

REAL-TIME MMSE TURBO-EQUALIZATION ON THE TMS320C5509 FIXED-POINT DSP

Raphaël Le Bidan, Christophe Laot and Dominique Leroux

GET/ENST Bretagne, Signal & Communications Dept., CNRS TAMCIC
Technopôle Brest Iroise, CS 83818, 29283 BREST Cedex, FRANCE
raphael.lebidan@enst-bretagne.fr

ABSTRACT

We describe the implementation of a low-complexity minimum mean-square error (MMSE) turbo-equalizer on Texas Instruments (TI) TMS320VC5509 device, a low-cost 16-bit fixed-point DSP typically designed for mobile terminals. A data rate of 207 Kb/s per iteration has been achieved. With carefully optimized data quantization, the resulting fixed-point receiver exhibits virtually no performance degradation with respect to an ideal (unquantized) floating-point turbo-equalizer.

1. INTRODUCTION

Intersymbol interference (ISI) constitutes a major obstacle to reliable high data rate transmissions over frequency-selective channels. Turbo-equalization pioneered in [1] combines equalization and channel decoding in an iterative process and realizes an attractive solution to overcome the impairments caused by ISI.

The optimal turbo-equalizer relies on a BCJR-MAP Soft-Input Soft-Output (SISO) equalizer [2], whose complexity precludes a practical implementation when considering multilevel modulations over long delay spread channels. Research efforts have thus been devoted to the design of efficient low-complexity SISO equalizers. Among them, the class of filtering-based SISO equalizers first introduced in [3] offer an interesting alternative to trellis-based equalizers. They maintain a reasonable complexity which grows essentially linearly with the dimension of the signal set and the length of the channel impulse response (CIR). We focus in this paper on the MMSE Interference Canceller - Linear Equalizer (IC-LE) proposed in [4], an attractive receiver for single-carrier broadband wireless transmissions in severe multipaths environments. Building upon the respective works of [5] and [6], the MMSE IC-LE generalizes the classical MMSE linear equalizer by exploiting the reliability of *a priori* information to adapt the equalization strategy accordingly.

Turbo-equalization has evolved over almost a decade of research into a mature technology for which we now foresee practical applications. Extending preliminary investigations reported in [7], this paper describes the real-time implementation of the MMSE IC-LE turbo-equalizer on TI TMS320C5509 device, a low-cost 16-bit fixed-point DSP targeted towards mobile handsets.

The paper is organized as follows. Section 2 describes the transmission system considered. We introduce the MMSE IC-LE SISO equalizer in section 3. Section 4 presents our experimental

demonstration platform. The DSP implementation of the turbo-equalizer is discussed in section 5, and its performance is examined in section 6. Conclusions are finally given in section 7.

2. TRANSMISSION SYSTEM

We will consider in the following the bit-interleaved coded modulation scheme shown in figure 1. Frames of 510 information bits $\{b_k\}$ are encoded by a rate 1/2 recursive systematic convolutional encoder with memory 2 and octal generator polynomials $(1, 5/7)$. 4 tailbits are appended to ensure zero-state trellis termination. The 1024 coded bits $\{c_k^i\}$ with $k = 0, 1, \dots, 511$ and $i = 1, 2$ are interleaved according to a pseudo-random permutation function, grouped and mapped onto $N = 512$ discrete-time QPSK symbols $\{x_n\}$ with zero mean and unit variance $\sigma_x^2 = 1$. These symbols are modulated and transmitted over a frequency-selective channel on a burst per burst basis. The channel is assumed invariant along the burst duration but may change independently between successive bursts (thanks e.g. to ideal frequency hopping). An appropriate guard interval is inserted at the end of each burst to prevent inter-block interference at the receiver side.

We assume a coherent receiver front-end and perfect synchronization, such that the cascade of transmit filtering, transmission over the channel, receive filtering and symbol-rate sampling may be represented by an equivalent discrete-time baseband channel, modeled as a FIR filter with L complex coefficients $\{h_\ell\}$. Following this convention, the channel output at time n is given by

$$y_n = \sum_{\ell=0}^{L-1} h_\ell x_{n-\ell} + w_n \quad (1)$$

where w_n denote uncorrelated complex gaussian noise samples with zero mean and total variance σ_w^2 .

The turbo-equalizer is depicted in figure 1. The SISO equalizer delivers extrinsic information $L_e^E(c_n^i)$ on the coded bits in log-likelihood ratio (LLR) form, that are deinterleaved and passed to the SISO channel decoder. This one generates in turn hard decisions $\{\hat{b}_k\}$ on the information sequence, as well as updated extrinsic information $L_e^D(c_k^i)$ on the coded bits. The quantities $L_e^D(c_k^i)$ are then interleaved and fed back to the equalizer where they are exploited as *a priori* information for a new equalization attempt. A fixed number of 5 iterations was considered in our application.

3. THE MMSE IC-LE SISO EQUALIZER

The overall structure of the MMSE IC-LE is depicted in figure 2. It comprises a soft symbol mapping module and an interference cancellation (IC) structure composed of two FIR filters with respective

This work was supported by France Télécom R&D under research contract CRE 011B032. The development tools were provided by the Texas Instruments ELITE University Program.

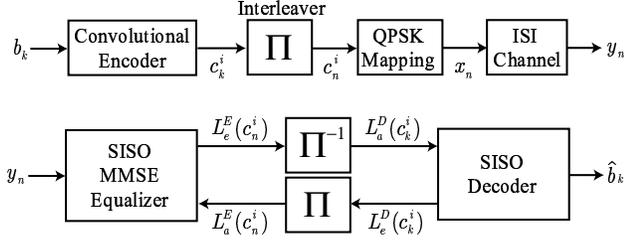


Fig. 1. Block diagram of the transmission scheme.

frequency responses $P(\omega)$ and $Q(\omega)$, followed by a SISO symbol demapper. A full description of the equalizer is provided in [4, 8]. We shall only recall the results pertaining to the implementation.

3.1. Soft symbol mapping

The soft mapping module generates soft symbol estimates $\{\bar{x}_n\}$. They are computed as the expected value of the transmitted symbols with respect to prior probabilities derived from the LLRs delivered by the decoder at the previous iteration. Considering QPSK signaling and assuming that the pair of coded bits (c_n^1, c_n^2) relates to symbol x_n at time n , we obtain

$$\bar{x}_n = \frac{\sigma_x}{\sqrt{2}} \left[\tanh\left(\frac{L_a^E(c_n^1)}{2}\right) + j \tanh\left(\frac{L_a^E(c_n^2)}{2}\right) \right] \quad (2)$$

The soft symbol mapper also computes the variance $\sigma_{\bar{x}}^2$ of the soft estimates. It can be shown that $E(\bar{x}_n) = 0$, yielding

$$\sigma_{\bar{x}}^2 = E(|\bar{x}_n|^2) \approx \frac{1}{N} \sum_{n=0}^{N-1} |\bar{x}_n|^2 \quad (3)$$

Parameter $\sigma_{\bar{x}}^2$ measures the reliability of the data estimates. It is taken into account into the computation of the IC-LE filters coefficients. At the first iteration, no prior information is available. Hence, $L_a^E(c_n^i) = 0$, $\bar{x}_n = 0$ and $\sigma_{\bar{x}}^2 = 0$. As the LLRs reliability increases across the iterative process, $\bar{x}_n \rightarrow x_n$ and $\sigma_{\bar{x}}^2 \rightarrow \sigma_x^2$.

3.2. Interference canceller

The core of the MMSE IC-LE lies in the interference cancellation structure. The equalized sample z_n at time n is given by

$$z_n = \sum_l p_l y_{n-l} - \sum_m q_m \bar{x}_{n-m} \quad (4)$$

where we impose the condition $q_0 = 0$ to prevent the subtraction of the desired signal. The filters coefficients are optimized according to the minimization of the mean-square error $E(|z_n - x_n|^2)$. They are computed once a burst from an estimate of the channel impulse response, and then applied to the whole received sequence. The CIR estimate is typically obtained from a known training sequence embedded in each transmitted packet. Introducing the Fourier transform $H(\omega)$ of the CIR, we obtain [4, 8]

$$P(\omega) = \frac{\sigma_x^2}{1 + \beta \sigma_{\bar{x}}^2} \frac{H^*(\omega)}{(\sigma_x^2 - \sigma_{\bar{x}}^2)|H(\omega)|^2 + \sigma_w^2} \quad (5)$$

with

$$\beta = \frac{1}{2\pi} \int_{-\pi}^{+\pi} \frac{\sigma_x^2 |H(\omega)|^2}{(\sigma_x^2 - \sigma_{\bar{x}}^2)|H(\omega)|^2 + \sigma_w^2} d\omega \quad (6)$$

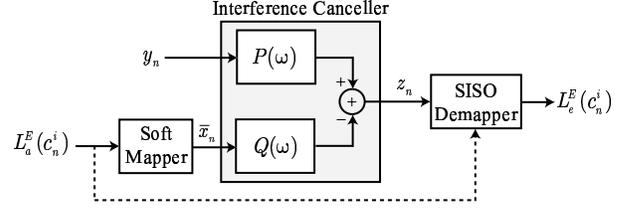


Fig. 2. Block diagram of the SISO MMSE IC-LE.

and, defining $G(\omega) = P(\omega)H(\omega)$,

$$Q(\omega) = G(\omega) - g_0, \quad \text{with } g_0 = \frac{1}{2\pi} \int_{-\pi}^{+\pi} G(\omega) d\omega \quad (7)$$

One readily verifies that for $\sigma_{\bar{x}}^2 \rightarrow 0$, the IC-LE reduces to the conventional MMSE linear equalizer. In contrast, it converges towards the ideal MMSE interference canceller when $\sigma_{\bar{x}}^2 \rightarrow \sigma_x^2$, thereby achieving the matched-filter bound. Hence, we observe that the MMSE IC-LE adapts its equalization strategy as a function of the reliability of the soft data estimates, information that is in fact captured by the variance $\sigma_{\bar{x}}^2$ of these estimates.

Filters $P(\omega)$ and $Q(\omega)$ have infinite length and thus are not directly amenable to a practical implementation. The optimum filters under finite-length constraints may be derived from matrix algebra [8]. This operation requires however a matrix inversion with $\mathcal{O}(N_p^2)$ complexity, N_p denoting the number of taps for the feed-forward filter $P(\omega)$. We rather focused here on an approximate low-complexity alternative relying on the Fast Fourier Transform (FFT) with $\mathcal{O}(N_p \log_2 N_p)$ complexity. Our studies did not show significant performance loss when using this solution in comparison with the optimal matrix inversion approach, for $N_p \geq 32$ [8]. The resulting procedure is summarized in table 1. Note that the feedback filter $Q(\omega)$ has length $N_q = N_p + L - 1$.

-
1. Compute the FFT $\{H_n\}$ of $\{h_n\}$ on N_p points
 2. Compute $D_n = (\sigma_x^2 - \sigma_{\bar{x}}^2)|H_n|^2 + \sigma_w^2$ and $P'_n = H_n^*/D_n$ for $n = 0..N_p - 1$
 3. Compute $\beta = \frac{1}{N_p} \sum_{n=0}^{N_p-1} H_n P'_n$ and $g_0 = \sigma_x^2 \beta / (1 + \beta \sigma_{\bar{x}}^2)$
 4. Compute $P_n = \sigma_x^2 P'_n / (1 + \beta \sigma_{\bar{x}}^2)$ and take the IFFT of $\{P_n\}$ on N_p points to get $\{p_n\}$
 5. Compute $\{q_m\}$ as the convolution of $\{p_n\}$ with $\{h_n\}$ and set $q_0 = 0$
-

Table 1. Filters coefficients computation procedure.

3.3. SISO symbol demapper

The SISO demapper finally delivers extrinsic LLRs $L_e^E(c_n^i)$ on the coded bits, computed from the knowledge of the equalized sample z_n at time n and possibly from the *a priori* LLRs $L_a^E(c_n^i)$ for high-order modulations [9]. Let us write the equalized sample as

$$z_n = g_0 x_n + \nu_n \quad (8)$$

where ν_n denote the residual noise and interference term at the canceller output. The SISO demapper operates under the assump-

tion that ν_n is gaussian with variance σ_ν^2 [5]. It can be shown that $\sigma_\nu^2 = \sigma_x^2 g_0 (1 - g_0)$ (see e.g. [9]). This yields in the QPSK case

$$L_e^E(c_n^1) = \frac{4/\sqrt{2}}{1 - g_0} \text{Re}(z_n), \quad L_e^E(c_n^2) = \frac{4/\sqrt{2}}{1 - g_0} \text{Im}(z_n) \quad (9)$$

4. OVERVIEW OF THE DEMONSTRATION PLATFORM

The demonstration platform is composed of a host PC communicating with a target DSP evaluation board, as shown in figure 3. The PC runs a monitoring application that generates the data at the channel output, sends the resulting signal to the DSP that implements the turbo-equalizer, retrieves the processed data and updates the link metrology (bit-error rate and frame-error rate).

The DSP board includes a TI TMS320VC5509 DSP device operating at 120 MHz (240 MIPS). The C5509 is a high performance low-cost 16-bit fixed-point DSP with low power consumption, typically targeted towards mobile terminals. Communication between the DSP board and the host PC relies on the *Real-Time Data eXchange* (RTDX) technology offered by TI. RTDX allows data transfers with rates ranging from 30 Kb/s to 2 Mb/s between the computer and the DSP, throughout the JTAG emulation link, without stopping the target application.

5. DSP IMPLEMENTATION ISSUES

The turbo-equalizer has been designed to operate with SNR values in the 0...20 dB range, with maximum delay spreads of $L = 16$ channel taps. Perfect knowledge of the CIR and noise variance σ_w^2 was assumed.

We have chosen to implement the turbo-equalizer using the C language in order to speed-up the development process and favor code portability. Data quantization has been carefully optimized at each stage of the receiver in order to maintain the highest possible precision while avoiding underflows or overflows resulting from the use of 16-bit fixed-point arithmetic. Preliminary simulations were performed with a floating-point C model of the turbo-equalizer in order to find the proper number of bits required to accurately represent the quantities involved in the iterative process.

In the following, we shall use the notation $S(m.n)$ to describe a signed fixed-point number with m bits of dynamic (sign bit excluded) and n bits for the fractional part (precision). At the receiver input, the observations $\{y_n\}$ and channel taps $\{h_\ell\}$ are quantized in $S(0.15)$ format, also called Q15 representation.

5.1. SISO equalizer implementation

The soft mapping module takes *a priori* LLRs $L_a^E(c_n^i)$ in $S(3.5)$ representation and delivers soft symbol estimates in Q15 format. Parameter σ_x^2 is also stored in Q15 representation. The $\tanh(\lambda/2)$ operation arising in (2) has been precomputed and stored in RAM. Quantization range was limited to $\lambda \in [-8, +8)$ with a quantization step of 2^{-5} in order to match the range of the input LLRs. The resulting look-up table (LUT) has 512 entries.

The IC has been realized using $N_p = 32$ taps for the feedforward filter and $N_q = 47$ taps for the feedback filter. We found by simulation that the $S(3.12)$ fixed-point format was well-suited to accurately represent the filter taps. The equalizer's implementation takes advantage of the optimized FIR filtering and FFT functions provided by TI Digital Signal Processing Library (DSPLIB).

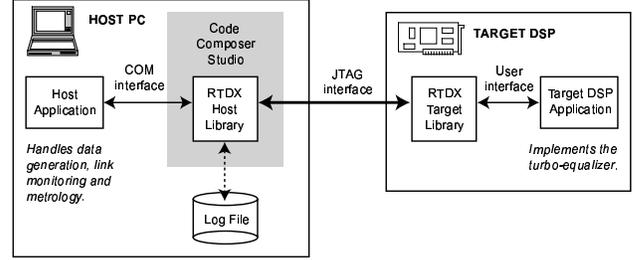


Fig. 3. Block diagram of the demonstration platform.

The IC delivers equalized samples $\{z_n\}$ in $S(3.12)$ representation to the SISO demapping module, which generates in turn extrinsic LLRs $L_e^E(c_n^i)$ in $S(4.5)$ format that are sent to the decoder.

5.2. SISO decoder implementation

The SISO decoder implements the Max-Log-Map algorithm [10]. Decoding proceeds in 2 steps. The backward recursion is performed first, and the resulting backward state metrics are stored in RAM. The decoder then performs the forward recursion and simultaneously delivers extrinsic LLRs on coded bits in $S(3.5)$ representation, as well as hard decisions $\{\hat{b}_k\}$ on the information sequence. No additional storage is required for the transitions metrics which are recomputed when needed.

State metrics accumulation remains the critical issue in the decoder design as these quantities may overflow during the forward and backward recursions. The metric growth problem was solved by taking advantage of the fact that the C5509 intrinsically uses two's complement arithmetic for its operations. This approach has the benefit of not requiring any explicit normalization operation at all if the difference between any two state metrics fits within the DSP 16-bits representation [10]. We used the method proposed in [11] to find the exact maximum values assumed by the state metrics and LLRs during decoding, and optimized the fixed-point format of the input LLRs accordingly.

6. SYSTEM PERFORMANCE

6.1. Achievable bit-rate and storage requirements

Table 2 summarizes the average number of DSP cycles required to perform the different signal processing functions in the receiver. These measurements were obtained using the optimization level -o3 of Code Composer Studio C compiler. Accounting for the fact that 1 DSP cycle executes in 8.33 ns, we obtain a data rate of 207 Kb/s per iteration or, equivalently, 41 Kb/s with 5 iterations. To the best of the authors knowledge, this is the first DSP implementation result reported so far for a turbo-equalizer in the literature. The SISO equalizer and decoder account for 53% and 43% of the total running time per iteration respectively. Note that the latter function may benefit from assembly language optimizations by taking advantage of the dedicated Add-Compare-Select instructions provided by the C55x DSP family.

The turbo-equalizer implementation has a code size of 3747 words (1 word = 16 bits) and uses 10118 words of data. These values are fully compatible with the 32 Kwords of on-chip dual-access RAM available on the C5509. We emphasize that no particular attempt was made to optimize the storage requirements.

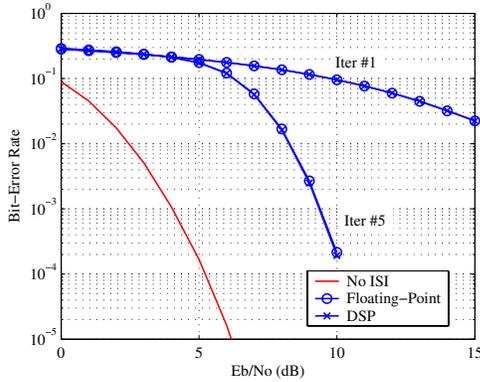


Fig. 4. BER performance over the Proakis C static channel.

| Function | Cycles number |
|-------------------------------|---------------|
| SISO mapping | 17258 |
| Equalization | 115965 |
| SISO demapping | 21835 |
| Interleaving / Deinterleaving | 6157 each |
| SISO decoding | 126977 |
| Total / iteration | 294349 |

Table 2. Average number of DSP cycles per function

6.2. Experimental results

In order to quantify the performance loss obtained with respect to an ideal (unquantized) floating-point receiver, simulations were conducted with the DSP demonstration platform over the severe-ISI time-invariant Proakis C channel with response $\{0.227, 0.460, 0.688, 0.460, 0.227\}$, and over the time-varying quasi-static EQ6 channel model with 6 symbol-spaced taps of equal average power $1/6$. BER results are shown in figures 4 and 5 at the first and fifth iterations. We observe that the fixed-point DSP implementation exhibits virtually no performance degradation in comparison with the floating-point receiver. These results have been further confirmed by additional simulations over other channel models.

7. CONCLUSIONS

We have described the real-time implementation of an efficient MMSE turbo-equalizer on the TMS320C5509 fixed-point DSP. Using only the C language, a data rate of 207 Kb/s per iteration has been obtained. The resulting implementation does not exhibit any performance loss with respect to an ideal unquantized receiver.

Future work will include rewriting critical code sections in optimized assembly language in order to improve the data rate, and adding support for higher-order (QAM) modulations. This work also constitutes the preliminary step towards the realization of an hardware FPGA prototype able to operate at several Mb/s.

8. REFERENCES

[1] C. Douillard, M. Jézéquel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol in-

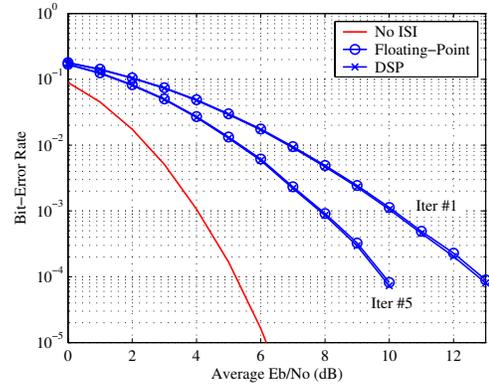


Fig. 5. BER performance over the EQ6 quasi-static channel.

terference: Turbo-equalization," *European Trans. Telecommun.*, vol. 6, no. 5, pp. 507–511, Sept.-Oct. 1995.

- [2] G. Bauch, H. Khorram, and J. Hagenauer, "Iterative equalization and decoding in mobile communications systems," in *Proc. 2nd European Personal Mobile Commun. Conf. EPMCC'97*, Bonn, Germany, Sept.-Oct. 1997, pp. 307–312.
- [3] A. Glavieux, C. Laot, and J. Labat, "Turbo equalization over a frequency selective channel," in *Proc. Int. Symp. on Turbo Codes*, Brest, France, 3-5 Sept. 1997, pp. 96–102.
- [4] C. Laot, R. Le Bidan, and D. Leroux, "Low complexity linear turbo equalization: A possible solution for EDGE," *Submitted to IEEE Trans. Wireless Commun.*, Aug. 2002, [Online]. Available: <http://www-sc.enst-bretagne.fr/~laot>.
- [5] M. Tüchler, A. C. Singer, and R. Kötter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. Signal Processing*, vol. 50, no. 3, pp. 673–683, Mar. 2002.
- [6] A. M. Chan and G. W. Wornell, "A class of block-iterative equalizers for intersymbol interference channels: Fixed channel results," *IEEE Trans. Commun.*, vol. 49, no. 11, pp. 1966–1976, Nov. 2001.
- [7] R. Le Bidan, C. Laot, and D. Leroux, "Fixed-point implementation of an efficient low-complexity turbo-equalization scheme," in *Proc. 3rd Int. Symp. on Turbo Codes & Related Topics*, Brest, France, 1-5 Sept. 2003, pp. 415–418.
- [8] R. Le Bidan, "Turbo-equalization for bandwidth-efficient digital communications over frequency-selective channels," Ph.D. dissertation, INSA Rennes, France, Nov. 2003.
- [9] A. Dejonghe and L. Vandendorpe, "Turbo-equalization for multilevel modulation: An efficient low-complexity scheme," in *Proc. IEEE Int. Conf. Commun. ICC'02*, vol. 3, New York City, NY, 28 Apr.-2 May 2002, pp. 1863–1867.
- [10] G. Montorsi and S. Benedetto, "Design of fixed-point iterative decoders for concatenated codes with interleavers," *IEEE J. Select. Areas Commun.*, vol. 19, no. 5, pp. 871–882, May 2001.
- [11] E. Boutillon, W. J. Gross, and P. G. Gulak, "VLSI architectures for the MAP algorithm," *IEEE Trans. Commun.*, vol. 51, no. 2, pp. 175–185, Feb. 2003.