, P4 is "data transparent".
The transition from the "bit layer" to the "symbol layer" is made in the symbol mapper.

3. Basic properties of the speedlevels

The 10 speedlevels have the following basic properties in the data symbol area, refer to table 3.1. For speedlevel 1-4 there are no short information packages.

SL	Modulation	SF	R, long	R, short	N (Bytes), long	N (Bytes), short
1	DBPSK, Chirp	1	1/2	X)	25	X)
2	DQPSK	16	1/2	X)	43	X)
3	DQPSK	16	5/6	X)	72	X)
4	DQPSK	8	5/6	X)	144	X)
5	BPSK	1	1/3	1/3	206	43
6	BPSK	1	5/6	A)	517	43
7	QPSK	1	5/6	1/3	1034	87
8	8-PSK	1	5/6	1/3	1552	131
9	16-QAM	1	5/6	1/3	2069	175
10	32-QAM	1	5/6	A)	2587	175

Table 2.1: Basic properties of the speedlevels

SF means spreading factor; at SF > 1 the modulation is spread, see the section about the packet structure at the symbol layer.

N is the length of the user data field (bit layer) incl. status byte and 2 CRC bytes.

X) These packets do not contain a payload section.

A) Short packets on SL 6 are identical to short packets on SL 5. Short packets on SL 10 are identical to short packets on SL 9. (For speed switching, however, other SNR thresholds are used in the ARQ protocol.)

The speedlevel 1 "long" is called **chirp mode**, speedlevels 2-4 "long" / "short" or 1 "short" are called **robust mode** and speedlevels 5-10 normal mode. This distinction is reasonable, since the modulation of these two groups is fundamentally different.

The symbol rate (after spreading) is always **1800 symbols / s**, except for the narrowband chirp mode. The symbol filter (pulse shaping) uses a quasi-RRC filter with a roll-off factor of 0.33. The filter deviates slightly from the nominal RRC characteristic due to optimization for lowering side lobes and better orthogonality. Pulse shape and spectrum are described and illustrated in more detail in the section about the modulator.

The chirp mode employs a 2-tone chirp DBPSK modulation. The symbol rate in the chirp mode is 66.66 symbols / s. For details refer to the special chapter on chirp mode.

Robust mode uses spread DQPSK modulation. The normal mode uses coherent PSK / QAM modulation with interspersed training sequences.

4. Source Coding

The P4 source coding is identically to the source coding of PACTOR-3, see section 12.

5. Structure of the packets (bit layer)

P4 packets are very similar to P3 packets, except of their difference in payload data lengths:

Figure 5.1: Structure of the packets (net, prior to channel coding, bit layer)

The data bytes in the "Datablock" are followed by the status byte, followed by 2 CRC bytes (CRC-16).

Datablock:

Variable quantity of payload bytes (true payload, without status byte / CRC bytes). The number of real payload data bits per packet is always divisible by 8.

SL	long	short
1	22	X)
2	40	X)
3	69	X)
4	141	X)
5	203	40
6	514	40
7	1031	84
8	1549	128
9	2066	172
10	2584	172

Table 4.1: User data lengths (bytes) of the P4 packets

X) Packets without payload data.

Status byte:

The 8 bits are assigned as follows, LSB at the right side:

E / S1 S0 / M2 M1 M0 / C1 C0

E: Bit for the extended status, see below.

If this bit is set, the extended status appears at the beginning of the packet. For the structure of the extended status, see the following section.

S1 S0: Protocol status

0 0: „nothing“
0 1: BK, equals to „BK bit is set“
1 0: QRT, equals to „QRT bit is set“
1 1: SPUG, equals to „Long-Cycle-/Speedup-Suggestion bit is set“

M2 M1 M0: Mode-Bits, same as P3, but the “reserved mode“ may utilize the new P4 compression (not implemented yet).

000: „ASCII“ (8 bit, „transparent“)
001: Huffman
010: Huffman, „swapped“ (reversed capitalization)
011: reserved, e.g. for later new P4 compression mode
100: Pseudo-Markow, German
101: Pseudo-Markow, German, „swapped“
110: Pseudo-Markow, English
111: Pseudo-Markow, English, „swapped“

C1 C0: Packet counter, same as P1-P3.

CRC:

Same as P3: CCITT-CRC16 with 0xFFFF preassigned CRC register and CRC complement at the end of the CRC calculation. The lower 8 bit of the16 bit CRC will be transmitted first.

6. Structure of the packets (symbol layer)

6.1 Structure of the "Chirp Mode" packets

Refer to section "Chirp Mode".

6.2 Structure of the "Robust Mode" packets

Robust packets, like all PACTOR packets, start with a header (synchronization, packet variant detection). The data symbols to be transmitted follow in the data symbol block. This contains the 8- or 16-times spread DPSK symbols. The data symbol block is always initiated by a single 8- or 16-times spread phase reference symbol.

Symbol spreading can be regarded as part of the modulator but can also be considered as a separate stage of package construction on the symbol layer. Nevertheless, it will be described below. The actual symbol rate without spreading is 1800/8 or 1800/16 symbols /s. That is the symbol rate of the actual information-carrying DPSK symbols. The spread symbol rate is always 1800 symbols/s.

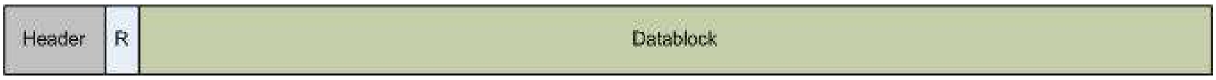


Figure 6.1: "Robust Mode" Packet

Header: One of 19 Chu19 sequences, always spread by factor 16

R: Phase reference symbol (phase 0), spread by factor 8 or 16

Datablock: DQPSK payload symbols, spread by factor 8 or 16 (SF 8, SF 16)

Datablock lengths:

Short:	64,	no user data field
Long, SF 16:	348,	speedlevel 2, 3
Long, SF 8:	696,	speedlevel 4

The spreading of the symbols takes place by complex multiplication with the corresponding spreading sequence.

The following spreading sequences are used:

For spreading factor 16:

```
short scsSpread16[32] =
{
      31159,-10139,32319,5401,32217,5976,29448,14370,
      32609,-3209,11128,30820,-30546,-11858,-12215,-30405,
      32636,2929,-31477,9102,31522,8948,-27518,17789,
      32524,-3981,12148,-30432,-32393,-4939,-986,32752
};
```

For spreading factor 8:

```
short scsSpread8[16] =
{
      32767, 0, 30273, 12539, 0, 32767, -30273, -12539,
      32767, 0, -30273, -12539, 0, 32767, 30273, 12539
};
```

This is C source code. The arrays successively contain the real and imaginary parts of the 16 and 8 symbols of the spreading sequences used.

6.4 Structure of the “Normal Mode” packets

Normal packets, like all PACTOR packets, start with a header (synchronization, packet variant detection). The data symbols to be transmitted are divided into data symbol blocks ("data blocks"); these are each framed by 32 symbol training sequences.

Figure 6.2: „Normal Mode“-Packet (short)

Header:	One of 19 Chu19 sequences, SF8, refer to section on the structure of the header
T:	Training sequence (32 symbols long, 2 * CAZAC-16)
Data block:	Block with user data

Data block lengths (quantity of symbols):

Short:	176
Short_BreaKin:	210
Long:	207

Data blocks / Packet:

Short:	6
Short_BreaKin:	4
Long:	24

Quantity of data symbols (in total):

Short:	6 * 175 = 1056
Short_BreaKin:	4 * 210 = 840
Long:	24 * 207 = 4968

The CAZAC-16 sequences used for the training sequences have the following symbols:

```
short scsTraining[32] =
{
      1, 0, -1, 0, 0, -1, -1, 0, -1, 0, 0, -1, 0, -1, 0,  1,
      1, 0,  1, 0, 0, -1,  1, 0, -1, 0, 0,  1, 0, -1, 0, -1
};
```

This is C source code. The array contains successively the real and imaginary parts of the 16 symbols of the CAZAC-16 sequence C [1-16] being used. The 32-symbol training sequences themselves are created by a onetime cyclic repetition of C, in the way of C[9-16], C[1-16], C[1-9].

The training sequences are consecutively numbered from 1 to 6 and 1 to 24, respectively. For each "odd" training sequence, the normal CAZAC sequence is used for the construction, whereas each "even" training sequence is going to be complex conjugated.

The following C source code shows the construction of the training sequences:

```
//---------------------------------------------------------------------------
// Inserting training symbols (normal mode)
//---------------------------------------------------------------------------

short *burstGenTraining(short *sym, short SequenceNumber)
{
      int i;
      short *training;

      if ( SequenceNumber &  1 )
            training = scsTraining;       // CAZAC-16
      else
            training = scsTrainingCc;     // complex conjugate CAZAC-16

      for (i=16; i < 32; i++)
            *sym++ = (short)( training[i] << 12 );
      for (i=0; i < 32; i++)
            *sym++ = (short)( training[i] << 12 );
      for (i=0; i < 16; i++)
            *sym++ = (short)( training[i] << 12 );

      return(sym);
}
```

7. Channel Coding

Robust mode and normal mode generally use a (partially punctured) concatenated convolutional code for channel coding. (The chirp mode, on the other hand, employs a normal convolutional code, see chirp mode section.) The following describes the channel coding of the robust mode and the normal mode.

The PACTOR IV channel encoder consists of two equal recursive, systematic component-encoders (k = 4) with the generator polynomials $(15, 13)_8$. It is identical to the $(13, 15)_8$ encoder used in the 3GPP W CDMA (UMTS):

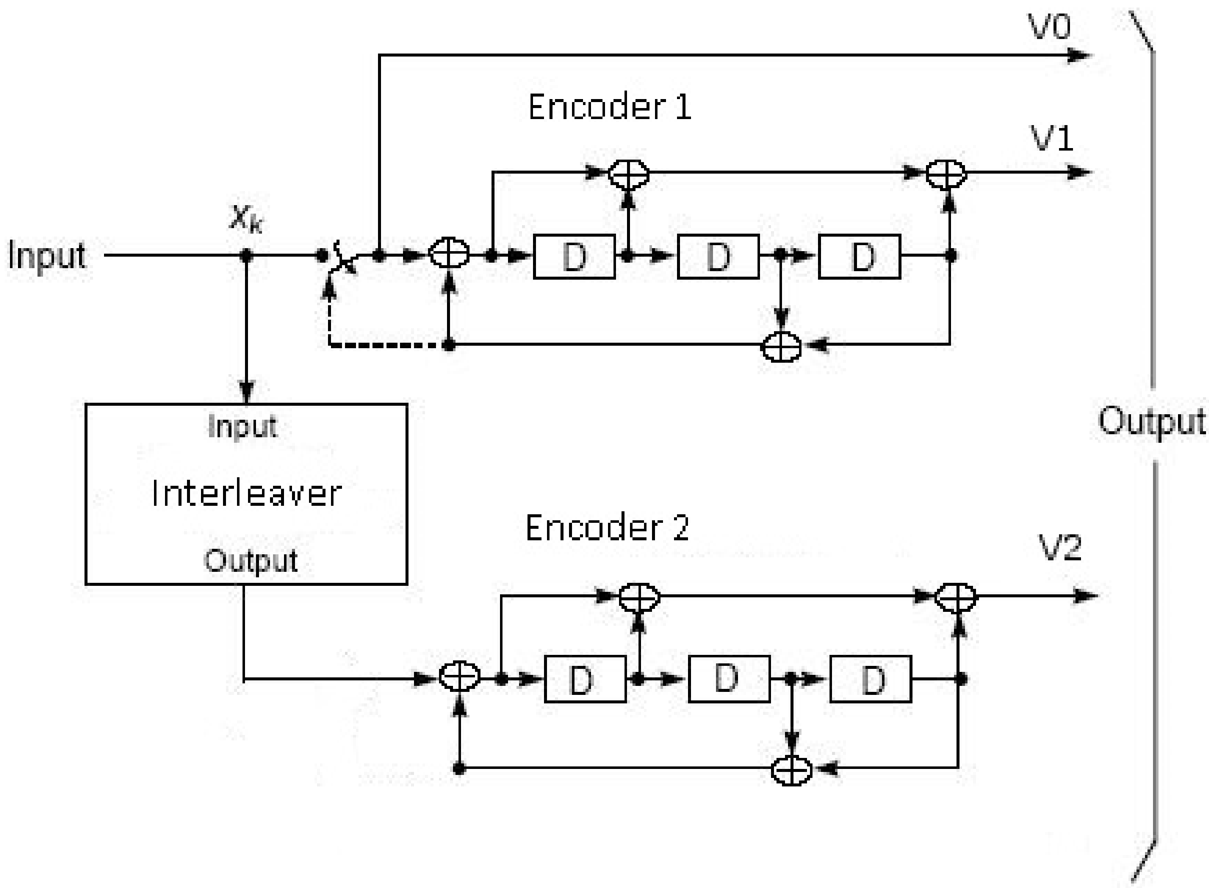


Figure 7.1: Channel Encoder

For the sake of simplicity, the two identical encoders are numbered in the following text; Encoder 1 (C1) is the encoder that receives non-interleaved data as input, Encoder 2 (C2) is the encoder that receives interleaved data as input

.

The switch behind Xk is used to generate the tail bits. Normally, it is connected to the data input Xk. For generating the 3 tail bits it is switched "on", so that only zeros are pushed into the shift register. At V0 the 3 tail bits can be taken from.

After the channel coding, the data bits have the following sequence:

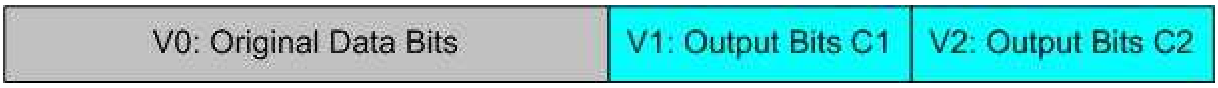


Figure 7.2: Structure of the data area (bit layer) after channel coding

The data fields have the following lengths:

Speedlevel	Data_Long	C1/C2_Long	Data_Short	C1/C2_Short
1	A)	A)	X)	X)
2	348	174	X)	X)
3	580	58	X)	X)
4	1160	116	X)	X)
5	1656	1656	352	352
6	4140	414	352	352
7	8280	828	704	704
8	12420	1242	1056	1056
9	16560	1656	1408	1408
10	20700	2070	1408	1408

Table 7.1: Bit lengths of the data field as well as the two parity fields.

The lengths of the two parity areas are always identical.

A) The chirp mode uses as channel coding the P2 / P3 convolutional code with k = 9.

X) No user data field

8.1 Puncturing

Puncturing matrices:

V0 corresponds with the original data
V1 corresponds with the data from the component encoder 1
V2 corresponds with the data from the component encoder 2

A 1 in the punctuation matrix means that this bit is being transmitted. A 0 means that the bit is discarded.

Rate 1/2:

V0	1 1
V1	1 0
V2	0 1

This means that all original data bits are transmitted un-punctured. However, the "parity bits" from the component encoders are only transmitted alternately, the remaining bits are discarded. Only every odd bit (1st, 3rd, 5th ...) of the component encoder 1 is transmitted, as well as every even bit (2nd, 4th, 6th ...) of the component encoder 2.
The result is a rate of 2 data bits / 4 total bits, which finally is a rate 1/2 code.

Rate 5/6:

V0	1 1 1 1 1 1 1 1 1 1
V1	1 0 0 0 0 0 0 0 0 0
V2	0 0 0 0 1 0 0 0 0 0

This means that all original data bits are transmitted un-punctured. However, only every 10th "parity bit" is transmitted from the component encoders and the remaining bits are discarded. Only every 1st, 11th, 21st, etc. bit of component encoder 1 is transmitted, but only every 5th, 15th, 25th, etc. of component encoder 2.
The result is a rate of 10 data bits / 12 total bits which finally is a rate 5/6 code.

9. Symbol Mapping

The data bytes to be transmitted are construed as a series of individual bits in the arrangement

Byte0_Bit0, Byte0_Bit1, ..., Byte0_Bit7, Byte1_Bit0, Byte1_Bit1, ..., Byte1_Bit7, etc.

These bits are grouped into bit groups for symbol mapping. In BPSK, the "group" consists of only 1 bit each, in 32-QAM it consists of 5 bits.

In the bit groups, the first bit to be transmitted appear on the very left side, the last bit to be sent is bit 0. The first bit group for the transmission of 32-QAM, as an example, has the following arrangement:

Byte0_Bit0, Byte0_Bit1, Byte0_Bit2, Byte0_Bit3, Byte0_Bit4

The „Byte0_Bit4“, therefore represents the LS bit in the bit group for the symbol mapping.

The following constellations are utilized in PACTOR-4:

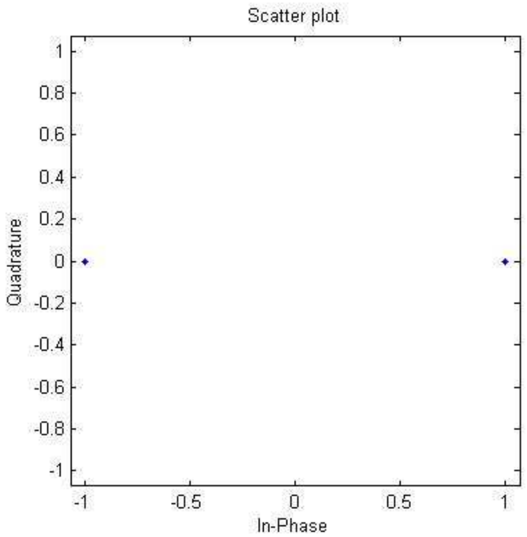


Figure 9.1: BPSK

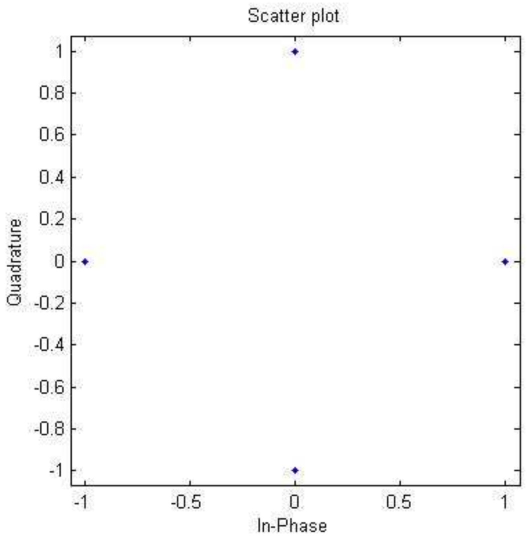


Figure 9.2: QPSK

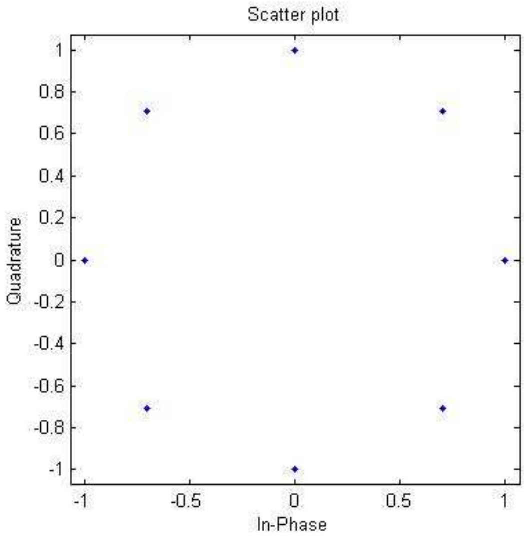


Figure 9.3: 8-PSK

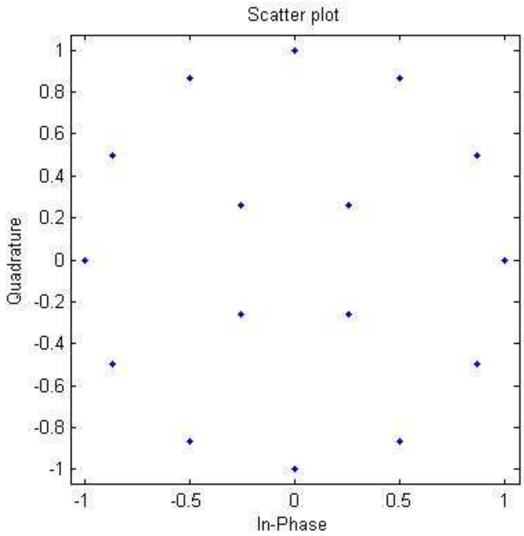


Figure 9.4: 16-QAM

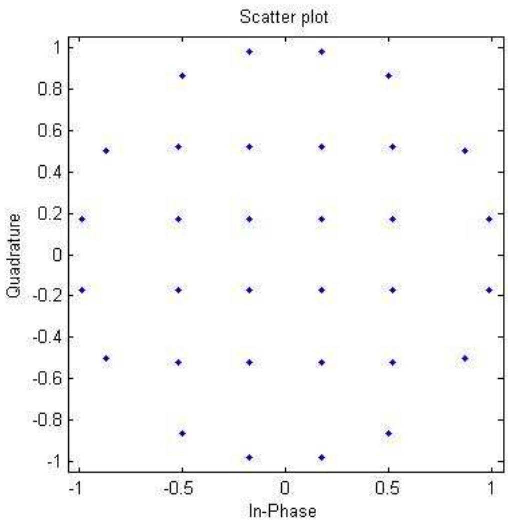


Figure 9.5: 32-QAM

10. Modulator

10.1 Normal Mode, Robust Mode

In the modulator, the 1800 Bd pulses are shaped with a pseudo-RC filter and finally shifted in frequency by +1500 Hz. The center frequency of the (real) audio signal of the DR-7800 modem is 1500 Hz. The bandwidth of the signal is approximately 2400 Hz (@ -25 dB).

P4 pseudo RRC symbol filter for 1800 Bd symbols:

// RRCF with 16 real samples per symbol (T)

```
short scsRrcf[129] =
{
      226, 215, 194, 161, 117, 61, -6, -84,
      -171, -263, -356, -446, -525, -586, -623, -627,
      -592, -514, -388, -215, 4, 262, 550, 855,
      1162, 1452, 1705, 1898, 2013, 2030, 1933, 1711,
      1358, 875, 270, -440, -1230, -2070, -2919, -3733,
      -4464, -5058, -5466, -5638, -5530, -5103, -4330, -3194,
      -1689, 174, 2374, 4873, 7621, 10554, 13601, 16682,
      19712, 22603, 25270, 27634, 29621, 31168, 32228, 32767,
      32767, 32228, 31168, 29621, 27634, 25270, 22603, 19712,
      16682, 13601, 10554, 7621, 4873, 2374, 174, -1689,
      -3194, -4330, -5103, -5530, -5638, -5466, -5058, -4464,
      -3733, -2919, -2070, -1230, -440, 270, 875, 1358,
      1711, 1933, 2030, 2013, 1898, 1705, 1452, 1162,
      855, 550, 262, 4, -215, -388, -514, -592,
      -627, -623, -586, -525, -446, -356, -263, -171,
      -84, -6, 61, 117, 161, 194, 215, 226,
      0
};
```

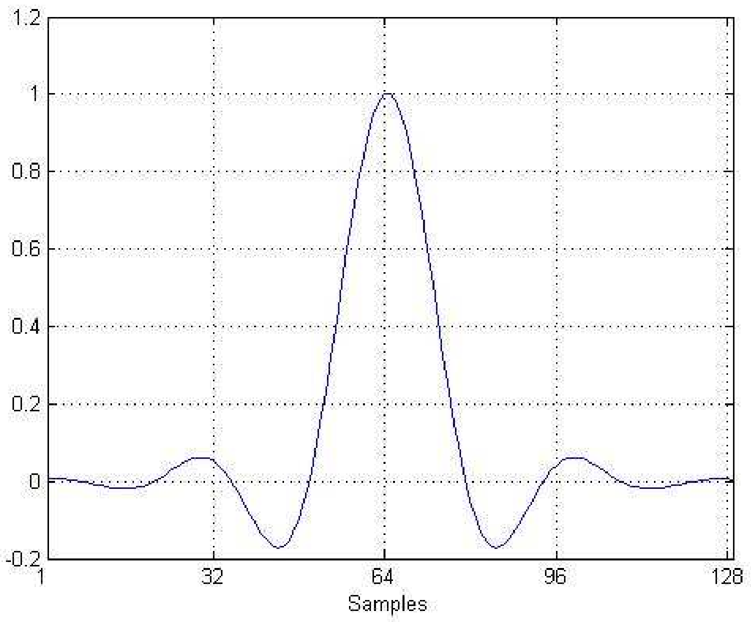


Figure 10.1: P4-RRC pulse in time domain

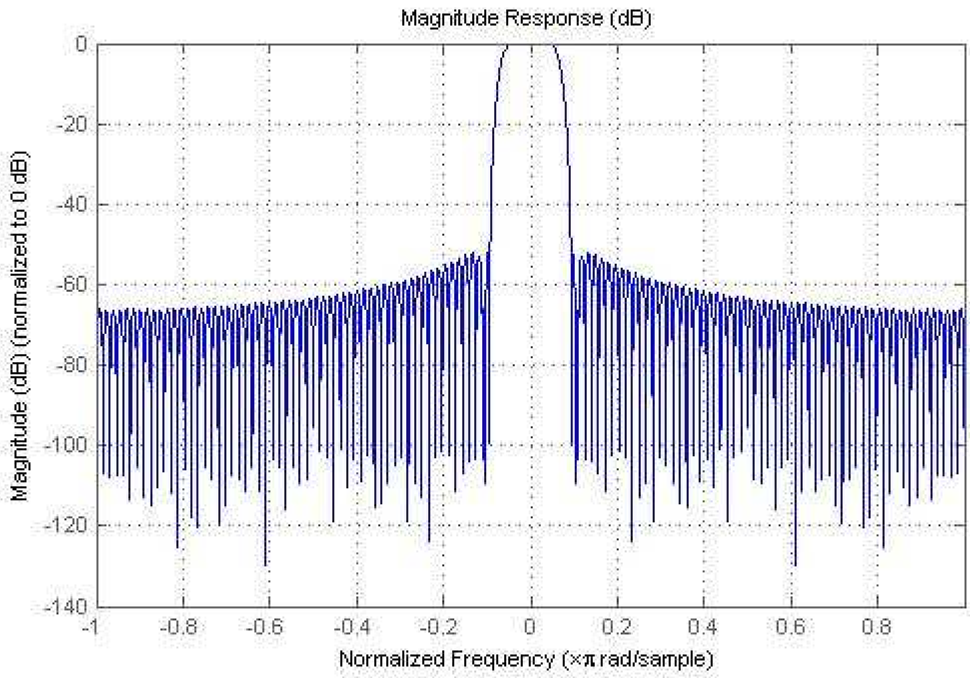


Figure 10.2: spectrum of a P4-RRC pulse

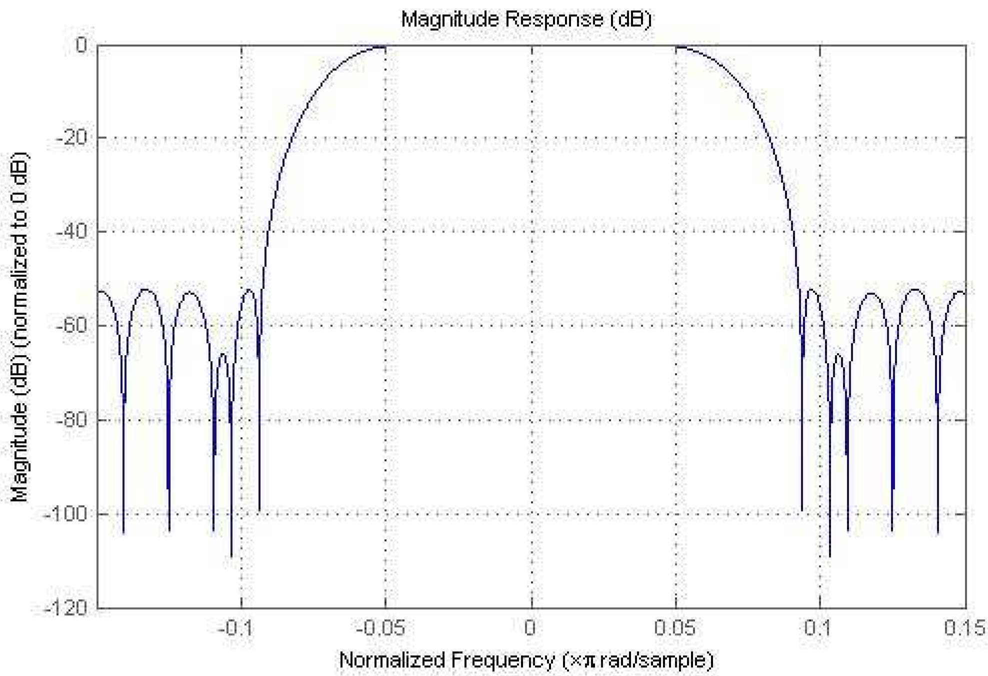


Figure 10.3: Detailed view of the spectrum

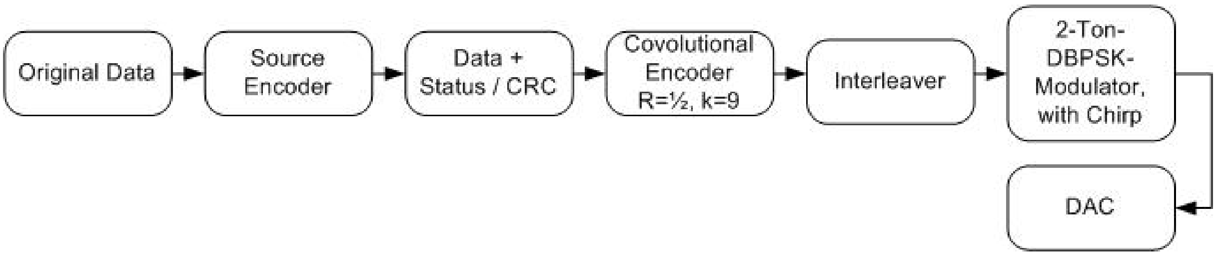
10.2 Modulator of the Chirp Mode

For details to the modulator of the Chirp Mode, refer to chapter „Chirp Mode".

11. Chirp Mode

Data transfer at the lowest PACTOR-4 speedlevel (speedlevel 1, SL 1) is generally performed as a two-tone chirp mode with long packets (normal cycle time 3.75 s).
The PACTOR-4 speedlevel 1 is based on the PACTOR-2 speedlevel 1. The main difference is the chirp of the two carrier frequencies.

11.1 Basic Signal Flow (transmission side)



11.2 Packet Construction / Packet Length

The basic packet construction (net data field, bit layer) is as always on PACTOR:

Header, 22 payload data bytes, 1 status byte, 2 CRC bytes (+1 flush byte)

The CRC corresponds to the usual PACTOR-CRC16.

11.3 Header

Same as PACTOR-2 header for speedlevel 1 / long.

11.4 Source-Encoder

Corresponds with the normal P4 source encoder, and with this is identically with the P3 source encoder, see section 12.

11.5 Convolutional Coding

k = 9, R= ½, same as P2, speedlevel 1.

The polynomials are

G1=111101011;
G2=101110001.

11.6 Interleaver

Same as P2, speedlevel 1.

Interleaving takes place basically as block-bit interleaving. The binary “send data” record of a packet is transferred as a block to the block interleaver. The permutation pointer P of the interleaver is calculated according to the following algorithm:

```
S=1;                                    // auxiliary variable
P=0;                                    // initial permutation pointer
M=INTERLEAVER_DEPTH;

for(I=0;I<PACKET_SIZE;I++)      // loop over all bits
{

//        OUTPUT_PACKET(I)=INPUT_PACKET(P);        // perform the permutation.

          P=P+M;
          if (P>PACKET_SIZE)
                    P=S++;

}
```

PACKET_SIZE keeps the quantity of bits of the packet to be permutated.
INTERLEAVER_DEPTH represents the block interleaver depth = 16 at speedlevel 1.

Usually the PACKET_SIZE is dividable by INTERLEAVER_DEPTH.
(Generally: INTERLEAVER_DEPTH should be >= 2(k-1) of the convolutional code being used.)

INPUT_PACKET is the packet to be permutated (bitwise arrangement, Bit 0 is the LSB, the first bit of the packet).

OUTPUT_PACKET is the packet having been permutated (bit wise arrangement, Bit 0 is the LSB, the first bit of the packet).

11.7 Modulation

2 carriers, modulated with 66,66 Bd (T = 15 ms) DBPSK each.
The utilized differential phases are PI/4 when a 0 is being transferred, and 5*PI/4 when a 1 is being transferred – just like P2.

Symbols on the carrier with the lower frequency always appear **delayed by T / 2**.

11.7.1 Carrier Frequencies (before start of the Chirp)

550 and 1530 Hz.

11.7.2 Chirp

Starts exactly 4,5 T (67,5 ms) after the start of the last header symbol, this happens on both carriers simultaneously.

11.7.3 Chirp rate

294,0 Hz/s.

11.8 Symbol Filter

Very similar to the RRC pulse, but optimized concerning spectral side lobes and orthogonality:

```
float scsRrc_chirp[32] =
{
      0.0013033, 0.00099339, 0.00015167, -0.0021225,
      -0.0058615, -0.01035, -0.014008, -0.014558,
      -0.0095413, 0.002945, 0.023476, 0.050767,
      0.081603, 0.11134, 0.13489, 0.14791, 0.14791,
      0.13489, 0.11134, 0.081603, 0.050767, 0.023476,
      0.002945, -0.0095413, -0.014558, -0.014008,
      -0.01035, -0.0058615, -0.0021225, 0.00015167,
      0.00099339, 0.0013033
};
```

It is C source code. The array contains 32 real coefficients, 8 coefficients per symbol (T). The maximum value is 0.14791 (normalization to the coefficient sum).

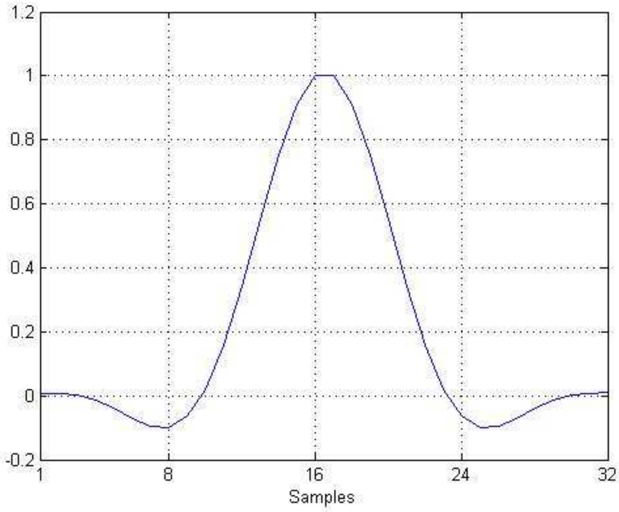


Figure 14.1: Quasi-RRC pulse in time domain

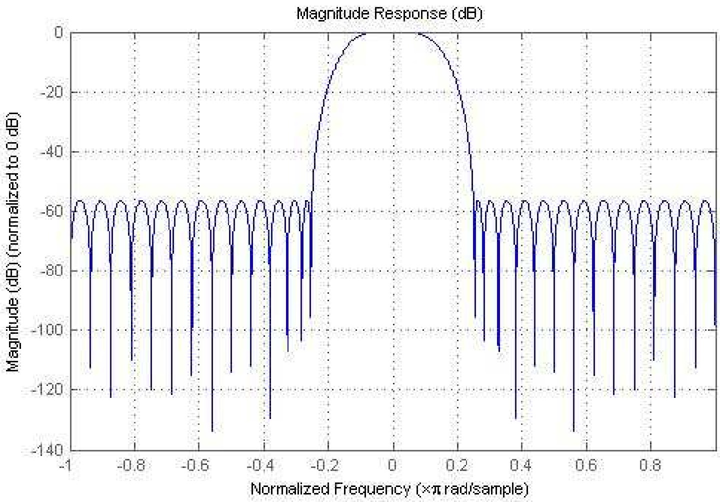


Figure 14.2: spectrum of the des Quasi-RRC pulse

11.11 Quantity of Symbols

2 reference symbols (1 reference symbol per carrier),
16 header symbols (8 header symbols per carrier),
416 user data symbols (208 user data symbols per carrier).

Packet length (physically) at an impulse filter of 4*T length:
216 * T + 4 * T + T/2 = 3307,5 ms.

11.12 Throughput / Net data length

208 symbols per data field * 2 carrier = 416 total symbols (bits).
After convolutional coding: 416 / 2 = 208 bits = 26 bytes.
26 bytes – flush byte – status byte – CRC bytes = 26 – 4 = 22 user bytes.
22 bytes / 3,75 s = 5,8666 bytes / s = **46,933 bps**.

12. Advanced PACTOR Data Compression

P2, P3 and P4 use automatic online data compression comprising Huffman and run-length encoding as well as Pseudo-Markov Compression (PMC, see below). The information sending system (ISS) automatically checks, whether one of these compression modes or the original ASCII code leads to the shortest data package, which depends on the probability of occurrence of the characters. Hence, there is no risk of losing throughput capacity. In case of a binary data transfer, the on-line data compression normally switches off automatically due to the character distribution. An external data compression in the terminal program is usually performed instead.

Huffman compression exploits the “one-dimensional” probability distribution of the characters in plain texts. The more frequently a character occurs, the shorter its Huffman symbol has to be. More details including the code table used in the PACTOR protocols can be found in the description of the PACTOR-I standard.

Markov compression can be considered as a “double” Huffman compression, since it not only makes use of the simple probability distribution, but of the “two-dimensional” probability. For each preceding character, a probability distribution of the very next character can be calculated. For example, if the actual character is “e”, it is very likely that “i” or “s” occurs next, but extremely unlikely that an “X” follows. The resulting probability distributions are much more concentrated than the simple one-dimensional distribution and thus lead to a considerably better compression. Unfortunately, there are two drawbacks: Since for each ASCII character a separate coding table is required, the entire Markov coding table becomes impractically large. Additionally, the two-dimensional distribution and thus the achievable compression factor depends much more on the kind of text than the simple character distribution. We have therefore chosen a slightly modified approach which we called Pseudo-Markov Compression (PMC), because it can be considered as a hybrid between Markov- and Huffman encoding. In PMC, the Markov encoding is limited to the 16 most frequent “preceding” characters. All other characters trigger normal Huffman compression of the very next character. This reduces the Markov coding table to a reasonable size and also makes the character probabilities less critical, since especially the less frequent characters tend to have unstable probability distributions. Nevertheless, for optimum compression, two different tables for English and German texts are defined in the PACTOR-2 and -3 protocols and automatically chosen. When transferring plain text, PMC yields a compression factor of around 1.9 compared to 8-bit ASCII.

Run-length encoding allows the effective compression of longer sequences of identical bytes. The special prefix byte “0x1D” is defined, which initiates a 3 byte run length code. The second byte is called the “code byte” and contains the original code of the transferred byte within the range of the entire ASCII character set. The third byte provides the number of code bytes to be displayed on the receiving side within the range between “0x01” to “0x60”. Values between “0x00” and “0x1f” are transferred as “0x60” to “0x7f”, values between “0x20” and “0x60” are transferred without any change. For example, the sequence “AAAAAAAA” is transferred using the 3 byte run-length code “0x1D 0x41 0x68”.

```
#define IDLE 0x1E

/*****************************************************************
* printr
*
* generic data output routine
*****************************************************************/

void printr(unsigned char outbyte)
{
	// put outbyte to display or whatever output device

	// printf("%c",outbyte);
}


/*****************************************************************
 * ascii_decoder
 *
 * rxpackx: pointer to start of original (encoded) data field
 * rxlen: number of bytes contained in original (encoded) data field
 *****************************************************************/

void ascii_decoder(unsigned char *rxpackx, unsigned int rxlen)
{
	unsigned int i;

	for (i = 0; i < rxlen; i++)
	{
		if (rxpackx[i] != IDLE)
		{
			printr (rxpackx[i]);
		}
	}
}
```

```
/*****************************************************************
 * huffman_decoder
 *
 * rxpackx: pointer to start of original (encoded) data field
 * rxlen: number of bytes contained in original (encoded) data field
 *****************************************************************/

void huffman_decoder(unsigned char *rxpackx,unsigned int rxlen)
{
        unsigned int i, j;
        unsigned short offset;
        unsigned char curbyte, outbyte;

        offset = 0;

        for (i = 0; i < rxlen; i++)
        {
                curbyte = rxpackx[i];           // next input byte

                for (j = 0; j < 8; j++)
                {
                        if (curbyte & 1)
                                offset += 2;
                        offset = huftab[offset >> 1];
                        curbyte >>= 1;

                        if (offset > 767)               // data ready?
                        {
                                outbyte = offset & 0xff;
                                if (outbyte != IDLE)
                                {
                                        printr (outbyte);
                                }
                                else
                                        offset = 0;             // next output byte
                        }
                }
        }
}
```

```
/*****************************************************************
 * pmc_decoder
 *
 * rxpackx: pointer to start of original (encoded) data field
 * rxlen: number of bytes contained in original (encoded) data field
 * english: 0=German data, 1=English data
 *****************************************************************/

void pmc_decoder(unsigned char *rxpackx,unsigned int rxlen, unsigned char english)
{
	unsigned int i, j;
	unsigned short table_offset = 0, offset = 0;
	unsigned char curbyte, outbyte;

	for (i = 0;i < rxlen; i++)
	{
		curbyte = rxpackx[i];		// next input byte

		for (j = 0; j < 8; j++)
		{
			if (curbyte & 1)
				offset += 2;
			offset = huftab[(table_offset + offset) >> 1];
			curbyte >>= 1;

			if (offset > 767)		// data ready?
			{
				outbyte = offset & 0xff;

				table_offset = reftab[outbyte];
				if (table_offset)
				{
					if (english)
						table_offset += 8128;
				}

				if (outbyte != IDLE)
				{
					printr (outbyte);
				}

				offset = 0;		// next output byte
			}
		}
	}
}

// Huffman/PMC-Decoding tables

unsigned short huftab[]=
	{
```

4,8,12,16,800,20,24,28,32,869,36,40,
44,48,52,56,60,878,64,873,882,68,884,72,
76,80,883,84,88,868,865,92,96,100,104,885,
876,108,872,112,116,871,120,877,781,778,879,867,
124,128,132,136,140,144,866,870,148,152,887,836,
156,875,160,164,890,168,814,812,172,176,180,184,
188,851,192,196,200,204,208,833,837,212,880,886,
216,816,838,220,224,834,228,835,841,232,852,236,
847,240,244,248,848,252,256,260,817,264,850,268,
272,276,280,808,809,844,846,284,858,288,845,292,
825,296,855,821,889,300,818,819,820,822,823,824,
840,304,842,853,854,796,888,308,843,831,829,312,
316,320,881,849,874,839,324,813,328,826,332,336,
340,344,348,801,815,352,356,360,810,364,802,805,
807,368,863,372,376,380,384,806,811,830,832,804,
828,856,388,803,392,396,400,404,408,412,416,420,
424,428,432,857,436,440,444,448,827,860,452,456,
460,464,468,472,476,480,484,488,492,496,500,504,
859,861,895,894,893,892,891,864,862,799,797,795,
793,792,791,790,789,788,787,786,785,784,798,783,
782,780,779,777,776,775,774,773,772,771,770,769,
768,794,

// german rx-markov-table follows

4,8,12,800,16,20,24,28,32,36,868,40,
44,48,52,56,60,869,64,68,72,76,80,84,
88,865,92,883,96,100,104,108,887,885,112,116,
866,890,873,120,836,848,124,851,128,886,833,132,
877,136,834,140,845,871,144,148,152,837,878,870,
156,854,160,875,850,164,168,838,172,176,852,180,
843,184,188,192,844,872,196,200,204,839,840,808,
208,876,212,835,879,817,216,781,220,224,841,855,
874,784,228,232,882,236,853,880,240,244,818,867,
822,248,252,813,858,256,811,820,821,824,847,856,
859,260,264,884,268,790,819,846,272,825,276,280,
284,288,881,292,823,842,296,300,304,308,312,316,
320,324,807,816,888,328,332,336,340,344,348,352,
356,360,364,368,372,376,380,384,388,392,396,400,
404,408,412,416,420,424,428,432,436,440,444,448,
452,456,460,464,468,472,476,480,484,488,492,496,
500,504,768,769,770,771,772,773,774,775,776,777,
778,779,780,782,783,785,786,787,788,789,791,792,
793,794,795,796,797,798,799,801,802,803,804,805,
806,809,810,812,814,815,826,827,828,829,830,831,
832,849,857,860,861,862,863,864,889,891,892,893,
894,895,
4,8,12,16,882,878,20,873,24,800,28,32,
36,883,40,44,876,48,884,52,56,877,60,64,
68,72,76,872,866,80,84,871,870,781,868,88,
867,885,92,875,96,100,104,812,108,813,888,112,

881,890,116,814,887,791,120,880,124,869,865,886,
128,132,136,140,144,148,152,156,160,164,809,879,
168,172,176,180,184,188,192,196,200,204,208,212,
216,220,224,228,232,236,240,244,248,252,256,260,
264,268,272,276,280,284,288,292,296,300,304,308,
312,316,320,324,328,332,336,340,344,348,352,356,
360,364,368,372,376,380,384,388,392,396,400,404,
408,412,416,420,424,428,432,436,440,444,448,452,
456,460,464,468,472,476,480,484,488,492,496,500,
504,768,769,770,771,772,773,774,775,776,777,778,
779,780,782,783,784,785,786,787,788,789,790,792,
793,794,795,796,797,798,799,801,802,803,804,805,
806,807,808,810,811,815,816,817,818,819,820,821,
822,823,824,825,826,827,828,829,830,831,832,833,
834,835,836,837,838,839,840,841,842,843,844,845,
846,847,848,849,850,851,852,853,854,855,856,857,
858,859,860,861,862,863,864,874,889,891,892,893,
894,895,
4,8,12,800,16,20,24,871,28,868,32,869,
36,40,44,884,48,52,878,56,883,60,865,814,
64,885,812,873,68,890,72,781,76,80,84,875,
88,872,879,92,870,96,876,100,813,104,108,112,
866,886,116,783,120,124,887,128,784,880,877,132,
136,140,809,826,882,144,148,152,156,160,164,168,
807,867,172,176,180,184,188,192,196,200,204,208,
212,216,220,224,228,232,236,240,244,248,252,256,
260,264,268,272,276,280,284,288,292,296,300,304,
308,312,316,320,324,328,332,336,340,344,348,352,
356,360,364,368,372,376,380,384,388,392,396,400,
404,408,412,416,420,424,428,432,436,440,444,448,
452,456,460,464,468,472,476,480,484,488,492,496,
500,504,768,769,770,771,772,773,774,775,776,777,
778,779,780,782,785,786,787,788,789,790,791,792,
793,794,795,796,797,798,799,801,802,803,804,805,
806,808,810,811,815,816,817,818,819,820,821,822,
823,824,825,827,828,829,830,831,832,833,834,835,
836,837,838,839,840,841,842,843,844,845,846,847,
848,849,850,851,852,853,854,855,856,857,858,859,
860,861,862,863,864,874,881,888,889,891,892,893,
894,895,
4,8,12,16,869,878,20,24,884,28,32,871,
36,867,883,40,44,879,876,48,877,52,882,56,
886,60,800,64,875,68,72,880,76,865,80,866,
868,84,88,92,872,96,100,781,104,813,890,108,
870,112,116,120,124,791,885,128,132,136,140,144,
148,152,815,873,888,156,160,164,168,172,176,180,
184,188,192,196,200,204,208,212,216,220,224,228,
232,236,240,244,248,252,256,260,264,268,272,276,
280,284,288,292,296,300,304,308,312,316,320,324,
328,332,336,340,344,348,352,356,360,364,368,372,
376,380,384,388,392,396,400,404,408,412,416,420,

424,428,432,436,440,444,448,452,456,460,464,468,
472,476,480,484,488,492,496,500,504,768,769,770,
771,772,773,774,775,776,777,778,779,780,782,783,
784,785,786,787,788,789,790,792,793,794,795,796,
797,798,799,801,802,803,804,805,806,807,808,809,
810,811,812,814,816,817,818,819,820,821,822,823,
824,825,826,827,828,829,830,831,832,833,834,835,
836,837,838,839,840,841,842,843,844,845,846,847,
848,849,850,851,852,853,854,855,856,857,858,859,
860,861,862,863,864,874,881,887,889,891,892,893,
894,895,
4,8,12,800,16,20,24,28,32,869,36,40,
44,48,868,879,873,52,884,56,865,60,883,64,
68,72,885,76,866,870,878,80,84,781,890,871,
88,877,92,867,96,875,887,813,100,876,104,814,
872,783,812,108,112,882,784,116,120,880,124,128,
132,809,136,140,144,148,152,156,160,164,168,172,
176,180,184,188,192,196,200,204,208,212,216,220,
224,228,232,236,240,244,248,252,256,260,264,268,
272,276,280,284,288,292,296,300,304,308,312,316,
320,324,328,332,336,340,344,348,352,356,360,364,
368,372,376,380,384,388,392,396,400,404,408,412,
416,420,424,428,432,436,440,444,448,452,456,460,
464,468,472,476,480,484,488,492,496,500,504,768,
769,770,771,772,773,774,775,776,777,778,779,780,
782,785,786,787,788,789,790,791,792,793,794,795,
796,797,798,799,801,802,803,804,805,806,807,808,
810,811,815,816,817,818,819,820,821,822,823,824,
825,826,827,828,829,830,831,832,833,834,835,836,
837,838,839,840,841,842,843,844,845,846,847,848,
849,850,851,852,853,854,855,856,857,858,859,860,
861,862,863,864,874,881,886,888,889,891,892,893,
894,895,
4,8,12,869,16,800,20,24,28,873,32,36,
882,40,44,865,48,52,56,890,885,814,60,64,
68,884,812,883,813,72,887,76,879,80,781,84,
876,88,92,96,870,100,104,108,880,112,784,872,
116,120,783,124,866,867,871,128,809,878,132,875,
877,886,889,136,140,144,826,148,152,156,160,164,
168,807,172,176,180,184,188,192,196,200,204,208,
212,216,220,224,228,232,236,240,244,248,252,256,
260,264,268,272,276,280,284,288,292,296,300,304,
308,312,316,320,324,328,332,336,340,344,348,352,
356,360,364,368,372,376,380,384,388,392,396,400,
404,408,412,416,420,424,428,432,436,440,444,448,
452,456,460,464,468,472,476,480,484,488,492,496,
500,504,768,769,770,771,772,773,774,775,776,777,
778,779,780,782,785,786,787,788,789,790,791,792,
793,794,795,796,797,798,799,801,802,803,804,805,
806,808,810,811,815,816,817,818,819,820,821,822,
823,824,825,827,828,829,830,831,832,833,834,835,

836,837,838,839,840,841,842,843,844,845,846,847,
848,849,850,851,852,853,854,855,856,857,858,859,
860,861,862,863,864,868,874,881,888,891,892,893,
894,895,
4,8,12,16,20,884,800,24,28,869,32,36,
867,873,883,879,40,44,880,48,52,56,865,60,
64,68,72,76,781,813,80,871,885,84,88,886,
887,92,870,812,96,866,100,814,104,875,108,889,
882,890,112,876,116,120,868,877,124,872,878,128,
132,136,140,807,826,144,148,152,156,160,164,168,
172,176,180,184,188,192,196,200,204,208,212,216,
220,224,228,232,236,240,244,248,252,256,260,264,
268,272,276,280,284,288,292,296,300,304,308,312,
316,320,324,328,332,336,340,344,348,352,356,360,
364,368,372,376,380,384,388,392,396,400,404,408,
412,416,420,424,428,432,436,440,444,448,452,456,
460,464,468,472,476,480,484,488,492,496,500,504,
768,769,770,771,772,773,774,775,776,777,778,779,
780,782,783,784,785,786,787,788,789,790,791,792,
793,794,795,796,797,798,799,801,802,803,804,805,
806,808,809,810,811,815,816,817,818,819,820,821,
822,823,824,825,827,828,829,830,831,832,833,834,
835,836,837,838,839,840,841,842,843,844,845,846,
847,848,849,850,851,852,853,854,855,856,857,858,
859,860,861,862,863,864,874,881,888,891,892,893,
894,895,
4,869,8,12,873,800,16,20,24,865,28,885,
32,876,36,40,882,44,48,781,52,879,56,60,
887,64,812,813,68,814,866,72,76,80,878,883,
84,88,92,889,96,100,104,108,784,872,875,886,
890,112,116,120,124,128,132,136,140,144,148,152,
156,160,164,168,172,176,180,184,188,192,196,200,
204,208,212,216,220,224,228,232,236,240,244,248,
252,256,260,264,268,272,276,280,284,288,292,296,
300,304,308,312,316,320,324,328,332,336,340,344,
348,352,356,360,364,368,372,376,380,384,388,392,
396,400,404,408,412,416,420,424,428,432,436,440,
444,448,452,456,460,464,468,472,476,480,484,488,
492,496,500,504,768,769,770,771,772,773,774,775,
776,777,778,779,780,782,783,785,786,787,788,789,
790,791,792,793,794,795,796,797,798,799,801,802,
803,804,805,806,807,808,809,810,811,815,816,817,
818,819,820,821,822,823,824,825,826,827,828,829,
830,831,832,833,834,835,836,837,838,839,840,841,
842,843,844,845,846,847,848,849,850,851,852,853,
854,855,856,857,858,859,860,861,862,863,864,867,
868,870,871,874,877,880,881,884,888,891,892,893,
894,895,
4,8,12,16,20,878,24,885,876,28,884,32,
882,36,40,44,883,48,867,52,877,56,866,60,
64,800,872,868,68,72,875,871,791,76,80,84,

88,889,888,92,873,890,96,781,870,880,814,100,
104,108,813,112,116,120,124,815,886,128,132,136,
140,144,148,152,156,160,164,168,172,176,180,184,
188,192,196,200,204,208,212,216,220,224,228,232,
236,240,244,248,252,256,260,264,268,272,276,280,
284,288,292,296,300,304,308,312,316,320,324,328,
332,336,340,344,348,352,356,360,364,368,372,376,
380,384,388,392,396,400,404,408,412,416,420,424,
428,432,436,440,444,448,452,456,460,464,468,472,
476,480,484,488,492,496,500,504,768,769,770,771,
772,773,774,775,776,777,778,779,780,782,783,784,
785,786,787,788,789,790,792,793,794,795,796,797,
798,799,801,802,803,804,805,806,807,808,809,810,
811,812,816,817,818,819,820,821,822,823,824,825,
826,827,828,829,830,831,832,833,834,835,836,837,
838,839,840,841,842,843,844,845,846,847,848,849,
850,851,852,853,854,855,856,857,858,859,860,861,
862,863,864,865,869,874,879,881,887,891,892,893,
894,895,
4,878,8,12,16,20,24,883,28,882,870,32,
36,40,800,44,877,48,867,884,869,52,56,876,
791,871,60,813,64,68,880,72,868,76,80,84,
865,875,88,866,872,873,92,96,885,100,104,108,
112,812,890,116,120,124,128,132,136,140,144,148,
152,156,160,164,168,172,176,180,184,188,192,196,
200,204,208,212,216,220,224,228,232,236,240,244,
248,252,256,260,264,268,272,276,280,284,288,292,
296,300,304,308,312,316,320,324,328,332,336,340,
344,348,352,356,360,364,368,372,376,380,384,388,
392,396,400,404,408,412,416,420,424,428,432,436,
440,444,448,452,456,460,464,468,472,476,480,484,
488,492,496,500,504,768,769,770,771,772,773,774,
775,776,777,778,779,780,781,782,783,784,785,786,
787,788,789,790,792,793,794,795,796,797,798,799,
801,802,803,804,805,806,807,808,809,810,811,814,
815,816,817,818,819,820,821,822,823,824,825,826,
827,828,829,830,831,832,833,834,835,836,837,838,
839,840,841,842,843,844,845,846,847,848,849,850,
851,852,853,854,855,856,857,858,859,860,861,862,
863,864,874,879,881,886,887,888,889,891,892,893,
894,895,
4,8,869,12,16,20,876,24,28,873,32,884,
36,40,865,883,44,800,48,879,52,885,56,60,
64,68,867,72,76,80,84,868,88,871,866,92,
781,96,783,812,814,100,104,870,880,813,108,112,
116,120,784,878,882,887,124,128,132,826,875,886,
889,136,140,144,791,809,872,890,148,152,156,160,
164,168,172,176,180,184,188,192,196,200,204,208,
212,216,220,224,228,232,236,240,244,248,252,256,
260,264,268,272,276,280,284,288,292,296,300,304,
308,312,316,320,324,328,332,336,340,344,348,352,

356,360,364,368,372,376,380,384,388,392,396,400,
404,408,412,416,420,424,428,432,436,440,444,448,
452,456,460,464,468,472,476,480,484,488,492,496,
500,504,768,769,770,771,772,773,774,775,776,777,
778,779,780,782,785,786,787,788,789,790,792,793,
794,795,796,797,798,799,801,802,803,804,805,806,
807,808,810,811,815,816,817,818,819,820,821,822,
823,824,825,827,828,829,830,831,832,833,834,835,
836,837,838,839,840,841,842,843,844,845,846,847,
848,849,850,851,852,853,854,855,856,857,858,859,
860,861,862,863,864,874,877,881,888,891,892,893,
894,895,
4,8,12,16,869,800,20,24,28,32,36,882,
865,40,878,884,44,876,48,52,56,873,60,879,
64,68,72,781,883,885,76,887,80,84,783,814,
88,877,870,890,812,813,875,92,866,96,100,104,
871,108,112,784,827,868,116,120,124,128,132,136,
140,144,148,152,156,160,164,168,172,176,180,184,
188,192,196,200,204,208,212,216,220,224,228,232,
236,240,244,248,252,256,260,264,268,272,276,280,
284,288,292,296,300,304,308,312,316,320,324,328,
332,336,340,344,348,352,356,360,364,368,372,376,
380,384,388,392,396,400,404,408,412,416,420,424,
428,432,436,440,444,448,452,456,460,464,468,472,
476,480,484,488,492,496,500,504,768,769,770,771,
772,773,774,775,776,777,778,779,780,782,785,786,
787,788,789,790,791,792,793,794,795,796,797,798,
799,801,802,803,804,805,806,807,808,809,810,811,
815,816,817,818,819,820,821,822,823,824,825,826,
828,829,830,831,832,833,834,835,836,837,838,839,
840,841,842,843,844,845,846,847,848,849,850,851,
852,853,854,855,856,857,858,859,860,861,862,863,
864,867,872,874,880,881,886,888,889,891,892,893,
894,895,
4,8,869,12,800,16,20,24,28,883,32,36,
40,44,884,885,876,873,882,48,865,878,52,56,
814,875,60,781,64,812,68,866,72,813,76,80,
784,84,88,92,877,96,100,104,870,871,879,890,
108,112,116,120,124,128,132,136,140,144,148,152,
156,160,164,168,172,176,180,184,188,192,196,200,
204,208,212,216,220,224,228,232,236,240,244,248,
252,256,260,264,268,272,276,280,284,288,292,296,
300,304,308,312,316,320,324,328,332,336,340,344,
348,352,356,360,364,368,372,376,380,384,388,392,
396,400,404,408,412,416,420,424,428,432,436,440,
444,448,452,456,460,464,468,472,476,480,484,488,
492,496,500,504,768,769,770,771,772,773,774,775,
776,777,778,779,780,782,783,785,786,787,788,789,
790,791,792,793,794,795,796,797,798,799,801,802,
803,804,805,806,807,808,809,810,811,815,816,817,
818,819,820,821,822,823,824,825,826,827,828,829,

830,831,832,833,834,835,836,837,838,839,840,841,
842,843,844,845,846,847,848,849,850,851,852,853,
854,855,856,857,858,859,860,861,862,863,864,867,
868,872,874,880,881,886,887,888,889,891,892,893,
894,895,
4,8,12,800,16,20,24,873,28,865,32,869,
877,36,40,44,48,880,52,885,884,56,781,783,
883,60,64,68,72,76,879,813,80,84,88,866,
784,876,92,870,96,791,812,854,868,886,100,855,
871,104,108,814,826,112,116,120,124,128,132,136,
140,144,148,152,156,160,164,168,172,176,180,184,
188,192,196,200,204,208,212,216,220,224,228,232,
236,240,244,248,252,256,260,264,268,272,276,280,
284,288,292,296,300,304,308,312,316,320,324,328,
332,336,340,344,348,352,356,360,364,368,372,376,
380,384,388,392,396,400,404,408,412,416,420,424,
428,432,436,440,444,448,452,456,460,464,468,472,
476,480,484,488,492,496,500,504,768,769,770,771,
772,773,774,775,776,777,778,779,780,782,785,786,
787,788,789,790,792,793,794,795,796,797,798,799,
801,802,803,804,805,806,807,808,809,810,811,815,
816,817,818,819,820,821,822,823,824,825,827,828,
829,830,831,832,833,834,835,836,837,838,839,840,
841,842,843,844,845,846,847,848,849,850,851,852,
853,856,857,858,859,860,861,862,863,864,867,872,
874,875,878,881,882,887,888,889,890,891,892,893,
894,895,
4,8,12,16,20,878,24,28,882,32,36,40,
877,44,868,876,871,48,867,52,56,884,60,64,
883,800,68,887,890,870,72,872,880,76,80,866,
813,84,88,92,888,96,869,100,791,104,835,875,
886,108,112,781,885,116,120,812,865,879,124,128,
132,136,140,144,148,152,156,160,164,168,172,176,
180,184,188,192,196,200,204,208,212,216,220,224,
228,232,236,240,244,248,252,256,260,264,268,272,
276,280,284,288,292,296,300,304,308,312,316,320,
324,328,332,336,340,344,348,352,356,360,364,368,
372,376,380,384,388,392,396,400,404,408,412,416,
420,424,428,432,436,440,444,448,452,456,460,464,
468,472,476,480,484,488,492,496,500,504,768,769,
770,771,772,773,774,775,776,777,778,779,780,782,
783,784,785,786,787,788,789,790,792,793,794,795,
796,797,798,799,801,802,803,804,805,806,807,808,
809,810,811,814,815,816,817,818,819,820,821,822,
823,824,825,826,827,828,829,830,831,832,833,834,
836,837,838,839,840,841,842,843,844,845,846,847,
848,849,850,851,852,853,854,855,856,857,858,859,
860,861,862,863,864,873,874,881,889,891,892,893,
894,895,
872,4,8,875,12,16,869,20,879,24,28,865,
32,884,36,781,876,40,44,877,48,52,814,889,

56,60,64,68,72,76,80,84,88,92,96,100,
104,108,112,116,120,124,128,132,136,140,144,148,
152,156,160,164,168,172,176,180,184,188,192,196,
200,204,208,212,216,220,224,228,232,236,240,244,
248,252,256,260,264,268,272,276,280,284,288,292,
296,300,304,308,312,316,320,324,328,332,336,340,
344,348,352,356,360,364,368,372,376,380,384,388,
392,396,400,404,408,412,416,420,424,428,432,436,
440,444,448,452,456,460,464,468,472,476,480,484,
488,492,496,500,504,768,769,770,771,772,773,774,
775,776,777,778,779,780,782,783,784,785,786,787,
788,789,790,791,792,793,794,795,796,797,798,799,
800,801,802,803,804,805,806,807,808,809,810,811,
812,813,815,816,817,818,819,820,821,822,823,824,
825,826,827,828,829,830,831,832,833,834,835,836,
837,838,839,840,841,842,843,844,845,846,847,848,
849,850,851,852,853,854,855,856,857,858,859,860,
861,862,863,864,866,867,868,870,871,873,874,878,
880,881,882,883,885,886,887,888,890,891,892,893,
894,895,

// english markov table follows

4,8,12,16,800,20,24,28,32,884,36,40,
44,48,865,52,56,60,64,879,68,873,72,883,
867,76,80,84,866,88,887,92,96,880,877,100,
868,104,108,870,781,112,882,116,869,120,124,876,
128,848,878,872,885,132,833,136,140,144,852,148,
152,156,160,164,836,817,840,168,172,838,176,180,
851,184,808,850,886,188,835,871,192,841,845,196,
200,204,818,208,834,212,807,216,220,224,821,839,
875,228,837,874,881,232,236,240,854,244,824,843,
889,248,252,819,846,849,256,820,855,822,825,844,
847,853,260,264,268,272,276,842,280,813,284,288,
292,296,300,304,308,312,816,858,890,316,320,324,
328,332,336,340,344,348,352,356,360,364,368,372,
376,810,811,380,384,388,392,396,400,404,408,412,
416,420,424,428,432,436,440,444,448,452,456,460,
464,468,472,476,480,484,488,492,496,500,504,768,
769,770,771,772,773,774,775,776,777,778,779,780,
782,783,784,785,786,787,788,789,790,791,792,793,
794,795,796,797,798,799,801,802,803,804,805,806,
809,812,814,815,823,826,827,828,829,830,831,832,
856,857,859,860,861,862,863,864,888,891,892,893,
894,895,
4,8,800,12,16,20,24,882,28,32,883,36,
878,40,868,44,48,52,56,865,60,867,877,64,
884,68,876,72,76,886,80,869,84,88,881,814,
92,781,870,812,888,96,813,873,889,880,887,100,
104,108,885,112,116,120,124,128,871,132,136,140,
866,872,879,144,807,148,152,156,160,164,168,809,

826,172,176,180,184,188,192,196,200,204,208,212,
216,831,220,224,228,232,236,240,244,248,252,256,
260,264,268,272,276,280,284,288,292,296,300,304,
308,312,316,320,324,328,332,336,340,344,348,352,
356,360,364,368,372,376,380,384,388,392,396,400,
404,408,412,416,420,424,428,432,436,440,444,448,
452,456,460,464,468,472,476,480,484,488,492,496,
500,504,768,769,770,771,772,773,774,775,776,777,
778,779,780,782,783,784,785,786,787,788,789,790,
791,792,793,794,795,796,797,798,799,801,802,803,
804,805,806,808,810,811,815,816,817,818,819,820,
821,822,823,824,825,827,828,829,830,832,833,834,
835,836,837,838,839,840,841,842,843,844,845,846,
847,848,849,850,851,852,853,854,855,856,857,858,
859,860,861,862,863,864,874,875,890,891,892,893,
894,895,
4,8,12,16,800,20,24,28,32,868,871,36,
884,40,865,869,883,44,867,48,879,52,56,60,
64,876,68,72,873,870,76,889,813,878,781,80,
812,885,886,875,814,84,88,92,96,100,104,108,
112,116,120,124,128,132,880,136,140,144,148,152,
156,160,164,168,172,176,180,184,188,192,196,200,
204,208,212,216,220,224,228,232,236,240,244,248,
252,256,260,264,268,272,276,280,284,288,292,296,
300,304,308,312,316,320,324,328,332,336,340,344,
348,352,356,360,364,368,372,376,380,384,388,392,
396,400,404,408,412,416,420,424,428,432,436,440,
444,448,452,456,460,464,468,472,476,480,484,488,
492,496,500,504,768,769,770,771,772,773,774,775,
776,777,778,779,780,782,783,784,785,786,787,788,
789,790,791,792,793,794,795,796,797,798,799,801,
802,803,804,805,806,807,808,809,810,811,815,816,
817,818,819,820,821,822,823,824,825,826,827,828,
829,830,831,832,833,834,835,836,837,838,839,840,
841,842,843,844,845,846,847,848,849,850,851,852,
853,854,855,856,857,858,859,860,861,862,863,864,
866,872,874,877,881,882,887,888,890,891,892,893,
894,895,
4,8,12,16,878,20,24,884,28,883,32,879,
36,40,44,871,48,867,877,52,868,876,870,56,
882,869,60,886,64,865,68,866,880,72,813,76,
875,80,84,890,88,92,814,96,100,104,888,108,
112,116,120,124,128,132,872,136,140,144,148,152,
156,160,164,168,172,176,180,184,188,192,196,200,
204,208,212,216,220,224,228,232,236,240,244,248,
252,256,260,264,268,272,276,280,284,288,292,296,
300,304,308,312,316,320,324,328,332,336,340,344,
348,352,356,360,364,368,372,376,380,384,388,392,
396,400,404,408,412,416,420,424,428,432,436,440,
444,448,452,456,460,464,468,472,476,480,484,488,
492,496,500,504,768,769,770,771,772,773,774,775,

776,777,778,779,780,781,782,783,784,785,786,787,
788,789,790,791,792,793,794,795,796,797,798,799,
800,801,802,803,804,805,806,807,808,809,810,811,
812,815,816,817,818,819,820,821,822,823,824,825,
826,827,828,829,830,831,832,833,834,835,836,837,
838,839,840,841,842,843,844,845,846,847,848,849,
850,851,852,853,854,855,856,857,858,859,860,861,
862,863,864,873,874,881,885,887,889,891,892,893,
894,895,
4,8,12,869,16,20,800,24,28,879,865,32,
36,40,44,48,884,52,873,868,889,877,882,56,
883,60,64,68,72,812,76,870,80,814,885,84,
88,876,781,875,92,96,866,100,813,878,104,108,
867,886,887,112,871,116,120,124,880,128,132,136,
140,144,148,807,152,156,160,164,168,172,176,180,
184,188,192,196,200,204,208,212,216,220,224,228,
232,236,240,244,248,252,256,260,264,268,272,276,
280,284,288,292,296,300,304,308,312,316,320,324,
328,332,336,340,344,348,352,356,360,364,368,372,
376,380,384,388,392,396,400,404,408,412,416,420,
424,428,432,436,440,444,448,452,456,460,464,468,
472,476,480,484,488,492,496,500,504,768,769,770,
771,772,773,774,775,776,777,778,779,780,782,783,
784,785,786,787,788,789,790,791,792,793,794,795,
796,797,798,799,801,802,803,804,805,806,808,809,
810,811,815,816,817,818,819,820,821,822,823,824,
825,826,827,828,829,830,831,832,833,834,835,836,
837,838,839,840,841,842,843,844,845,846,847,848,
849,850,851,852,853,854,855,856,857,858,859,860,
861,862,863,864,872,874,881,888,890,891,892,893,
894,895,
4,8,872,12,16,800,20,873,24,869,28,879,
32,36,865,882,40,883,44,48,52,885,56,887,
889,60,64,812,68,72,867,884,76,814,781,80,
878,813,876,84,88,92,96,100,104,108,112,116,
120,124,128,132,136,140,815,868,877,880,890,144,
148,152,156,160,164,168,172,176,180,184,188,192,
196,200,204,208,212,216,220,224,228,232,236,240,
244,248,252,256,260,264,268,272,276,280,284,288,
292,296,300,304,308,312,316,320,324,328,332,336,
340,344,348,352,356,360,364,368,372,376,380,384,
388,392,396,400,404,408,412,416,420,424,428,432,
436,440,444,448,452,456,460,464,468,472,476,480,
484,488,492,496,500,504,768,769,770,771,772,773,
774,775,776,777,778,779,780,782,783,784,785,786,
787,788,789,790,791,792,793,794,795,796,797,798,
799,801,802,803,804,805,806,807,808,809,810,811,
816,817,818,819,820,821,822,823,824,825,826,827,
828,829,830,831,832,833,834,835,836,837,838,839,
840,841,842,843,844,845,846,847,848,849,850,851,
852,853,854,855,856,857,858,859,860,861,862,863,

864,866,870,871,874,875,881,886,888,891,892,893,
894,895,
4,8,800,12,16,20,24,28,884,873,869,32,
36,40,44,48,52,883,872,56,812,879,889,880,
60,885,814,64,68,781,72,865,877,76,867,870,
80,813,84,887,88,876,92,807,96,815,100,104,
108,809,112,116,120,124,128,132,136,140,144,148,
152,156,160,164,168,172,176,180,184,188,192,196,
200,204,208,212,216,220,224,228,232,236,240,244,
248,252,256,260,264,268,272,276,280,284,288,292,
296,300,304,308,312,316,320,324,328,332,336,340,
344,348,352,356,360,364,368,372,376,380,384,388,
392,396,400,404,408,412,416,420,424,428,432,436,
440,444,448,452,456,460,464,468,472,476,480,484,
488,492,496,500,504,768,769,770,771,772,773,774,
775,776,777,778,779,780,782,783,784,785,786,787,
788,789,790,791,792,793,794,795,796,797,798,799,
801,802,803,804,805,806,808,810,811,816,817,818,
819,820,821,822,823,824,825,826,827,828,829,830,
831,832,833,834,835,836,837,838,839,840,841,842,
843,844,845,846,847,848,849,850,851,852,853,854,
855,856,857,858,859,860,861,862,863,864,866,868,
871,874,875,878,881,882,886,888,890,891,892,893,
894,895,
4,8,800,12,869,16,873,20,24,28,32,36,
885,40,44,48,865,52,883,56,879,60,887,812,
814,64,68,884,889,868,72,781,834,76,813,882,
80,886,84,88,92,876,96,100,826,874,104,108,
112,116,809,871,872,120,124,128,132,136,140,144,
148,152,156,160,164,168,172,176,180,184,188,192,
196,200,204,208,212,216,220,224,228,232,236,240,
244,248,252,256,260,264,268,272,276,280,284,288,
292,296,300,304,308,312,316,320,324,328,332,336,
340,344,348,352,356,360,364,368,372,376,380,384,
388,392,396,400,404,408,412,416,420,424,428,432,
436,440,444,448,452,456,460,464,468,472,476,480,
484,488,492,496,500,504,768,769,770,771,772,773,
774,775,776,777,778,779,780,782,783,784,785,786,
787,788,789,790,791,792,793,794,795,796,797,798,
799,801,802,803,804,805,806,807,808,810,811,815,
816,817,818,819,820,821,822,823,824,825,827,828,
829,830,831,832,833,835,836,837,838,839,840,841,
842,843,844,845,846,847,848,849,850,851,852,853,
854,855,856,857,858,859,860,861,862,863,864,866,
867,870,875,877,878,880,881,888,890,891,892,893,
894,895,
4,8,12,16,20,878,24,28,876,884,32,882,
36,800,40,883,44,48,52,867,868,56,60,873,
886,866,64,877,880,885,68,871,889,72,76,875,
80,781,888,84,813,88,92,96,870,100,104,108,
112,874,116,120,124,128,132,136,140,812,887,144,

148,152,156,160,164,168,172,176,180,184,188,192,
196,200,204,208,212,216,220,224,228,232,236,240,
244,248,252,256,260,264,268,272,276,280,284,288,
292,296,300,304,308,312,316,320,324,328,332,336,
340,344,348,352,356,360,364,368,372,376,380,384,
388,392,396,400,404,408,412,416,420,424,428,432,
436,440,444,448,452,456,460,464,468,472,476,480,
484,488,492,496,500,504,768,769,770,771,772,773,
774,775,776,777,778,779,780,782,783,784,785,786,
787,788,789,790,791,792,793,794,795,796,797,798,
799,801,802,803,804,805,806,807,808,809,810,811,
814,815,816,817,818,819,820,821,822,823,824,825,
826,827,828,829,830,831,832,833,834,835,836,837,
838,839,840,841,842,843,844,845,846,847,848,849,
850,851,852,853,854,855,856,857,858,859,860,861,
862,863,864,865,869,872,879,881,890,891,892,893,
894,895,
4,8,12,16,20,24,883,28,876,32,884,36,
40,882,877,44,873,878,48,868,869,867,871,52,
865,880,56,60,870,64,800,866,813,68,72,76,
80,84,88,92,96,100,104,108,112,116,120,124,
128,132,136,140,144,148,152,156,160,164,168,172,
176,180,184,188,192,196,200,204,208,212,216,220,
224,228,232,236,240,244,248,252,256,260,264,268,
272,276,280,284,288,292,296,300,304,308,312,316,
320,324,328,332,336,340,344,348,352,356,360,364,
368,372,376,380,384,388,392,396,400,404,408,412,
416,420,424,428,432,436,440,444,448,452,456,460,
464,468,472,476,480,484,488,492,496,500,504,768,
769,770,771,772,773,774,775,776,777,778,779,780,
781,782,783,784,785,786,787,788,789,790,791,792,
793,794,795,796,797,798,799,801,802,803,804,805,
806,807,808,809,810,811,812,814,815,816,817,818,
819,820,821,822,823,824,825,826,827,828,829,830,
831,832,833,834,835,836,837,838,839,840,841,842,
843,844,845,846,847,848,849,850,851,852,853,854,
855,856,857,858,859,860,861,862,863,864,872,874,
875,879,881,885,886,887,888,889,890,891,892,893,
894,895,
4,8,12,16,20,800,24,869,28,32,876,889,
873,36,879,865,40,883,44,48,885,52,56,884,
60,64,868,68,812,781,882,72,814,76,813,80,
84,88,866,807,887,92,870,871,877,96,100,104,
108,112,116,867,886,120,124,128,132,136,140,144,
148,152,156,160,164,168,172,176,180,184,188,192,
196,200,204,208,212,216,220,224,228,232,236,240,
244,248,252,256,260,264,268,272,276,280,284,288,
292,296,300,304,308,312,316,320,324,328,332,336,
340,344,348,352,356,360,364,368,372,376,380,384,
388,392,396,400,404,408,412,416,420,424,428,432,
436,440,444,448,452,456,460,464,468,472,476,480,

484,488,492,496,500,504,768,769,770,771,772,773,
774,775,776,777,778,779,780,782,783,784,785,786,
787,788,789,790,791,792,793,794,795,796,797,798,
799,801,802,803,804,805,806,808,809,810,811,815,
816,817,818,819,820,821,822,823,824,825,826,827,
828,829,830,831,832,833,834,835,836,837,838,839,
840,841,842,843,844,845,846,847,848,849,850,851,
852,853,854,855,856,857,858,859,860,861,862,863,
864,872,874,875,878,880,881,888,890,891,892,893,
894,895,
4,869,8,12,16,865,800,873,20,879,24,28,
32,882,36,880,885,40,44,48,52,56,60,781,
814,64,812,68,877,878,884,72,813,883,76,80,
84,88,92,867,889,96,100,104,108,112,116,120,
124,128,132,136,140,144,148,152,156,160,164,168,
172,176,180,184,188,192,196,200,204,208,212,216,
220,224,228,232,236,240,244,248,252,256,260,264,
268,272,276,280,284,288,292,296,300,304,308,312,
316,320,324,328,332,336,340,344,348,352,356,360,
364,368,372,376,380,384,388,392,396,400,404,408,
412,416,420,424,428,432,436,440,444,448,452,456,
460,464,468,472,476,480,484,488,492,496,500,504,
768,769,770,771,772,773,774,775,776,777,778,779,
780,782,783,784,785,786,787,788,789,790,791,792,
793,794,795,796,797,798,799,801,802,803,804,805,
806,807,808,809,810,811,815,816,817,818,819,820,
821,822,823,824,825,826,827,828,829,830,831,832,
833,834,835,836,837,838,839,840,841,842,843,844,
845,846,847,848,849,850,851,852,853,854,855,856,
857,858,859,860,861,862,863,864,866,868,870,871,
872,874,875,876,881,886,887,888,890,891,892,893,
894,895,
4,8,800,12,16,20,24,878,872,28,869,32,
873,36,865,40,44,885,48,52,781,814,879,884,
882,56,60,876,64,812,871,889,68,72,76,80,
807,813,883,84,88,92,96,100,104,108,112,116,
120,124,128,132,136,140,144,148,152,156,160,164,
168,172,176,180,184,188,192,196,200,204,208,212,
216,220,224,228,232,236,240,244,248,252,256,260,
264,268,272,276,280,284,288,292,296,300,304,308,
312,316,320,324,328,332,336,340,344,348,352,356,
360,364,368,372,376,380,384,388,392,396,400,404,
408,412,416,420,424,428,432,436,440,444,448,452,
456,460,464,468,472,476,480,484,488,492,496,500,
504,768,769,770,771,772,773,774,775,776,777,778,
779,780,782,783,784,785,786,787,788,789,790,791,
792,793,794,795,796,797,798,799,801,802,803,804,
805,806,808,809,810,811,815,816,817,818,819,820,
821,822,823,824,825,826,827,828,829,830,831,832,
833,834,835,836,837,838,839,840,841,842,843,844,
845,846,847,848,849,850,851,852,853,854,855,856,

857,858,859,860,861,862,863,864,866,867,868,870,
874,875,877,880,881,886,887,888,890,891,892,893,
894,895,
4,8,12,16,20,865,869,879,24,28,880,800,
32,873,36,885,866,883,40,44,814,877,48,52,
56,60,812,64,68,878,887,72,813,870,76,80,
781,854,84,88,92,96,100,104,108,112,116,120,
124,128,132,136,140,144,148,152,156,160,164,168,
172,176,180,184,188,192,196,200,204,208,212,216,
220,224,228,232,236,240,244,248,252,256,260,264,
268,272,276,280,284,288,292,296,300,304,308,312,
316,320,324,328,332,336,340,344,348,352,356,360,
364,368,372,376,380,384,388,392,396,400,404,408,
412,416,420,424,428,432,436,440,444,448,452,456,
460,464,468,472,476,480,484,488,492,496,500,504,
768,769,770,771,772,773,774,775,776,777,778,779,
780,782,783,784,785,786,787,788,789,790,791,792,
793,794,795,796,797,798,799,801,802,803,804,805,
806,807,808,809,810,811,815,816,817,818,819,820,
821,822,823,824,825,826,827,828,829,830,831,832,
833,834,835,836,837,838,839,840,841,842,843,844,
845,846,847,848,849,850,851,852,853,855,856,857,
858,859,860,861,862,863,864,867,868,871,872,874,
875,876,881,882,884,886,888,889,890,891,892,893,
894,895,
4,8,12,16,878,20,24,882,28,32,870,800,
36,40,868,44,48,52,877,885,56,887,60,884,
876,64,880,886,883,68,867,72,866,76,80,84,
88,871,869,879,92,813,96,873,888,100,104,108,
814,865,112,812,116,781,889,120,124,128,875,132,
136,140,144,148,152,872,156,160,164,168,172,176,
180,184,188,192,196,200,204,208,212,216,220,224,
228,232,236,240,244,248,252,256,260,264,268,272,
276,280,284,288,292,296,300,304,308,312,316,320,
324,328,332,336,340,344,348,352,356,360,364,368,
372,376,380,384,388,392,396,400,404,408,412,416,
420,424,428,432,436,440,444,448,452,456,460,464,
468,472,476,480,484,488,492,496,500,504,768,769,
770,771,772,773,774,775,776,777,778,779,780,782,
783,784,785,786,787,788,789,790,791,792,793,794,
795,796,797,798,799,801,802,803,804,805,806,807,
808,809,810,811,815,816,817,818,819,820,821,822,
823,824,825,826,827,828,829,830,831,832,833,834,
835,836,837,838,839,840,841,842,843,844,845,846,
847,848,849,850,851,852,853,854,855,856,857,858,
859,860,861,862,863,864,874,881,890,891,892,893,
894,895,
4,8,12,879,16,20,24,872,865,869,28,884,
32,36,40,44,873,875,48,882,889,52,876,885,
800,867,56,781,814,60,64,68,813,72,76,80,
84,88,92,96,100,104,108,112,116,120,124,128,

```
            132,136,140,144,148,152,156,160,164,168,172,176,
            180,184,188,192,196,200,204,208,212,216,220,224,
            228,232,236,240,244,248,252,256,260,264,268,272,
            276,280,284,288,292,296,300,304,308,312,316,320,
            324,328,332,336,340,344,348,352,356,360,364,368,
            372,376,380,384,388,392,396,400,404,408,412,416,
            420,424,428,432,436,440,444,448,452,456,460,464,
            468,472,476,480,484,488,492,496,500,504,768,769,
            770,771,772,773,774,775,776,777,778,779,780,782,
            783,784,785,786,787,788,789,790,791,792,793,794,
            795,796,797,798,799,801,802,803,804,805,806,807,
            808,809,810,811,812,815,816,817,818,819,820,821,
            822,823,824,825,826,827,828,829,830,831,832,833,
            834,835,836,837,838,839,840,841,842,843,844,845,
            846,847,848,849,850,851,852,853,854,855,856,857,
            858,859,860,861,862,863,864,866,868,870,871,874,
            877,878,880,881,883,886,887,888,890,891,892,893,
            894,895
            };

unsigned short reftab[]=
                {
            0,0,0,0,0,0,0,0,0,0,0,// 0-10
            0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,0,0,0,0,
            0,508,0,0,0,0,0,0,0,0,                      // 31-40
            0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,4572,0,8128,4064,               // 91-100
            1016,0,6604,6096,2032,0,0,5588,7112,1524,   // 101-110
            7620,0,0,2540,3556,3048,5080,0,0,0,         // 111-120
            0,0,0,0,0,0,0                               //121-127
            };
```

SCS
Spezielle Communications Systeme GmbH & Co. KG
Röntgenstraße 36
63454 Hanau
GERMANY

Internet: www.p4dragon.com
E-Mail: info@p4dragon.com

Tel.: +49 (0)6181 85 00 00
Fax.: +49 (0)6181 99 02 38