

Successive Interference Cancellation for 3G Downlink: Algorithm and VLSI Architecture

Sandro Belfanti*, Christian Benkeser*, Karim Badawi*, Qiuting Huang*, Andreas Burg†

*Integrated Systems Laboratory
ETH Zurich, 8092 Zurich, Switzerland

†Telecommunications Circuits Laboratory
EPF Lausanne, 1015 Lausanne, Switzerland

Abstract—This paper presents a VLSI implementation of an MMSE successive interference cancellation multiuser detector (SIC-MUD) for the downlink of a TD-SCDMA system. Computation in the frequency domain, group-wise interference cancellation, and pre-computation of filter coefficients enable an efficient architecture suitable for mobile handsets. Our implementation in 0.13 μm CMOS technology proves that the SIC-MUD is a viable solution for the TD-SCDMA downlink, providing a notable performance gain at a moderate increase in complexity compared to linear equalizers.

I. INTRODUCTION

In recent years, a new family of 3G CDMA cellular standards based on time division duplex (TDD), as opposed to the frequency division duplex (FDD) commonly used in 3G systems in Europe, has received increasing attention. This is partially due to the fact that no duplexers are required in such radio transceivers and partially because asymmetric data rates can be accommodated with efficient spectrum utilization. A narrow-band version of the UMTS TDD standard, known as low chip rate (LCR) or TD-SCDMA [1], has been successfully launched in China only few years ago, and the corresponding evolved 3.5G standard TD-HSPA has been specified for future high-speed packet data transmission.

Unfortunately, in CDMA systems multipath communication channels cause inter-symbol interference (ISI) and consequently destroy the orthogonality of spreading codes, leading to severe multiple access interference (MAI) between the users simultaneously accessing the channel. Hence, digital baseband receiver performance of such systems strongly depends on efficient channel equalization and detection algorithms. However, due to the large dimensions of such systems, (optimum) joint maximum-likelihood sequence detection is in general not feasible. Instead, power- and area-efficient linear algorithms, such as linear minimum mean squared error (MMSE) equalization, are typically employed (e.g., [2]). Unfortunately, the performance of linear equalization is fairly low, especially in heavily loaded scenarios where most or all spreading codes are used simultaneously.

Interference-cancellation (IC) multiuser detectors (MUD) can outperform linear equalizers (LEs) in combatting ISI and MAI in CDMA systems [3]. However, the corresponding computational complexity is higher, and increases in the number

of codes. Therefore, IC-MUDs are only used in uplink, where base-stations do not need to rely on low-power and low-cost receivers, and where linear equalization performs even less well than in downlink.

We propose to use IC-MUDs also in the downlink of the 3G TDD standard TD-SCDMA to improve receiver performance. Contrary to 3G FDD, where up to 256 users can be multiplexed via CDMA on the same channel, TD-SCDMA specifies a maximum of only 16 codes, rendering IC-MUDs a viable solution also for downlink. In this paper, we present an efficient way of realizing an MMSE successive IC-MUD (SIC-MUD) for TD-SCDMA, and prove our concepts with a corresponding low-complexity VLSI implementation.

II. SYSTEM MODEL

TD-SCDMA uses a combination of TDMA and CDMA, where up to 16 codes are active simultaneously during the same timeslot. Each user is assigned one or multiple codes, depending on the required throughput. Each timeslot contains one burst, consisting of two data blocks of 352 chips each, separated by a 144-chip midamble and followed by a guard period of length 16 chips. We denote the k -th spreading code by $\mathbf{c}^{(k)} = [c_0^{(k)} \dots c_K^{(k)}]^T$, where $K = 16$ is the spreading factor. The $N = 44$ QPSK modulated data symbols corresponding to the k -th code are given by $\mathbf{d}^{(k)}$. To simplify the notation we introduce the stacked symbol vector

$$\mathbf{d} = [d_0^{(0)} d_0^{(1)} \dots d_0^{(Q)} d_1^{(0)} d_1^{(1)} \dots d_{N-1}^{(Q-1)}]^T,$$

where Q is the number of active codes in the current timeslot. We define the block-diagonal spreading matrix \mathbf{C} containing the active spreading codes $[\mathbf{c}^{(0)} \dots \mathbf{c}^{(Q-1)}]$ on the diagonal, such that the sum of chips of all users is given by $\mathbf{s} = \mathbf{C}\mathbf{d}$, assuming that transmit power is equal for all codes. A burst denoted by $\bar{\mathbf{s}}$ is formed by adding the midamble and the guard period. At the user-equipment (UE) side, the received baseband signal \mathbf{r} with multipath propagation is given by

$$\mathbf{r} = \mathbf{H}\bar{\mathbf{s}} + \mathbf{n},$$

where \mathbf{H} is the channel matrix and \mathbf{n} is zero mean additive Gaussian noise with a variance of σ_N^2 . The matrix \mathbf{H} is a Toeplitz matrix containing the equivalent channel impulse response (CIR) $\mathbf{h} = [h_0, \dots, h_{W-1}]^T$, which also includes the

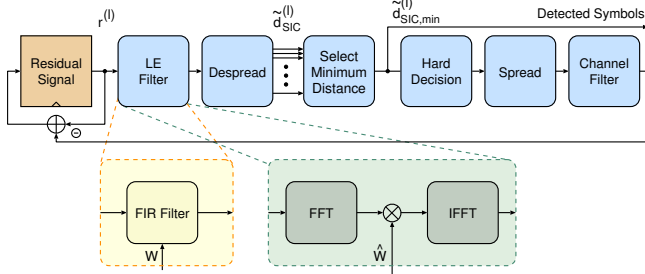


Fig. 1: Block diagram for the SIC-MUD. Two different possibilities for the implementation of the LE filter are highlighted.

effect of a root-raised cosine (RRC) transmit filter as specified, and the corresponding RRC matched filter at the receiver.

Since each burst is followed by a guard period, the last 16 columns of \mathbf{H} are multiplied by zero and consequently have no influence on the received signal. These columns can be altered without affecting the vector \mathbf{r} . This property can be exploited to formulate an equivalent channel model using a circulant channel matrix \mathbf{H}_{circ} . \mathbf{H}_{circ} is the same as \mathbf{H} except for the last columns, and therefore

$$\mathbf{r} = \mathbf{H}_{\text{circ}}\bar{\mathbf{s}} + \mathbf{n}.$$

III. SUCCESSIVE INTERFERENCE CANCELLATION

The SIC-MUD [4] is an iterative algorithm based on the concept of interfering signal regeneration and subsequent cancellation. Our implementation of the algorithm operates on bursts corresponding to N transmit data symbols.

The first step in each iteration is to estimate the transmit data symbols $\mathbf{d}^{(k)}$ for all codes in parallel. To this end, a chip-level LE and a despreader equalize the effect of the channel to recover K almost-orthogonal spreading codes, forming symbol-estimates $\tilde{\mathbf{d}}_{\text{SIC}}^{(l)}$ for the l -th iteration (Fig. 1).

During the first iteration, the LE filter is given by the MMSE criterion:

$$\mathbf{W}_{\text{MMSE}} = \mathbf{H}^H (\mathbf{H}\mathbf{H}^H + \sigma_N^2 \mathbf{I})^{-1}. \quad (1)$$

The resulting symbol-estimates are ranked according to their reliability, which is measured in terms of signal to interference plus noise ratio (SINR). Note that the ordering is realized on a per symbol basis, i.e., every symbol can have a separate cancellation order. The symbol corresponding to the highest SINR is selected as output. Subsequently, this selected symbol is regarded as potential cause of interference for the remaining codes. To cancel the contribution of the selected symbol, its interference is regenerated through spreading and filtering with the estimated CIR in the *channel filter* (Fig. 1). The regeneration can be performed in two ways: using hard-decisions or soft-estimates [5]. For the hard-decision (HD) SIC-MUD the detected symbol is mapped to the closest constellation point to obtain the HD, which is used for cancellation. For the soft-decision (SD) SIC-MUD a soft symbol-estimate is calculated, which represents the uncertainty of the decision. Finally, the first iteration concludes with the cancellation of the reconstructed interfering signal from the received signal by means of subtraction.

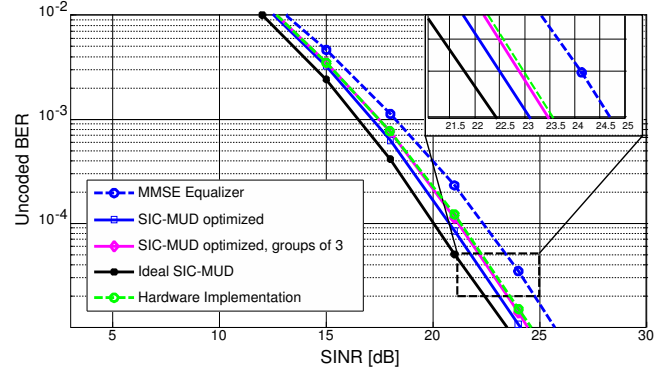


Fig. 2: Comparison between SIC-MUD and MMSE equalizer *Case 2* channel. Only the *Hardware Implementation* simulation uses shortened filters.

During subsequent iterations this procedure is repeated for the residual signal after l -th cancellation, denoted by $\mathbf{r}^{(l)}$. This signal is passed through the LE filter, however, the filter calculation has to be slightly different [6] compared to the MMSE filter calculation in the first iteration to account for the already cancelled symbols:

$$\mathbf{W}_{\text{SIC}} = \mathbf{H}^H (\mathbf{H}\mathbf{C}\mathbf{A}\mathbf{C}^H \mathbf{H}^H + \sigma_N^2 \mathbf{I})^{-1}, \quad (2)$$

where \mathbf{A} is a diagonal matrix in which the diagonal elements σ_i^2 are the variances of the corresponding data symbol in $\tilde{\mathbf{d}}_{\text{SIC}}^{(l)}$. For the HD SIC-MUD the σ_i^2 are either 1 if $\tilde{d}_{\text{SIC},i}^{(l)}$ is not yet cancelled or 0 otherwise, assuming perfect cancellation. In the case of SDs, the diagonal elements represent the estimation error of the soft-symbol. The SIC-MUD algorithm proceeds, as described above, by successively cancelling the corresponding interference after each iteration until all symbols for all the codes have been detected.

The BER of the SIC-MUD is compared to the MMSE equalizer in Fig. 2. Both algorithms were simulated in a fully loaded system with 16 active codes for the *Case 1* and *Case 2* channel defined in the TD-SCDMA standard [1]. Perfect channel knowledge has been assumed during our simulations. Compared to the MMSE, the SIC-MUD shows a gain of 2.2 dB and 2.3 dB for *Case 1* and *Case 2*, respectively. Note that Fig. 2 shows the simulation results for the *Case 2* channel only.

IV. LOW-COMPLEXITY SIC-MUD FOR VLSI IMPLEMENTATION

HD SIC-MUD implementations are less complex than SD SIC-MUD designs, because no soft-symbol computation is required, the word widths in the feedback-path are smaller, and the filter coefficients computation in (2) is simpler (Sec. III). However, in both HD SIC-MUD and SD SIC-MUD, the coefficients of the filter (2) have to be re-computed in each iteration, because \mathbf{A} has to be updated according to the cancellations, which requires costly matrix inversion and multiplication.

We propose an approximation of the HD SIC-MUD, where the MMSE filter coefficients required for the first IC iteration are used for *all* iterations. Thus, (1) can be pre-computed, to avoid expensive filter coefficient re-computation between sub-

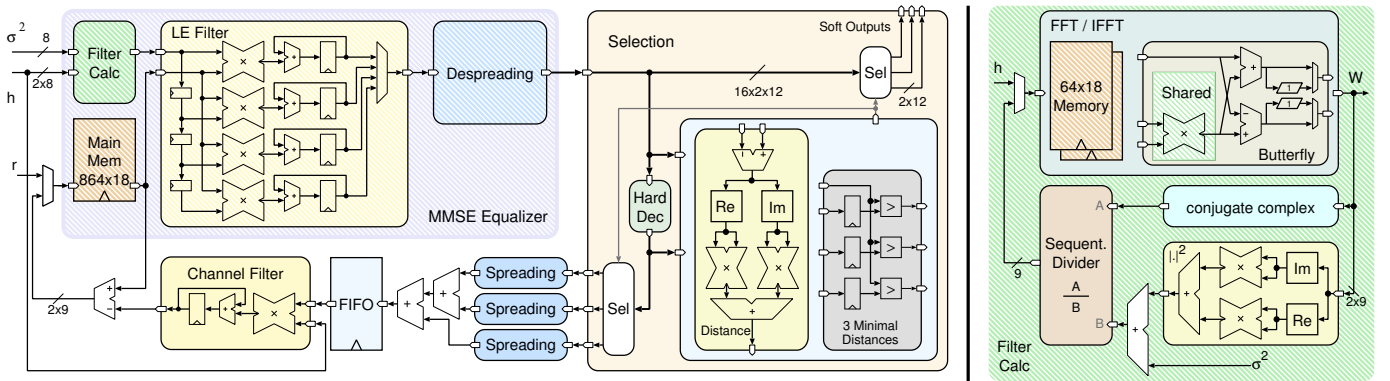


Fig. 3: Left: architecture of the implemented SIC-MUD. Right: detailed block-diagram of the filter calculation.

sequent iterations. Moreover, our approach enables an efficient FFT-based calculation of the MMSE filter coefficients¹.

The filter calculation (1) requires the inversion of an $N \times N$ matrix. Using the circulant channel matrix \mathbf{H}_{circ} instead of \mathbf{H} , the computation can be carried out efficiently in the frequency domain [7], since the discrete Fourier transformation (DFT) matrix diagonalizes any circulant matrix. In the frequency domain, the equalization step reduces to a set of element-wise calculations $\hat{s}_{\text{MMSE},i} = \hat{W}_i \hat{r}_i$, where \hat{r} and $\hat{\mathbf{W}}$ are the DFTs of \mathbf{r} and \mathbf{W} respectively. The filter coefficients \hat{W}_i are calculated from the DFT of \mathbf{h} , denoted by $\hat{\mathbf{h}}$, according to

$$\hat{W}_i = \frac{\hat{h}_i^*}{|\hat{h}_i|^2 + \sigma_N^2}. \quad (3)$$

Transforming a received burst of r_i to the frequency domain and equalizing requires the calculation of a fast Fourier transform (FFT) to obtain \hat{r}_i of each burst element, one multiplication per \hat{r}_i for the equalization, and an inverse fast Fourier Transform (IFFT) to convert the signal back to the time-domain.² Clearly, for TD-SCDMA with bursts of 864 chips, this approach requires large buffer capacities to store the bursts transformed to the frequency domain.

The memory requirements can be significantly reduced by employing time-domain equalization using a finite impulse response (FIR) filter. Instead of transforming the entire burst to frequency domain and back, the filter coefficients are transformed to the time-domain according to $\mathbf{W} = \text{IFFT}\{\hat{\mathbf{W}}\}$. Thus, the filter coefficients can still be efficiently computed in frequency domain, but the actual equalization is performed in time-domain, where burst-wise processing is not required. The performance of our optimized approach that computes filter coefficients in frequency domain, avoids filter coefficient recalculation, and performs channel equalization in time-domain is within 0.7 dB of the ideal HD SIC-MUD algorithm (Fig. 2).

In order to increase the throughput of the proposed algorithm, several codes can be cancelled in parallel [9] by arranging codes into groups of similar reliability, trading performance against latency of the SIC-MUD iterations. Simulations

have shown that this group-wise SIC-MUD with a fixed group size of $M_{\text{SIC}} = 3$, as used in our hardware implementation, degrades BER performance by only 0.3 dB (Fig. 2).

V. ARCHITECTURE AND RESULTS

The block diagram of the VLSI implementation of the time-domain equalization approach described in Sec. III is depicted in the left-hand side of Fig. 3. The circuit consists of three main parts: the linear equalization to obtain $\tilde{\mathbf{d}}_{\text{SIC}}^{(l)}$ which comprises an MMSE equalizer, a selection unit which selects the most reliable codes used for cancellation, and the feedback path for the group-wise SIC-MUD with $M_{\text{SIC}} = 3$.

A. Filtering and Iterative Cancellation

When a new burst is received, the samples are loaded into the main memory, which stores one burst of 9 bit I/Q-samples. After each iteration the samples in the main memory are updated, such that the residual signal $\mathbf{r}^{(l)}$ after the l -th interference cancellation is stored. The LE is implemented as an FIR filter. Since the filter length is crucial for the complexity of the architecture, the number of FIR coefficients was shortened from 864, the length of one burst, to only 64, obtained through numerical simulations. With this approximation 4 multipliers are sufficient to achieve the required throughput at a clock frequency of 200 Mhz (see Sec. V-C).

Next, the filter output is despread, which results in K symbols in parallel. For each symbol a HD is calculated by mapping the symbol to the closest constellation point. Subsequently, the squared distances between the HDs and the estimated symbols are computed, and the M_{SIC} symbols corresponding to the smallest distances are determined to perform a SINR-based ordering (see Sec. III).

These M_{SIC} symbols are found by comparing each squared distance, one symbol at the time, to M_{SIC} registers storing the minimum values. If the new squared distance is smaller it is saved along with the corresponding $\tilde{d}_{\text{SIC}}^{(l)}$ and HD. After all K symbols have been processed the registers contain the $M_{\text{SIC}} = 3$ minimal distances, the corresponding $\tilde{d}_{\text{SIC}}^{(l)}$ and HDs. The $\tilde{d}_{\text{SIC}}^{(l)}$ are used as soft outputs of the SIC-MUD, while the HDs are required for interference cancellation (left-hand side of Fig. 3). To avoid cancelling the same code twice, a buffer with 16 bits for every symbol of the processed

¹Note that a similar FFT-based calculation fails when re-computing the filter coefficients in each iteration according to (2), because IC iterations destroy the Toeplitz property of the matrix \mathbf{A} .

²The DFT can be efficiently realized with the FFT[8].

TABLE I: Synthesis results for 200 MHz

Unit	Size ^a [kGE]	Percentage [%]
Total	94.4	100
Filter Calculation ^b	18.6	19.7
LE Filter ^b	20.2	21.4
Despreading	17.9	18.9
Selection	14.4	15.2
Spreading	3.2	3.4
FIFO	4.1	4.3
Channel FIR	2.7	2.8
Main Memory	12.7	13.5

^aOne gate equivalent corresponds to a two-input drive-one NAND gate.

^bOne complex multiplier considered in the *LE Filter* synthesis results is used by the *Filter Calculation* block.

burst keeps track of the codes already cancelled. The selected HDs are then re-spread and summed up. The additional spreading units and the adders are, apart from the slightly more complex minimum distance search, the only additional hardware resources required to support the group-wise SIC. The implementation overhead due to grouping is small, as can be seen in Tbl. I, which contains the synthesis results discussed in Sec. V-C. The sum of the signals is fed through an FIR filter, the *channel filter*, containing the channel taps as coefficients. Its short length enables an implementation using a single multiplier, minimizing silicon area. The result of the FIR filtering is subtracted from the residual signal of the previous iteration $\mathbf{r}^{(l-1)}$, generating the new signal $\mathbf{r}^{(l)}$, which is stored in the main memory.

B. Filter Coefficient Calculation

The filter coefficient calculation is performed in the frequency domain (see Sec. III). The computation takes place once per burst and consists of three main components: the FFT/IFFT unit, the calculation of the denominator from (3) and a sequential divider (right-hand side of Fig. 3). The FFT and the IFFT share the same hardware. The FFT is implemented using a radix-2 decimation-in-time architecture, employing a single radix-2 butterfly. To reduce silicon area, the butterfly implementation shares the complex multiplier with the LE filter. The FFT size was reduced to 128, since the loss in BER performance is negligible, as the simulation with the bit-true hardware model in Fig. 2 illustrates. The sequential divider is time-shared to calculate first the real and then the imaginary part of the quotient. The frequency domain coefficients are then transformed back to the time domain by reusing the FFT/IFFT block described above.

C. Synthesis Results

The SIC-MUD architecture was synthesized for a $0.13\mu\text{m}$ process, resulting in a total area of 0.48mm^2 . The detailed synthesis results are provided in Tbl. I. The clock has been constrained to 200 Mhz to achieve the required throughput, as will be shown later. The optimized algorithm performs 1.2 dB better than the MMSE equalizer and the loss due to previously described approximations and fixed-point representation of our hardware model is within 0.05 dB (Fig. 2).

Note that the calculation of the filter coefficients requires less than 20 kGE, i.e., only about 20% of the total area, because the computation in frequency domain is less complex and suitable for hardware integration. The SIC-MUD contains a complete MMSE equalizer, namely the four blocks: filter calculation, main memory, LE filter and the despreading (Fig. 3). This MMSE equalizer implementation was used to assess the additional complexity required by the SIC-MUD. Comparing the area of the complete architecture to this MMSE equalizer subunit shows that only an additional 24.3 kGE or 25.8 % are needed to realize the cancellation iterations. For a more fair comparison we consider an MMSE equalizer implementation using a single multiplier, because it would provide the required throughput at the same clock speed used in our SIC-MUD design. Compared to this optimized MMSE equalizer reference implementation, the silicon area of our SIC-MUD architecture is 43 % higher.

In order to meet the TD-SCDMA throughput requirement of 1.28 Mcps a clock frequency of $f = 200\text{MHz}$ is sufficient, even leaving a margin of $240\mu\text{s}$ for other less complex signal processing tasks, such as, e.g., the channel estimation. The key for achieving the required throughput with our SIC-MUD architecture at such a moderate clock frequency is the group-wise processing and the reuse of the pre-computed filter coefficients during all iterations.

VI. CONCLUSION

In this paper, we have shown that SIC-MUD is a suitable candidate for channel equalization and detection in the TD-SCDMA downlink. The proposed algorithmic optimizations allow for an efficient architecture that leads only to a moderate increase of silicon area when compared to MMSE linear equalizer realizations. The implemented design outperforms ideal MMSE equalization by 1.2 dB. Producing a noticeable gain at a hardware overhead, that is almost negligible when considering the total silicon area of a digital baseband transceiver IC, renders the SIC-MUD a viable alternative to traditional linear equalizers.

REFERENCES

- [1] 3GPP; *Tech. Spec. Grp. Radio Access Network; User Equipment (UE) radio transmission and reception (TDD)*, TS 25.102, Jan. 2012.
- [2] S. Wang, "Comparative study of MMSE-BLE joint-detection based on block-FFT and cholesky factorization for TD-SCDMA," in *WiCom*, oct. 2008, pp. 1–4.
- [3] S. Gollamudi and Y.-F. Huang, "On the dimensional limitations of linear multiuser detection," in *Proc. Allerton Conference*, 1998.
- [4] T. Guess, "Optimal sequences for CDMA with decision-feedback receivers," *IEEE Trans. Inf. Theory*, vol. 49, no. 4, pp. 886–900, apr 2003.
- [5] J. Wang and S. Li, "Soft versus hard interference cancellation in MMSE OSIC MIMO detector: A comparative study," in *Proc. ISWCS*, oct 2007.
- [6] Y.-S. Choi, P. Voltz, and F. Cassara, "On channel estimation and detection for multicarrier signals in fast and selective rayleigh fading channels," *IEEE Trans. Comm.*, vol. 49, no. 8, pp. 1375–1387, aug 2001.
- [7] M. Vollmer, M. Haardt, and J. Gotze, "Comparative study of joint-detection techniques for TD-CDMA based mobile radio systems," *IEEE JSAC*, vol. 19, no. 8, pp. 1461–1475, aug 2001.
- [8] J. Cooley and J. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Mathematics of Computation*, vol. 19, no. 90, pp. 297–301, 1965.
- [9] F. v.d. Wijk, G. Janssen, and R. Prasad, "Groupwise successive interference cancellation in a DS/CDMA system," in *Proc. PIMRC*, sep 1995.