Proceedings of the First International Workshop on

Digital Signal Processing
Techniques Applied to Space Communications

9–10 November 1988
ESTEC, Noordwijk, The Netherlands

european space agency / agence spatiale européenne
8-10, rue Mario-Nikis, 75738 PARIS CEDEX 15, France
Coded-8PSK Modem for Fixed and Mobile Satellite Services based on DSP

Peter Hoehner  Helmuth Lang

DFVLR
Institute of Communication Technology
Communication Theory Group
D-8031 Oberpfaffenhofen
West Germany

Abstract

The growing use of digital signal processing techniques is causing great changes in the structure of communication systems. We at DFVLR are currently developing a flexible test system that will allow different modulation and coding techniques to be quickly implemented and evaluated. The first step was to develop a modem that can produce Coded-8PSK as well as 4PSK signals. The 4PSK scheme is used as a reference. After a short introduction to 8PSK, the realised modem configuration and the algorithms employed are described. The last section provides an overview of planned future developments.

1 Introduction

Due to the development of modern Digital Signal Processing (DSP) chips, modulation and coding schemes that formerly could only be simulated offline can now be used in real systems. For example, DSP chips are now routinely used to adapt modem parameters in real time. To facilitate the testing and evaluation of different modulation schemes, DFVLR is developing a flexible system based on the digital signal processor TMS320C25 (Texas Instruments). This system includes special hardware and software modules as well as test equipment. Using this system, different modulation and coding techniques can be quickly implemented and evaluated.

2 C8PSK Modulation

The key point of Coded Modulation is the combination of channel coding and modulation. The encoded output symbols are mapped onto a signal space with increased cardinality. Proper code design results in a larger euclidean distance, compared to uncoded modulation, without changing the data rate, symbol rate, or the bandwidth. A functional blockdiagram of a Coded-8PSK (C8PSK) modem suitable for coherent detection on AWGN channels is shown in fig.4. We implemented a 4-state trellis-code as proposed by G. Ungerboeck [1]. Bits from the incoming data-stream are coded by a convolutional encoder. Three bits are output for every two bits from the data stream, see fig.5. Next, the coded bits are mapped into the 8PSK channel symbols. The convolutional encoder generates a discrete-time markov process with a one-to-one correspondence between the path-sequence and the channel symbols. Hence, the Viterbi-algorithm (VA) is the optimal decoder when the observations
are corrupted by memoryless noise [2]. The code is designed to maximize the euclidean distance between possible path sequences. The asymptotic coding gain is 3 dB compared to the uncoded 4PSK reference system.

3 System-Configuration

The experimental modem consists of a modulator, demodulator and the VAX-computer. The VAX-host is used for software development, simulations, and to calculate the necessary parameters. A mixed hardware-software configuration was chosen to make the system both flexible and easy to program.

All time critical functions and functions that handle data periodically, such as Input/Output for analog and digital signals, are realized in hardware. All functions that depend or branch on a decision, or that use tables or results of calculations are realized in software.

In the modulator hardware (fig.1), the digital input block collects the m-bit tuple for one symbol from the continuous data stream and stores it together with the Dynamic Control Word (DCW) in a First-In First-Out (FIFO) memory as the digital input word. The purpose of the DCW is to enable control functions for operational or testing purposes to be activated in synchronization with a certain information bit. Examples of that functions are: transmitter on/off, preamble insertion, trailing sequence insertion, signaling for FEC code switching. A Numerically Controlled Oscillator (NCO) driven by the master clock of the processor samples the incoming data. Its phase and frequency are set at the beginning, or when the data rate or the modulation technique are changed. The DSP-unit is therefore forced to fetch the FIFO-output at an arbitrary instance only once per symbol.

The output block parallelizes the consecutively computed k samples per symbol of the Inphase-(I) and Quadrature-(Q) components of the complex signal in a FIFO-memory with a size of 2 · 64 · 16 bits. The read operation of the two 12 bit wide k samples per symbol from the FIFO is also controlled by a NCO. This conversion into analog values results in a low quantization noise. After smoothing, the I- and Q-components modulate the carrier signal at 70 MHz in a vector modulator. A DCW is also generated which can be programmed by software to signal the eye center, transmitter on/off, or distortion on/off. This structure ensures autonomous and asynchronous operation without additional processor load.

The DSP-unit consists of the TMS320C25 processor, a status- and I/O-control block, a local memory, and the ports. In addition, specialized function-blocks can be attached to a certain processor-module (interleaver). Synchronization is done through memory-mapped status words. The Read- and Write-State-Words (RSW, WSW) allow functional control and operational checking of the hardware. The Static Control Word (SCW), which can be set by an external hex switch, enables an easy change between different program modules without interruption. So the status of the FIFOs is monitored through the RSW, and the NCOs are synchronized through the WSW. The setting of the NCO frequency is established through the bus. An additional memory-mapped 16 bit parallel digital I/O-port connects the processor to the host interface.

For investigations on the behaviour of control circuits, such as PLL lock in, a processor controlled noise source for adding distortions to the transmitted signal was included. To keep the processor load low a hardware Pseudo Random (PR) generator continuously supplies uniform and Gaussian distributed random variables to four memory mapped registers. The processor is able to restart the PR shift registers (sequence length up to $2^{32} - 1$ bits) so that this quasi-random process can be run repeatedly. The processor also sets masks to control the variance and accuracy. The crosscorrelations of the vari-
ables are in the same range as those of the software PR-generator available on the VAX mainframe. A Rayleigh distributed distortion can be generated by applying two Gaussian samples to the I- and Q-component of the transmitted signal.

At the input of the demodulator (fig.3) a coarse Automatic Gain Control (AGC) loop compensates for level changes to keep the vector demodulator at its correct operating point. The amplifier can be digitally controlled through the processor, and can be set to any value within its 65 dB dynamic range in less than 0.5µs. Its noise figure is always better than 1.7 dB. The power probe monitoring the output of the amplifier has a linear dB range of 30 dB, and supplies its digitized reading directly to the processor. All other elements of the control loop (filter, linearization, range) are realized in software. Currently we optimize their settings for different modulation schemes and for different channels.

The baseband I- and Q-components at the output of the vector demodulator are converted to 12 bit code words. Active lowpass filters in front of the A/D-converter prevent aliasing. The structure of the analog input, digital output and DSP-unit are very similar to those of the modulator. The clock for data output is derived either from the data input or totally autonomously; data transfer is buffered through FIFOs.

The monitoring unit acts as a quicklook facility. For example, the instantaneous value of the sampled input signal, the back-rotated values, and the smoothed error signals for phase and timing recovery can be displayed on an oscilloscope. It allows a good operational judgement of the demodulator status. If desired, all frequency sources (fig.2) in the processors, and the local oscillators of the vector-modulator and -demodulator can be fully synchronized with a stable reference source of the experimental earth terminal at DFVLR. In IF-loopback operation this allows a totally controllable setting of frequency- and phase-offset and/or their deterministic and/or stochastic distortions. In addition, in RF-loopback the influence of the TWTe-transmitter's backoff can be studied. Finally, transmitting over a real satellite channel the total synchronism of the processors and local oscillators also for up- and down-conversion allows an exact analysis of the real channel.

Two basic functions, operational and signal processing, are realized in software.

After a reset the ROM-resident operational software module initiates the booting of the necessary programs from the E²PROM-array of the global memory or from the host. Additionally, it tests the memory and the attached hardware. In the download program the DSP-modules are embedded in a software frame that keeps the necessary error modules, handles the parameters, sets the enclosed hardware, and loads the necessary tables either from the global PROM-area or from the host.

After that initialization the software runs the main loop in a start/stop mode triggered from the hardware. In the modulator the trigger signal is generated by the NCO of the digital input; in the demodulator the trigger signal is generated by the NCO of the complex analog input. At the tail of the main loop, the 'frame' software checks the hardware status for error conditions, and checks the SCW for branching.

4 DSP-Algorithms

4.1 Modulator

Besides data handling and coding, pulse shaping is the most important task of the modulator. This job accounts for more than 90% of the processing load. We implemented a 'square root' Nyquist filter at both the transmitter and the receiver. This is the optimal solution for slowly varying Gaussian channels. The normalized tap gains of the
linear FIR-filter solution are

\[ h_k = \left[ 1 - \alpha + \frac{\sin(\pi k/N + \pi/4) \cdot \cos(\pi k/N + \pi/4) + \sin(\pi k/N - \pi/4) \cdot \cos(\pi k/N - \pi/4)}{\pi(k/N - \frac{1}{4\alpha})} \right] \cdot \left[ 1 - \alpha + \frac{\sin(\pi k/N + \pi/4) \cdot \cos(\pi k/N + \pi/4) + \sin(\pi k/N - \pi/4) \cdot \cos(\pi k/N - \pi/4)}{\pi(k/N - \frac{1}{4\alpha})} \right] \]

where \( \alpha \) : rolloff-factor \((0 \leq \alpha \leq 1)\) 
\( N \) : oversampling ratio.

Oversampling is done by exciting the taps only once per symbol, thereby avoiding multiplications by zero. The FIR-coefficients in (1) are further optimized to reduce side-lobes due to truncation and stored in data-ROM. The oversampling factor \( N \) is 8, and the filter length is 9 symbols resulting in negligible distortion.

4.2 Receiver

In the receiver the oversampled digitized orthogonal components \( x_n \) are pulse-shaped first to meet Nyquist’s first criterion and to maximize the SNR. The FIR-filter

\[ y_n = \sum_k h_k \cdot x_{n-k} \]

acts as a matched filter. This convolution algorithm uses the single cycle multiply/add operation. Several linear phase FIR-coefficient sets are stored in tables, e.g. \( \alpha = 0.4 \), \( N = 4 \) and filter length = 9 symbols. The signal is decimated since processing one sample per symbol is sufficient for the succeeding blocks. This reduces the computational load.

Since a fixed local oscillator is used in the IF-section of the demodulator, the correction of the frequency offset must be done digitally by rotating the sequence \( y_n \) backwards with the estimate of the phase-error \( \Delta \phi_n \)

\[ z_n = y_n \cdot \exp(-\phi_{n-1}). \]

For low phase-errors \( \Delta \phi_n \), a good approximation is given by [3]

\[ \Delta \phi_n = \text{Im}\{z_n \cdot \hat{\alpha}_n\}, \quad (4) \]

where \( \hat{\alpha}_n \) is the tentative symbol decision.

The estimate \( \Delta \hat{\phi}_n \) is delivered to a decision-directed second order PLL which is able to cope with the frequency-distortion

\[ \begin{align*}
\hat{\phi}_n &= \hat{\phi}_{n-1} + \gamma_2 \cdot \Delta \phi_n \\
\phi_n &= \phi_{n-1} + \gamma_1 \cdot \Delta \phi_n + \phi_n \pmod{2\pi}
\end{align*} \]

\( \gamma_1 \) and \( \gamma_2 \) act as the loop constants and are given by

\[ \begin{align*}
\gamma_1 &= 1 - \exp(-2\xi \omega_n T_{SYM}) \\
\gamma_2 &= 1 - \gamma_1 - 2 \cos(\omega_n T_{SYM} \sqrt{1 - \xi^2}) \\
&\quad \cdot \exp(-\xi \omega_n T_{SYM})
\end{align*} \]

where \( \xi = 0.7071 \) damping-factor

\[ \omega_n = \frac{2B_L}{2\xi + \xi^2} \]

natural loop frequency

\( B_L \)

equivalent loop noise bandwidth.

These equations were derived by computing the equivalent discrete-time system for the analog PLL [4].

Different settings of the loop-bandwidth are stored in tables and can be exchanged on-line. After transforming the phase values \( \{0, 2\pi\} \) into integers \( \{0, 2^{INT} - 1\} \), the modulo-operation of (5) can be performed by masking. The sin/cos-transformation is done by table look up.

Clock-synchronisation is performed in a similar way. An error signal \( \Delta \hat{f}_n \) is generated by the decision-directed algorithm [5]

\[ \Delta \hat{f}_n = \text{Re}\{z_n \hat{\alpha}_{n-1} - z_{n-1} \hat{\alpha}_n\}. \quad (8) \]

Filtering of the error-signal is performed by a first order PLL since the symbol rate \( T_{SYM} \) is known a priori

\[ \begin{align*}
i_n &= i_{n-1} + \beta \cdot \Delta \hat{f}_n + T_{SYM}
\end{align*} \]

where \( \beta \) is the loop bandwidth factor.
As mentioned before, we implemented a NCO which generates the start of conversion impulses for the A/D-converters with a rate proportional to the actual counter value. Synchronous sampling is done by updating the NCO with \((1 + \beta \cdot \Delta t_n) \cdot T_{SYM}\). This scheme is simple and well suited for the data rates handled by the DSP. It forces sampling in the eye-center. Asynchronous sampling is possible too, but the algorithms are more involved due to skips, and were therefore not implemented.

The back-rotated samples \(z_n\) are finally delivered to the soft-decision VA which selects the most probable path, i.e. the path with the minimum euclidean distance between all paths permitted by the code and the observations \(\{z_n\}\) [2]. The final decisions \(\hat{a}_{n-D}\) are supplied with a sufficient decision delay \(D\). In addition tentative decisions \(\hat{a}_n\) and \(\hat{a}_{n-1}\) were used for decision-directed synchronization. To reduce the complexity of the VA, we took several optimizations into consideration. We chose the feedback-realization of the convolutional encoder [1] that results in simple decoding of the output bits. All possible code symmetries are also taken into account. Back-search of the path-history is substituted by register exchange. The decision delay \(D\) is adapted to the data format of the DSP (16 or 32 symbols).

Since the software is decoupled from demands of the hardware, it was possible to simply connect all the functional blocks. Macros are included if sensible. With a single TMS320C25 data rates up to 32 kbps (64 kbps when pulse-shaping is done externally) are possible for the receiver.

The IF back-to-back performance test (fig.6) shows that the degradation of the realized 8PSK modem is less than 1 dB at 9.6 kbps compared to theoretical results subject to synchronization and implementation losses. The measured coding gain is about 2.7 dB at \(BER = 10^{-5}\) as expected [6].

5 Future aspects

We have currently implemented a differentially coherent Trellis Coded Modem suitable for data transmission on fading channels [7]. We will investigate new algorithms for synchronization on fading channels, and TCM-receiver structures for frequency selective fading channels. The IF-system will be changed from a lowpass to a bandpass version, so that a stable and accurate extraction of the complex modulated signal will be ensured. The software and the attached test equipment will be extended to enable easy setup, flexibility and performance testing of the system.

References


[7] F. Edlbauer, 'Coded 8-PSK Modulation with Differentially Coherent Detection - An Efficient Modulation Scheme for Fading Channels', Globecom '87, Tokyo, Japan, pp. 42.2.1 - 42.2.4, Nov. 1987
Figure 1: Modulator Block Diagram

Figure 2: Timing Synchronization Concept

Figure 3: Demodulator Block Diagram
Figure 4: Functional Demodulator Block Diagram

Figure 5: Trellis diagramm

Figure 6: IF-back-to-back performance test for QPSK and C8PSK

(Theory, x^t uncoded, x^t differential encoded)