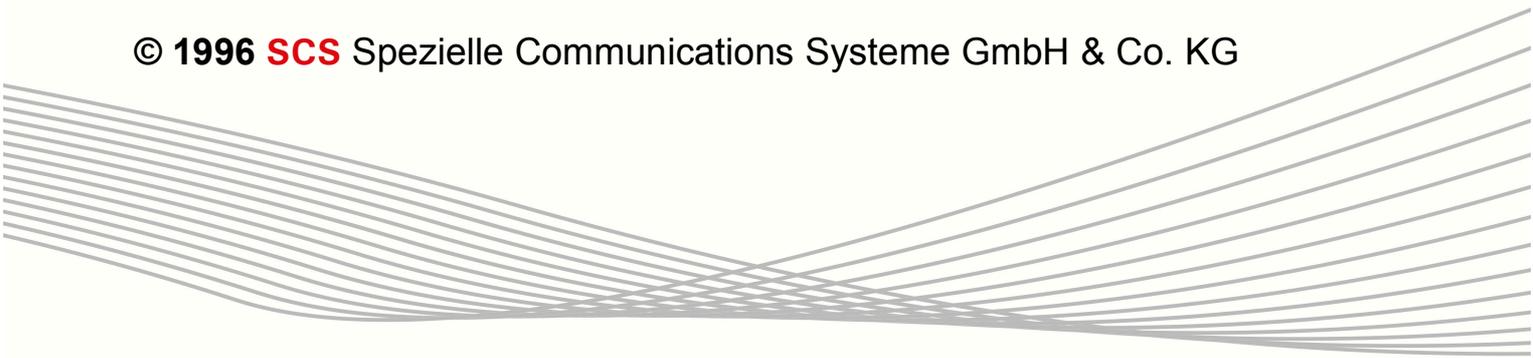




The PACTOR-2 Protocol

A Technical Description

© 1996 **SCS** Spezielle Communications Systeme GmbH & Co. KG



PACTOR-2

Protocol Details

1. Introduction

All new modes should provide significant improvements over existing systems, which must not only refer to the maximum throughput and the robustness. Other basic attributes, like signal bandwidth, required frequency accuracy and compatibility to existing standards also have to be taken into consideration.

Modulation and encoding methods that supply high throughput rates, e.g. 16-DPSK, normally suffer from a lack of robustness. On the other hand, systems distinguished by a high robustness, e.g. DBPSK combined with a rate 1/2 convolutional code, only provide a low maximum throughput. Therefore, adaptive digital modes have to be used that apply different modulation and encoding methods depending on the channel quality. Just changing the symbol rate however, leads to only little adaptivity and additionally results in variations of the bandwidth. In order to prevent any spillover in adjacent channels, the bandwidth should ideally always remain the same, regardless of whether a robust or a fast data transmission is performed. As 500 Hz CW filters are very commonly used and due to the usual spacing of the mailbox frequencies on short wave, a maximum bandwidth of 500 Hz should not be exceeded. In addition, there should not be too high a demand placed on the transceiver used, regarding its frequency adjustment and stability. For optimum results, maximum frequency deviations similar to FSK modes should be tolerated. This forces the use of powerful tracking methods, which allow a link also to be established if the deviation is up to +/- 80 Hz. Further, a new mode should be backwards compatible to an existing standard, preferably with automatic switching, in order to prevent a deficiency of QSO partners in the early stage.

PACTOR-2 meets all the above mentioned requirements. It is fully backwards compatible with the current PACTOR ("PACTOR-1") standard, as the initial link setup is still done in FSK. If both stations are capable of Level II, an automatic switching is performed. The PACTOR-2 protocol basically uses a two-tone DPSK system with raised cosine pulse shaping, which reduces the required bandwidth to less than 500 Hz. The maximum absolute transfer rate is 800 bits per second. Due to the improved on-line data compression, maximum effective throughput rates of more than 1200 bits per second can be obtained. PACTOR-2 is thus the fastest short wave digital mode. Very efficient error control coding using a Convolutional Code with a constraint length of 9 and a real Viterbi decoder with soft decision is applied at all speedlevels, in addition to analog Memory-ARQ. PACTOR-2 is also therefore, by far the most robust digital mode, which allows a link to be established and achieve a reasonable throughput in such poor propagation conditions that all other modes fail. In comparison with the current FSK PACTOR standard including analog Memory-ARQ, which had been the most robust digital mode until the release of PACTOR-2, a further gain of robustness of around 7 dB could be obtained. The following chapters describe details of the PACTOR-2 protocol.

2. Structure and Timing of the PACTOR-2 Packets

Similar to the current FSK PACTOR standard, PACTOR-2 is also a half-duplex synchronous ARQ system without any mark/space convention. It may thus be operated in USB as well as LSB position of the transceiver. The initial link setup is still performed using the FSK (PACTOR-1) protocol, in order to achieve compatibility to the previous level. If both stations are capable of PACTOR-2, an automatic switching to the higher level is performed. The basic PACTOR-2 frame structure is similar to PACTOR-1. It consists of a header, a variable data field, the status byte and the CRC. The standard cycle duration does not differ from FSK PACTOR and is still 1.25 seconds, which is one of the requirements to obtain easy compatibility to Level 1. Longer Control Signals (CS) had to be applied to achieve a higher robustness for the acknowledgment signal, required due

to the greater robustness of the data channel. The entire length of the standard packet had to be shortened to 0.8 seconds in order not to shorten the maximum possible propagation delay, which is thus still 170 milliseconds. The requirements to operate PACTOR-2 regarding the transmit delay and the receiver recovery time of the used equipment therefore remain unchanged in comparison with Level I.

Due to the signal propagation delay, and equipment switching delays, PACTOR-2 as well as PACTOR-1 has in the standard mode a maximum range for ARQ contacts of around 20,000 km. As with PACTOR-1, a long path option is available also for PACTOR-2, enabling contacts up to 40,000 km. The sending station calls the partner station in 'Long Path Mode'. Initial contact is established using the PACTOR-1 FSK protocol as previously mentioned, but with a cycle time of 1.4 seconds instead of 1.25. This longer cycle time allows for the much greater propagation delays found on 'Long Path' contacts. The link then automatically switches to PACTOR-2, with the same cycle duration. In the new data mode (see below), timing is also automatically adjusted to obtain longer receiving gaps.

Unlike the previous level, PACTOR-2 additionally switches to longer packets if the data blocks are not filled up with idles, (i.e. if the transmitter buffer indicates that more information has to be transferred than fitting into the standard packets). If the information sending station (ISS) prefers to use long packets, it sets the long cycle flag in the status word. The information receiving station (IRS) then finally can accept the proposed change of the cycle duration by sending a CS6. This situation, for example, occurs when reading longer files out of mailboxes. The long packets are basically made up like the short ones, but consist of a larger data field, which may contain up to 2208 bits of usable information. The length of these data packets is 3.28 seconds, which leads to an entire cycle duration of 3.75 seconds in this so-called data mode. Figure 1 shows the PACTOR-2 cycle duration and the packet construction in the standard mode as well as in the data mode.

When entering text manually in QSO traffic from operator to operator, the maximum throughput of the standard mode is normally not used up, thus the required higher flexibility of the system is still available due to the short packets. The efficient use of longer data packets on short wave is generally only possible, if powerful error control coding, with full frame interleaving is applied to cancel out error bursts or short fading periods. As already mentioned, PACTOR-2 uses a convolutional code with a constraint length of 9, a real Viterbi decoder and soft decision, in addition to analog Memory-ARQ. Due to the high coding gain and the resulting capacity of error correction without requesting a repetition of the entire packet, a significant increase in the effective throughput could be obtained. Proceeding from average bit error rates on short wave channels, simple block codes are usually unable to provide enough coding gain. This often leads to a decrease in speed when using longer data strings, as repetitions often cannot be avoided. For more details on these technical foundations, see the first part of this series.

PACTOR-2 uses six different CS, each consisting of 40 bits, all having exactly the maximum possible mutual hamming distance of 24 bits to each other. They thus reach exactly the Plotkin boundary and also represent a perfect code. This allows the advantageous use of the Cross Correlation method for decoding, which is also a kind of soft decision, leading to the correct detection of even inaudible CS. This checking is not only confined to a simple binary principle. A complex analog test procedure is applied, using the fine detail data from the DSP, to evaluate the single CS received, as well as the information summed up in the Memory-ARQ buffer. Similar to Level I, CS1 and CS2 are used to acknowledge/request packets and CS3 forces a break-in. CS4 and CS5 handle the speed changes, and CS6 is a toggle for the packet length. All CS are always sent in DBPSK in order to obtain a maximum of robustness.

3. Speedlevels and Error Control Coding

As mentioned in the introduction, PACTOR-2 uses a two-tone DPSK modulation system. Due to the raised cosine pulse shaping, the maximum required bandwidth is only around 450 Hz at minus 50 dB. ASK, which was also tested in the early stage, provided poorer results in weak conditions compared with a higher DPSK modulation, as different amplitude levels are more difficult to distinguish in noisy channels than more phase levels. Additionally, ASK increases the Crest Factor of the signal. For these reasons, it is not used in the final PACTOR-2 protocol. Basic information on these items can also be found in the first part of this series.

PACTOR-2 uses instead, different DPSK modulation schemes and various code rates. The Crest Factor of the PACTOR-2 signal is therefore only 1.45. The basic code used is an optimum rate 1/2 convolutional code with a constraint length of 9. Codes with higher rates, e.g. rate 2/3 and rate 7/8,

can be derived from that code by so-called puncturing. Prior to the transmission, certain of the symbols of the rate 1/2 encoded stream are 'punctured' or deleted, and not transmitted. At the receiving end, the punctured encoded bits are replaced with 'null' symbols prior to decoding with the rate 1/2 decoder. The decoder treats these null symbols neither as a received '1' nor as '0', but as an exactly intermediate value. No information is thus conveyed by that symbol that may influence the decoding process. The coding performance of 'punctured' code operation nearly matches the coding performance of the best known classic rate 2/3 or 7/8 codes with a comparable constraint length, provided that the puncture pattern is chosen carefully. The major advantage of this approach is that a single code rate decoder (in our case a rate 1/2 decoder) can implement a wide range of codes.

In the PTC-II, the Viterbi algorithm is implemented for decoding of the convolutional code. Nevertheless, as already indicated in the first part, there are several different methods to decode the PACTOR-2 signal, which require less processing power, but in return for this, also provide less coding gain. However, these methods at least allow compatibility to the PACTOR-2 standard and may therefore be applied in cheaper hardware.

The most robust PACTOR-2 speed level employs DBPSK with rate 1/2 coding, which per cycle allows an absolute throughput of 5 bytes in the standard mode and 36 bytes in the data mode respectively. In the next step, DQPSK with rate 1/2 coding is applied, which leads to an absolute throughput of 14 bytes in the standard mode and 76 bytes in the data mode. This is followed by 8-DPSK with a rate 2/3 coding, providing a throughput of 32 and 156 bytes per packet, respectively. Finally, in best propagation conditions, PACTOR-2 applies 16-DPSK with a rate 7/8 coding, which allows the maximum throughput of 59 bytes in a short packet and 276 bytes in the data mode. The mentioned transfer rates are all net rates referring to 8-bit ASCII, which are calculated after the error control coding and all other protocol overhead. As data compression is usually active, these throughput rates must be multiplied by the compression factor. The effective speed is therefore considerably higher in practice. All throughput rates and the corresponding modulation and encoding methods are summarized in table 1. The speed levels are automatically chosen by the PTC-II, considering the link statistics and the actual channel quality, thus no user intervention is required.

4. Online Data Compression

Like in the previous FSK PACTOR system, automatic on-line Huffman data compression is applied. Additionally, PACTOR-2 uses run-length encoding and, as a further novelty, Pseudo-Markov Compression (PMC, see below). Compared to 8-bit ASCII (plain text) PMC yields a compression factor of around 1.9, which leads to an effective speed of about 600 bits per second in average propagation conditions in data mode. PACTOR-2 is already around 3 times faster than PACTOR-1 and 15 times faster than AMTOR on average channels. However, the maximum effective speed in good conditions can exceed 1200 bits per second. As the PTC-II firmware automatically checks, whether PMC, Huffman encoding or the original ASCII code is the best choice, which depends on the probability of occurrence of the characters, there is no risk of losing throughput capacity. PACTOR-2 is of course still able to transfer any given binary information, e.g. programs or picture- and voice files. In these cases the on-line data compression is automatically switched off.

Ordinary Huffman compression exploits the 'one-dimensional' probability distribution of the characters in plain texts. The more frequently a character occurs, the shorter has to be the Huffman symbol that is assigned to the actual character. On the other hand, 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 sharper 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 also 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, because it can be considered as a hybrid between Markov- and Huffman encoding.

In this variant, 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 protocol and automatically chosen by the PTC-II.

5. Some Practical Aspects

Similar to Level I, the tones of the PACTOR-2 signal are spaced at 200 Hz. Their frequency may be defined freely in steps of 1 Hz by software command, as long as the shift remains 200 Hz. Thus you can easily switch between high- and low-tones, and also adjust any additionally required tone pair. This allows the utilizing of narrow CW filters in all transceivers that provide the option of activating the corresponding filters in the SSB mode.

In the PACTOR-2 system, the transferred information is swapped from one channel (tone) to the other in every cycle. Unlike FSK systems, the link is thus not blocked when strong narrow band QRM completely overpowers one channel (e.g. CW or carriers), but only its maximum speed is reduced. Usual FSK systems with a mark/space convention and without Memory-ARQ have to fail in such cases, because even if a so-called 'space-only' mode is applied, the strongest signal is automatically chosen. This always leads to a break-down of the link, as the QRM is stronger than the useful signal in the proposed case.

PACTOR-2 provides a comprehensive Listen-Mode, which is much more robust than known from PACTOR-I, because just the short header has to be received correctly, then the powerful error control coding can be fully utilized. Burst errors may be corrected also by monitoring stations and thus virtually do not affect the performance. The Unproto-Mode in PACTOR-2 allows to choose between all the above mentioned speed and encoding levels. On the receiving side, the correct mode is detected automatically and therefore needs no user-adjustment. For example, a fast and very robust QTC mode can thus be achieved, when a message is transmitted in the Unproto-Mode using DBPSK with rate 1/2 coding.

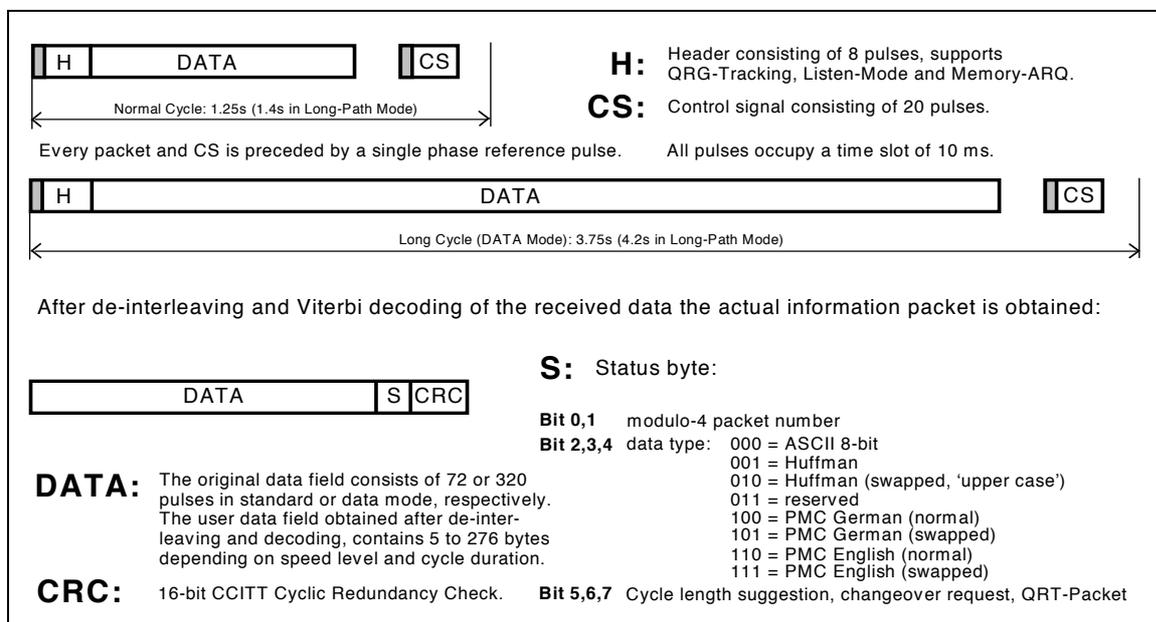


Figure 1: Structure and timing of the PACTOR-2 packets

Cycle	Modulation type	Code rate	Data bytes/packet (no compression)	Data bytes/sec (with PMC, f=1.9)
standard	DBPSK	1/2	5	7.6
data	DBPSK	1/2	36	18.2
standard	DQPSK	1/2	14	21.3
data	DQPSK	1/2	76	38.5
standard	8-DPSK	2/3	32	48.6
data	8-DPSK	2/3	156	79.0
standard	16-DPSK	7/8	59	89.7
data	16-DPSK	7/8	276	139.8

Table 1: PACTOR-2 speed and encoding levels with the resulting throughput rates (all throughput rates are net rates referring to 8-bit ASCII)

Annex I:

PACTOR-2 Channel Coding

PACTOR-2 always uses a Convolutional Code with $k=9$ and $R=1/2$ ein. The polynomials are

$G_1=111101011$;

$G_2=101110001$.

At higher code rates R , puncturing is applied.

	R	Puncturing Vector
SL1	1/2	1
SL2	1/2	1
SL3	2/3	1011
SL4	7/8	10100110101011

Annex II:

PACTOR-2 Bit Interleaving

Interleaving is generally performed as block bit interleaving. On the transmit side, the punctured / convolutional coded data set is sent to the permutation routine. The data set includes the status byte, the 2 CRC bytes as well as the 8 flushing bits from the Convolutional Encoder.

The permutation pointer P of the Interleaver is calculated as following:

```
S=1;           // Register S
P=0;           // initial permutation pointer
```

```
M=INTERLEAVER_DEPTH;
```

```
For (I=0; I<PACKET_SIZE; I++) // loop over the complete number of bits  
{
```

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

```
   P=P+M;
```

```
   if (P>PACKET_SIZE)
```

```
       P=S++;
```

```
}
```

PACKET_SIZE contains the number of bits of the packet that should be interleaved.
INTERLEAVER_DEPTH is the „interleaver depth”.

Usually PACKET_SIZE is divisible by INTERLEAVER_DEPTH.

(Generally: INTERLEAVER_DEPTH should be $\geq 2(k-1)$ of the constraint length of the Convolutional Code.)

PACTOR-2 utilizes 4 different interleaver depths:

SL1: 16

SL2: 8

SL3: 4

SL4: 2



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