Adaptive Implementation of DS-CDMA with Interference Cancellation

K R Madhumitha
M.Tech Vlsi And Embedded System
T John Institute Of Technology
Bangalore

Mrs.Roopashree. K. S
Asst. Professor Ece
T John Institute Of Technology
Bangalore

***Abstract*— This paper introduces iterative joint detection for multiple access interference using direct-sequence code-division multiple-access (DS-CDMA).Partition spread spectrum technique has been used for joint-detection which is a general method of interleave-division multiple access. To perform the decoding in iterative mode, turbo and sum-product decoding has been used. To reduce the implementation complexity linear scheme is used and to remove the interference produced by other users parallel processing has been adapted to improve the reliability. The parallel processing can be done in multiple stages. Synthesis has been performed on Xilinx design tool. Results obtained from the simulation are given here.**

Index Terms—Digital circuits, direct-sequence code-division multiple access, field programmable gate arrays, interference cancellation.

I. INTRODUCTION

Multiuser communication system uses Code division multiplexing so that maximum number of user simultaneously communicates over a channel. According to this technique the spectrum will not be divided into time or frequency slots, the users will be separated by an unique pseudo random signature in the frequency range.

While decoding the signals, the same sequence of pseudo-random is used. After receiving the signal the error parameters will be measured like minimum mean squared error based filters [1] or matched filters [2]. Matched filter implementation is used for second generation DS-CDMA and implemented as rake receiver. Due to ignoring the multiple access interference in DS-CDMA, some capacity limits are demonstrated. In conventional DS-CDMA required power is more because of near-far-effect [3]. Here we tried to increase the number of user according to hardware feasibility. The MAI detector has not been used because of complexity issues [4]. For error correction low –density parity check [5] decoders and turbo decoders [6] has been used.

Proposed detectors are better than conventional detectors in terms of complexity.

There are different type of detectors: Some are: parallel interference cancellation (PIC) [7], [8], multistage PIC [9], successive interference cancellation (SIC) [10], expectation maximization [11] and space alternating generalized expectation (SAGE) [12], [13]. Here we are considering Partition spread spectrum-CDMA detector.

PS-CDMA is a generalized form of interleave division multiplexing (IDMA) [14] and uses PIC and LDPC [15], [16]. Additionally, the PS-CDMA system does not require the strict power controls of conventional DS-CDMA [16]. Further, it can be shown that PS-CDMA with iterative decoding can approach both the performance of optimal detection of CDMA, [17], and also the capacity of the CDMA channel if used together with spatial coupling [18].

PS-CDMA is an iterative algorithm which is useful exploring the results according to the data. In PS-CDMA direct-sequence spread spectrum has been considered. In other iterative joint decoding algorithm with PS-CDMA divided into two parts demodulation and decoding.

The demodulation is an iterative process so it considers repetition codes and removes the interference while the decoding process applies standard off-the-shelf error control codes which are applied externally to largely interference-free individual signal streams. In this paper we are considering iterative demodulation, which is novel technique.

By dividing original DSSS into M sections we can observe the system and repetition of code also can be observed easily.

An interleaver also has been added as shown in fig.1. A correlated system will be created which is adaptable for iterative demodulation.

Each repetition of a data symbol is known as a partition and M partitions use up the same resources as a single spread symbol in conventional CDMA [19], [20]. Partitioning results in N⁄M chips/partition, and therefore the overall processing gain is preserved at M*N/M=N.

In PS-CDMA, at the receiving side K matched filters are used for estimation of each user. In the same manner sum-product algorithm is used to cancel the multiple access interference.

Cancellation will occur when the interleaving and soft decision is performed. As it turns out, involving only the repetition code in the cancellation of the mutual interference is sufficient to achieve optimal performance (see [16], [18]).

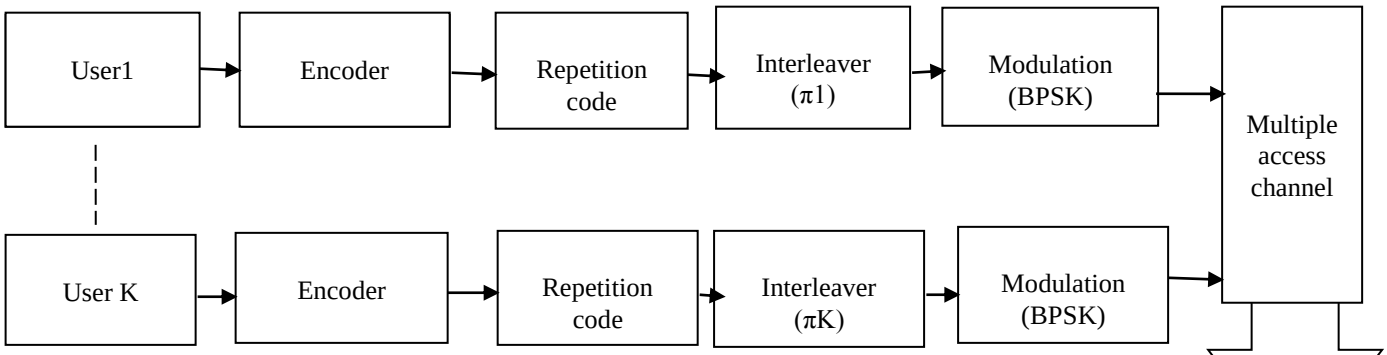

Fig 1. PS-CDMA transmitter diagram with effective repetition coding and interleaving of partitions.

The rest of the paper is organized as Section II, DS-CDMA system model along with the PS-based elements within an additive white Gaussian noise (AWGN) channel is presented. In Section III, PS-CDMA proposed hardware architecture is shown. Synthesis results for FPGA, for systems with up to 50-users, and synthesized circuits in 130 nm and 90 nm technologies are shown in Section IV. A comparison to other DS-CDMA systems using DSP and FPGA implementations is presented and in addition, comparisons to theory with respect to quantization effects are made.

The conclusion is in Section V.

II. SYSTEM MODEL

A. Transmitter

We consider a DS-CDMA system model using binary phase shift keying (BPSK) modulation. Fig. 1 shows the transmitter block diagram for each user. . At the transmitter encoded bits of user k, kϵ{1,…,K},q, qϵ{1,…,L} (d could be the result of high-rate LDPC encoding as this detector performs very well in conjunction with high rate codes [15]) are encoded by a repetition code (M,1) as part of the DSSS modulation operation. Typically M is chosen between 3–5 as larger values have limited additional effect on performance. The resulting coded bits, bj,k , jϵ{1…..Nm}, Nm = LM, are interleaved by a user-distinct interleaver Πk to obtain b'jk. Since the system uses DS-CDMA, the interleaved coded bits are spread by a user-distinct spreading code sj,k. These spreading codes have a shorten length, or processing gain of N/M, with respect to the baseline CDMA system use spreading sequences of length N. This normal vision is used to ensure fair power and spectral comparisons. Fig. 2 shows a small example of the relationship of data symbols and partitions for a single user, *L = 4, M = 2*.

Before transmission, each user has a power Pk assigned to it and its chips are normalized to the overall (uncoded) spreading gain of N, i.e., $c_k = 1/\sqrt{N}$. This gives the resulting individual transmissions at the coded, or partitioned, level as

$$\mathbf{y}_{j,k} = \sqrt{P_k} c_k b'_{j,k} \mathbf{s}_{j,k}, \quad j\epsilon\{1, \ldots, LN/M\} \qquad (1)$$

Where *Sj,k* = [*Sj,k,1,…, Sj,k ,N/M*] is the reduced length spreading sequence for user k and partition j.

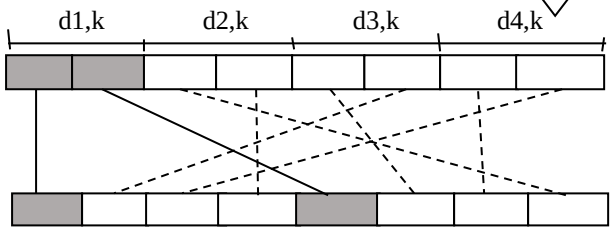

Fig 2. Interleaved partitions, π = [1 5 8 4 6 2 7 3], M=2, L=4

The system load $\alpha = K/N$ is a key parameter that compares performance of different systems. We show that for a given α, the full size of the system does not affect the bit error rate (BER) performance of the interference cancellation remarkably. The improvements of PS-CDMA over conventional CDMA can be seen with high system loads over and near $\alpha = 1$. In fact, the traditional use of a correlation receiver achieves spectral efficiencies of only around $\alpha = 0.1$ - 0.2, while the more advanced minimum mean-square error (MMSE) receiver can achieve values of without significant degradation [20]. The iterative receiver examined here shows nearly no degradation at values exceeding $\alpha = 1$.

B. Detector

All the user's signals *yj,k* aggregate in the AWGN channel, to give the received signal shown in (2), where n is the contribution of the additive white noise.

$$\mathbf{r}_j = \sum_{k=1}^{K} \mathbf{y}_{j,k} + n \qquad (2)$$

Simultaneousness is assumed here for simplicity, however, simulation results and the analysis in [16] show that this assumption is not critical for either performance or implementation complexity of the decoder itself.

The detector consists of K parallel individual data receivers, shown in Fig.3 with the single user's processor path outlined with a dashed box. The receivers decode their respective transmission embedded in *r* and then in parallel share their estimate of *yj,k* with each other to attain an interference reduced version of *r*. This means that all receivers share a common input **r** and an estimated channel aggregator at the chip level. .

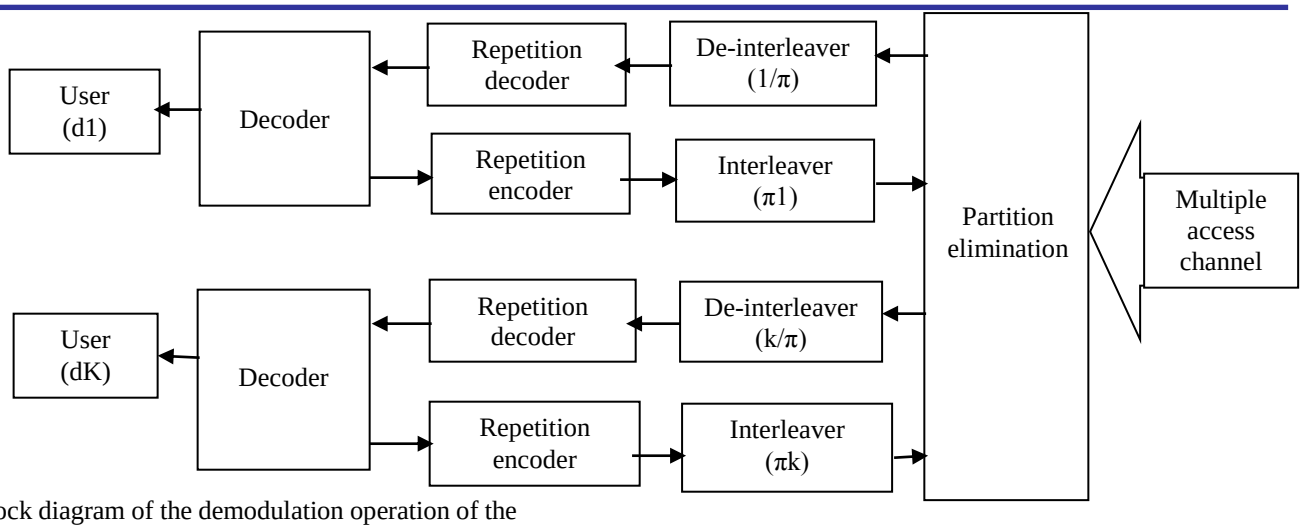

Fig.3.Block diagram of the demodulation operation of the PS-CDMA receiver.

Here the estimate of each received signal at the chip level is summed to approach an interference free estimate of the received signal at the ith iteration of processing. The first iteration has no estimates available and skips the cancellation stage.

We use a conventional matched filter to correlate the unique signature sequences sj,k with the incoming noisy waveform to give sufficient statistics of the user's signal to obtain an estimate of the user's transmitted partitions. We assume simultaneousness within the system for this implementation, but this is not an essential imperative.

Each received partition element after matched filtering with its individual spreading sequence is given as

$$\begin{aligned}\hat{b}'^{i}_{j,k} &= \sum_{n=1}^{\frac{N}{M}} \left(r_n - \hat{r}^{i}_{k,n}\right) s_{j,k,n} \\ &= \sum_{n=1}^{\frac{N}{M}} \left(\sum_{k=1}^{K} y_{j,k,n} + n - \sum_{\substack{k'=1 \\ k' \neq k}}^{K} \hat{y}^{i-1}_{n,k'} \right) s_{j,k,n} \\ &= \sum_{n=1}^{\frac{N}{M}} \left(y_{j,k,n} + I^{i-1}_{j,k,n} + n \right) s_{j,k,n} \end{aligned} \quad (3)$$

Equation (3) contains the contributing elements to the parti- tion estimates as the detector iterates. With adequate signal-to-noise ratio (SNR) the chip samples approaches the value of *yj,k,n* meaning the interference goes to zero with iterations.

The log-likelihood ratio (LLR) of *dk* at iteration *i* is computed as

$$\log\left(\frac{\Pr(d_k = 1)}{\Pr(d_k = 0)}\right) = \frac{2}{\sigma_0^2} \sum_{j=1}^{M} \hat{b}^{i}_{j,k} \quad (4)$$

where σ_0^2 is the variance of the interference and noise, which is shown in [22] to be Gaussian. Additionally to (4), log-likelihood ratios for each partitioned bit $b^{i}_{j,k}$ can be computed individually as the sum of the LLRs of all related partitions, i.e., as

$$\log\left(\frac{\Pr(b_{j,k} = 1)}{\Pr(b_{j,k} = 0)}\right) = \frac{2}{\sigma_0^2} \sum_{\substack{m=1 \\ m \neq j}}^{M} \hat{b}^{i}_{m,k} \quad (5)$$

From these LLR values, soft estimates of each partitioned bit can be computed as the expectation of (5), i.e., as

$$\tilde{b}^{i}_{j,k} = \tanh\left(\frac{1}{\sigma_0^2} \sum_{\substack{m=1 \\ m \neq j}}^{M} \hat{b}^{i}_{m,k}\right) \quad (6)$$

The argument in (6) is similar to the LLR addition that occurs in the iterative decoder of a low-density parity-check (LDPC) code, which aggregates these LLRs at its variable nodes. These variable nodes, as in this system, represent repetition codes.

In order to perform (6) the partitions are deinterleaved so that the summation occurs on partitions with the same origin. Using the user-distinct interleaver we can deinterleave the estimated partitions $\hat{b}'^{i}_{j,k}$ into their original natural order $\hat{b}^{i}_{j,k}$.

If we view each partition as a separate estimate of the original data symbol $d_{q,k}$ we can sum the *M* partitions and take the sign of the sum as the hard decision of the detector output. This sum can be reused in the extrinsic summation of (6) by adding all the partitions of a symbol and then subtracting the self-information to obtain the extrinsic estimate. This is analogous to the sum node processing of an LDPC decoder, but instead of parallel wires entering the summation we have serial partitions.

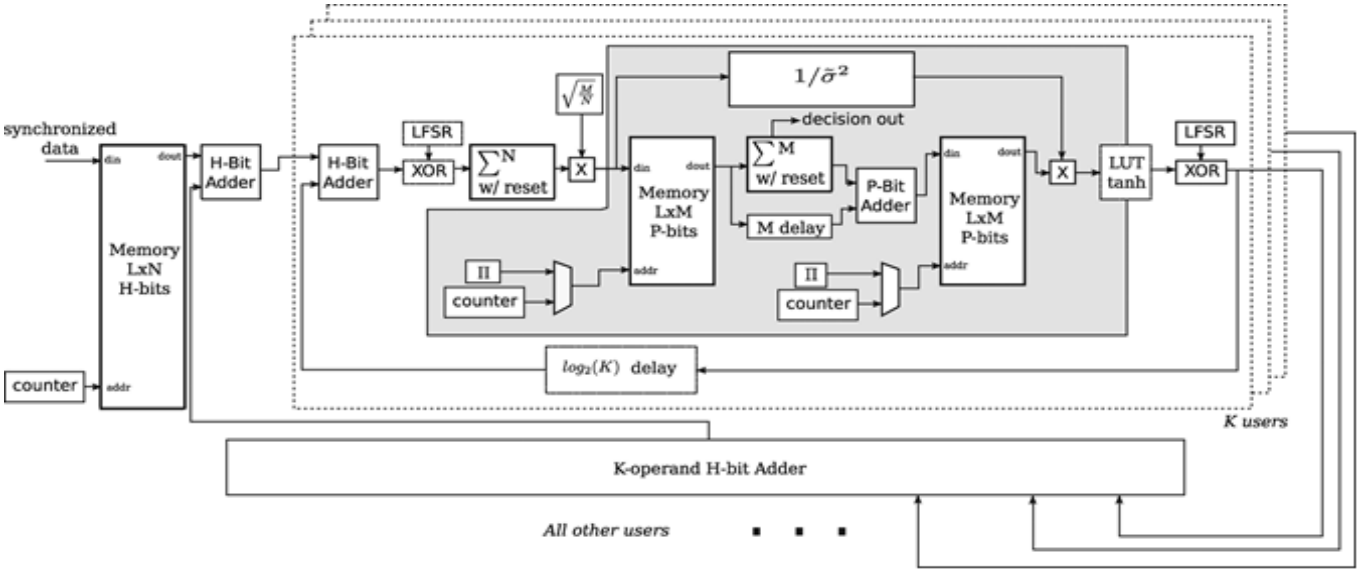

Fig 4. Architecture of the proposed iterative demodulator for PS-CDMA

We split (6) into two operations, the extrinsic estimate

$$b^i_{ext\ j,k} = \sum_{\substack{m=1 \\ m \neq j}}^{M} \hat{b}^i_{m,k} \tag{7}$$

and the variance-scaled tanh soft-bit operator:

$$\tilde{b}^i_{j,k} = \tanh\left(\frac{1}{\sigma_0^2} b^i_{ext\ m,k}\right) \tag{8}$$

Performing (7) and (8) sequentially gives the same result as (6).

In order to perform the partial interference cancellation of chips, we must recreate them identically to the user's respective transmitted signals. Using, the user-distinct interleaver and coding sequence an estimate of the interference signal for user *k* is generated as

$$\hat{\mathbf{r}}^i_k = \sum_{\substack{k'=1 \\ k' \neq k}}^{K} \sqrt{P_k} c_k \tilde{b}^{i}_{j,k} \mathbf{s}_{j,k} \tag{9}$$

and used in (3) for the next iteration.

All K users' chip estimates are aggregated and subtracted from the original received aggregate signal *rn*. If the estimates are perfect, the only values left are the contribution of noise to that chip. However, the interference cancellation has removed the user's own information, so that is added back in, which each user does within its respective receiver to attain a user specific version of the interference-reduced received signal. The system then iterates until MAI has been cancelled sufficiently to get acceptable BER. The effect of processing CDMA signals in this iterative fashion via partitioning the original DSSS signal into a few components is a dramatic increase in the achievable spectral efficiency with respect to conventional CDMA, as well as with respect to the at least equally complex MMSE receiver. These performance results are discussed in Section IV

III. SYSTEM ARCHITECTURE

In this section the design choices for the FPGA implementation are presented. Designs are implemented in VHDL and C++. The C++ program is used to plan design parameters for BER performance and perform bit-true calculations. In addition, the program is used to perform quantization analysis and its effect on BER. For the FPGA study Xilinx ISE 14.3 is used for synthesis compilation and Chip Scope Pro and operating system: windows 8 are the software requirements used.

Intel CORE, i5 and FPGA- SPARTAN-3 XC3400 are the hardware requirement specification. Fig. 4 shows the system architecture. All data paths use sign-magnitude representation and control signals are omitted. The elements within the dashed box fabricate the individual data receiver, while the contents within the shaded area assemble

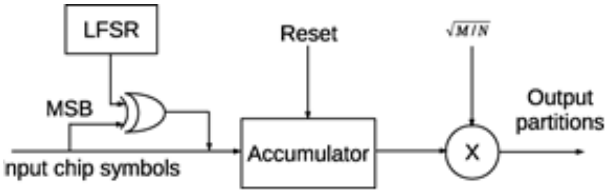

Fig.5. Matched filter block diagram.

the partition processing data path. The data paths each have a bit width or precision of P-bits or H-bits. All circuits within the shaded area use P-bits, otherwise circuits have

a precision of H-bits. Hard decisions for BER calculation are generated with the partition processing data path. The transmitted signal is stored in memory, using *L*N* entries each with H-bits. All receivers share this memory as the data is streamed to all K receivers simultaneously. After the memory, the H-bit adder is for interference cancellation. In the first iteration nothing is added or subtracted.

A. *Matched Filter*

Chip advent between all users is assumed to be synchronous, so the matched filter consists of an XOR gate, an accumulator (with reset), and a scaler. The scaler multiplies the partitions by $\sqrt{M/N}$. The XOR takes advantage of the sign-magnitude form by tying the output of the linear feedback shift register (LFSR) with the most significant bit (MSB) of the data. This avoids the need for multiple bit switching that two's complement would require. The design of the matched filter is shown in Fig. 5.

The rightmost bit of the LFSR is called the output bit. The taps are XOR'd sequentially with the output bit and then fed back into the leftmost bit. The sequence of bits in the rightmost position is called the output stream. The bits in the LFSR state which influence the input are called taps
The LFSR is a 51 stage delay line, with stages 1 and 4 tapped.
The value of each stage is programmed to a unique sequence for each user. The length and taps were chosen to supply a suitable pseudo-random number generator [23].

In the matched filter there is a transition between two processing domains, the chip processing domain and the partition processing domain. The chip processing domain involve a greater precision in data representation of H-bits, while the partition processing domain involve a less precise representation of P-bits. This occurs due to the soft processing that takes place in the tanh-limit, whereas the cancellation needs a precision that can handle fractions of transmitted amplitudes. After the matched filter the data takes two parallel paths. The first path forms estimates of the data and calculates hard decisions. The second path is used to form an estimate of the variance in the signal.

B. *Interleaver*

Interleaving, a technique for making forward error correction more robust with respect to burst errors. It has a depth of elements, and instead of storing the indices, they are calculated in order to reduce usage of memory resources. Low-density parity-check (LDPC) code is a linear error correcting code, a method of transmitting a message over a noisy transmission channel.

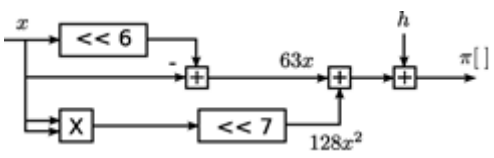

Fig 6. On-the-fly interleaver π

Turbo coding is an iterated soft-decoding scheme that combines two or more relatively simple convolutional codes and an interleaver to produce a block code that can perform to within a fraction of a decibel of the Shannon limit. The design can operate in parallel, but we chose a serial implementation shown in Fig. 6.

The permutation polynomial being calculated is

$$(63x + 128x^2 + h) \bmod (\mathrm{L} * \mathrm{M}). \qquad (10)$$

The value *x* in (10) is generated by a counter and only one multiplier is needed. The value *h* is introduced to create unique interleavers. The interleavers implemented in the FPGA rely on available block random access memory (BRAM). The sequential design introduces latency since the entire frame of partitions must be written in to memory before being read out. However, at the expense of another memory block and multiplexors the interleaver can be pipelined, since the writing index and reading index are generated separately.

C. *Partition Estimation*

In the previous section, Fig.3 contains the partition estimate module positioned between the deinterleaver and the interleaver. In the implementation, shown in Fig. 4, the partition estimation is split around the interleaver. The extrinsic step, (7), comes first followed by the tanh step, (8), after the interleaver. The tanh step of the estimation operation is separated from the extrinsic step in order to give the system a chance to develop an estimate on the variance statistic used to weight the tanh operation.
For the extrinsic step we use an accumulator and a delay element (flip-flops). A partition is delayed by *M* cycles so that the partitions of a symbol can be accumulated. Then the partition is subtracted from the sum, to obtain the extrinsic partition. This occurs for all *M* partitions in the delay pipeline. By looking at the MSB of the accumulator's sum we can also get the hard decision for the iteration.
For the tanh step, we use a small look-up table (LUT) with P-bit inputs and H-bit outputs. To avoid an extra multiplier, we have moved the scaling from the cancellation into the LUT.

D. *Variance calculation*

The variance is calculated assuming that the partitions of the matched filter give the correct hard decision, and is called the sliced variance [25]. It is one of the dimension-reduction methods. It targets the same differences but using a different arrangement for variances.

$$\frac{1}{\tilde{\sigma}^2} = \frac{2}{N_0} = \frac{LN}{\sum_{1}^{LN} \left(\hat{r_k} - \frac{1}{\sqrt{M}} \right)^2} \qquad (11)$$

The sliced variance operation is performed by a series of shifts with a multiplier and an accumulator followed by a look-up-table (LUT). The LUT performs the $1/x$ function to weight the extrinsic partitions as given in (6).

E. Cancellation

The final stage of the receiver creates an estimate of the transmitted signal. The P-bit result from the decision operation is sign-extended to an H-bit signal for the cancellation operation. Before cancellation, the chips are first scaled by $1/\sqrt{N}$ (performed in the tanh LUT) and then scrambled with the appropriate spreading sequence in order to align them with the transmitted values.

A large combinational component of the receiver is the common aggregator that sums all K-users estimates in order to cancel out interference. In this implementation a full-adder tree is used, which gives a $log_2(K)$ delay in each individual receiver to keep estimates aligned. The delay is implemented with a chain of $log_2(K)$ registers. The cycle-time cost of the adder tree could be improved by using specially design multi-operand adders [26] or using carry-save addition.

F. Register reuse

The Processing Unit contains shift register modules: the real and imaginary Coefficient Registers and the real and imaginary Data Registers. In addition to storage, these register modules order the data and coefficients according to the specific processing needs of the proto-Stallion modules. The Data Registers buffer the interface between the I/O-Control and Processing units, properly reusing the incoming data. Data and coefficient registers operate on the same operands. The register modules preserve the data and coefficients for the second operation. Finally, the Register modules implement the centering of the coefficients required by the adaptive tracking algorithms.

G. Minimization of fixed point scaling issues

The implementation of the receiver's algorithm must consider fixed point effects in order to optimize its results. In particular, use of fixed point numbers necessitates accommodation in three areas: coefficient storage, coefficient adaptation and decision module scaling.

Signal processing algorithms are often developed using infinite precision calculations; their implementations, however, must be limited precision. A peculiar problem arises when using fixed-point two's-compliment numbers for adaptive filtering. Digital systems typically employ truncation when reducing the precision of a result. The result of a two's-compliment truncation always reduces the value of the number, pushing it toward $-\infty$. Unfortunately, this produces two undesirable effects in adaptive filters. First, the coefficients are asymmetrically updated—coefficients that become more negative do so by a slightly larger amount than those that become more positive. Second, when the coefficients converge, the coefficients do not oscillate around their ideal values. Numbers smaller than the smallest representable scaled error value are represented as zero if they are positive, but as a small nonzero value when negative. The coefficients therefore adapt in only one direction, eventually losing convergence. The solution is to round rather than truncate; this restores the symmetry to the adaptation process.

The Decision module computes a series of multiplications; each of these multiplications produces a radix change in the result. The module must track these changes such that the resulting scaled error value is appropriately positioned for the coefficient update. The module maintains appropriate alignment by operating on a shifted product. Each 16-bit multiplication produces a 32-bit result, only 16 of which are needed for the next operation. Rather than always selecting the topmost bits, selecting a 16-bit window of less significant bits performs a left shift on the result, restoring the desired radix point position.

IV. Implementation Results

The implementation results for the FPGA test bed is summarized in this section. Logical equivalence between C++ simulations, VHDL simulations and the FPGA implementation was confirmed using ModelSim and a custom program that interfaces with a test circuit to calculate errors seen in the detector. Higher loaded systems can be operated using an exponential power distribution among the users as shown in (12), [16], with

$$P[k] = e^{\frac{\alpha k}{N}}; \quad \alpha = 0.5 \tag{12}$$

The systems under test use a partition gain of $M = 4$ with $L = 512$ symbols transmitted per packet. Each user has the same power, which is the worst-case scenario for this particular receiver. The overall system size does not affect the BER performance appreciably as long as the system load remains constant. However, as system load is increased, a higher SNR is required to achieve the same BER. In [27] it is shown that the error rate for partitioned signaling with large blocks, i.e., L goes to infinity, achieves an error rate given by

$$P_b = Q(\gamma_\infty)$$
$$\gamma_\infty^2 = \left(\frac{K}{N}\mathrm{E}\left[1-\tanh\left(\frac{M-1}{M}\gamma_\infty^2+\sqrt{\frac{M-1}{M}}\gamma_\infty\xi\right)\right]^2+\sigma^2\right)^{-1} \tag{13}$$

A. An FPGA Implementation

In Table I the FPGA synthesis test results are given for two different implementations on FPGAs. The test results for a 37-user system on a Virtex-II XCV28000 and a 50-user system on a Virtex-IV XC4VLX160 are described. Both FPGAs reside on Lyrtech development boards. Slice flip-flops are not highly utilized, since most slices contain logic that is not pipelined. The maximum aggregate throughput is calculated by (15) with $L = 512$, $N = 32$, I = 1 and $M = 4$.

Real time tests are performed with a DSP generated 80 MHz clock frequency and new throughputs of 43.5 Mb/s and 58.8 Mb/s respectively can be calculated as shown in (14) by setting $F = 80$ MHz, F is the system frequency and I is the number of iterations. The highest possible throughput is attained when $I = 1$, but if we consider various I, for example $I = 5$, the (useable) throughput for the 50-user system becomes 18.8 Mb/s. In addition, the system latency is determined by the number of iterations needed to reach the target BER.

If we assume BER= $1e^{-4}$, low system loads typically have $I < 5$ and for high system loads $I < 20$. Latency can be calculated by the denominator in (14) in units of cycles. Therefore a 50-user FPGA implementation requiring 5 iterations would have a latency of 108,544 cycles, which at 80 MHz is 1.3 ms.

$$T_{put} = \frac{K * L}{\underbrace{L * N + \underbrace{I * (L * M + L * N)}_{i>0}}_{i=0}} * F \tag{14}$$

The detector prepared for the FPGA has specific parame- ters that limit the number of users implementable. Each user requires two large memory blocks for serial interleaving. Each block contains the number of repeated partitions at the implemented bit-precision, LNP-bits, so for the results shown memory required per user ranges from 16,384-bits to 22,528-bits. The FPGA block RAM (BRAM) can be setup for several different data widths (from available primitives), but if the input data width is less than the BRAM width then BRAM bits are left unused. The implementation was chosen to handle various data precisions up to 16-bits. Therefore when P is 8 or 11, 8 to 5 bits are left unused respectively. In the case of $P = 8$ the memory blocks are 50% empty and present a significant under utilization of the BRAM. The logic breakdown per user is shown in Table II. A logic breakdown per module is shown in Table III.

The memory requirements for a single user is shown in Table II. In Table III the control module is a state-ma- chine and contributes significant overhead to the system. Two counters are used to control the operation of the design. A small counter is used to track the number of iterations and a larger counter is used to track the number of chips and partitions.

Sign-magnitude representation is used throughout the system, as this gives a simple change operation, XOR, within the matched filter module. Addition of sign-magnitude values is not as simple as two's complement, and significantly increases the logic utilization of the operation, since magnitude comparisons must be made before proceeding with the addition or subtraction. A conversion from sign-magnitude to two's complement representation should reduce overall logic utilization.

The FPGA results show that significant logic reductions can still be made within the control logic and cancellation operations. This indicates that implementation obtain more users and therefore a higher aggregate throughput. Another limitation to the design is the serial nature of the interleavers and their impact on memory usage and throughput. Each interleaver and de-interleaver adds LM cycles to each iteration, which lowers the throughput at a given number of iterations for successful cancellation.

From (11) the summation in the denominator occurs over the entire frame and requires a relatively large data path for the accumulator to avoid overflow. The feedback in the accumulator is on the critical path. In this implementation the entire variance block is optimized separately for timing driven analysis, since its output is ready long before it is needed.

TABLE I

Test results for PS-CDMA FPGA implementation

EXISTING			PROPOSED
IMPLEMENTATION PARAMETER	Virtex-II	Virtex-IV	Spartan 3
Number of slices	82%	84%	26%
Number of slice flip flops	36%	33%	23%
Number of 4 input LUTs	58%	62%	6%

TABLE II

Single user logic utilization on a virtex- IV & spartan 3

EXISTING			PROPOSED
FPGA Element	Virtex-IV utilization	Number occupied	Number occupied
Slice flip flops	0.511%	691	355
4-input LUT	0.931%	1,259	94
occupied slices	1.322%	894	205

TABLE III

Single user logic utilization contribution

PSCDMA Module	Slices	4-input LUTs
Control	18%	25%
Testing	26%	24%
Variance estimation	28%	26%
Processing	28%	25%

TABLE IV
Implementation of multiuser detection in DS-CDMA and DS-CDMA rake receivers

Reference	[9]	[30]	[31]	[32]	[33]	Existing	Proposed
Number of users (K)	15	4	8	16	5	50	8
Technology	FPGA XC2VP100	DSP ADSP2120 EZLAB	DSP TMS320 C6711	FPGA XCV400	DSP TMS320 C6416 DSK	FPGA XC4VLX160 Lyrtech	FPGA SPARTAN 3
Area	34,778 slices 69,557 flip-flops 48,312 4-LUTs	_	_	97,120 LE	_	67,584 slices 135,168 flip-flops 135,168 4-LUTs	768 slices 1536 flip-flops 1536 4-LUTs
Frequency	_		150MHz	_	600MHz	163MHz	343.389MHz
Power	_	_	_	_	_	_	27.34mw

V. SUMMARY

This paper presents a Multiuser detection also known as an iterative joint detection that exploits a priori knowledge about the spreading sequences and invokes the channel estimation, in order to remove multiple access interference using Direct Sequence- Code Division Multiple Access (DS-CDMA). It also discusses about an implementation of Interleave Division Multiple Access (IDMA) known as Partition Spreading Code Division Multiple Access (PS-CDMA) for high spectral efficiency, improved error performance and low receiver complexity. Decoding is performed using iterative methods from turbo and sum-product decoding. Results are computed for multiple Field Programmable Gate Array (FPGA). High throughput is achieved and an optimization of the blocks for different timing is performed. The control module, variance module and the cancellation modules are identified as logic blocks that could be reduced in their logic requirement, while the serial nature of the interleavers are identified as a bottleneck for the throughput and the major contributor to implementation area.

REFERENCES

[1] P. B. Rapajic and D. K. Borah, "Adaptive MMSE maximum likelihood CDMA multiuser detection," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 12, pp. 2110–2122, Dec. 1999.
[2] G. L. Turin, "An introduction to matched filters," *IRE Trans. Inf. Theory*, vol. 6, no. 3, pp. 311–329, Jun. 1960.
[3] R. Prasad and T. Ojanpera, "An overview of CDMA evolution toward wideband CDMA," *IEEE Commun. Surveys*, vol. 1, no. 1, pp. 2–29, 1998.
[4] S. Verdu, "Minimum probability of error for asynchronous Gaussian multiple access channels," *IEEE Trans. Inf. Theory*, vol. IT-32, no. 1, pp. 85–96, Jan. 1986.
[5] M. Li, C. A. Nour, C. Jego, and C. Douillard, "Design and FPGA prototyping of a bit-interleaved coded modulation receiver for the DVB-T2 standard," in *IEEE Workshop on Signal Processing Systems (SIPS) 2010*, Nov. 2010, pp. 162–167.
[6] E. Boutillon, C. Douillard, and G. Montorsi, "Iterative decoding of concatenated convolutional codes: Implementation issues," *Proc. IEEE*, vol. 95, no. 6, pp. 1201–1227, Jun. 2007.
[7] R. M. Buehrer and S. P. Nicoloso, "Comments on partial parallel interference cancellation for CDMA," *IEEE Trans. Commun.*, vol. 47, no. 5, pp. 658–661, May 1999.
[8] D. Divsalar, M. K. Simon, and D. Raphaeli, "Improved parallel interference cancellation for CDMA," *IEEE Trans. Commun.*, vol. 46, no. 2, pp. 258–268, Feb. 1999.
[9] A. O. Dahmane, L. Mejri, and R. Beguenane, "FPGA implementation of BP-DF-MPIC detectors for DS-CDMA systems in frequency selec- tive channels," in *2009 IEEE NEWCAS-TAISA*, Toulouse, France, Jun. 2009.
[10] P. Patel and J. Holtzman, "Analysis of a simple successive interfer- ence cancellation scheme in a DS/CDMA system," *IEEE J. Sel. Areas Commun.*, vol. 12, no. 5, pp. 796–807, Jun. 1994.
[11] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Statist. Soc., Ser. B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[12] J. A. Fessler and A. O. Hero, "Space-alternating generalized expecta- tion-maximization algorithm," *IEEE Trans. Signal Process.*, vol. 2, no. 10, pp. 2664–2671, Oct. 1994.

[13] L. B. Nelson and H. V. Poor, "Iterative multiuser receivers for CDMA channels: An EM-based approach," *IEEE Trans. Commun.*, vol. 44, no. 12, pp. 1700–1710, Dec. 1996.

[14] L. Ping, L. Liu, and W. K. Leung, "A simple approach to near-optimal multiuser detection: Interleave-division multiple-access," in *Proc. IEEE Wireless Communications and Networking Conf. (WCNC'03)*, Mar. 2003, vol. 1, pp. 391–396.

[15] L. Krzymien, D. Truhachev, C. Schlegel, and M. Burnashev, "Two-stage detection of partitioned random CDMA," *Eur. Trans. Telecommun.*, vol. 19, pp. 499–509.

[16] C. Schlegel, Z. Shi, and M. Burnashev, "Optimal power/rate alloca- tion and code selection for iterative joint detection of coded random CDMA," *IEEE Trans. Inf. Theory*, vol. 52, no. 9, pp. 4286–4294, Sep. 2006.

[17] M. Burnashev, C. Schlegel, W. A. Krzymien, and Z. Shi, "Analysis of the dynamics of iterative interference cancellation in iterative de- coding," *Problems of Information Transmission*, vol. 40, no. 4, pp. 297–317, 2004, (translated from Russian).

[18] C. Schlegel and D. Truhachev, "Multiple access demodulation in the lifted signal graph with spatial coupling," *arXiv:1107.4797, Subjects: Information Theory (cs.IT)*, 2011.

[19] J. G. Proakis, *Digital Communications*, 4th ed. New York: Irwin/Mc-Graw-Hill, 2001.

[20] C. Schlegel and A. Grant, *Coordinated Multiuser Communications*. New York: Springer, 2006.

[21] R. Dodd, C. Schlegel, and V. Gaudet, "Implementation of enhanced CDMA utilizing low complexity joint detection with iterative pro- cessing," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS 2010)*, May 2010, pp. 1763–1766.

[22] J. Boutros and G. Caire, "Iterative multiuser joint decoding: Unified framework and asymptotic analysis," *IEEE Trans. Inf. Theory*, vol. 48, no. 7, pp. 1772–1793, Jul. 2002.

[23] A. Alimohammad, S. F. Fard, and B. F. Cockburn, "Hardware-based error rate testing of digital baseband communication systems," in *Proc. IEEE Int. Test Conf.*, Oct. 2008, pp. 1–10.

[24] O. Y. Takeshita, "New deterministic interleaver designs for turbo codes," *IEEE Trans. Inf. Theory*, vol. 46, no. 6, pp. 1988–2006, Sep. 2000.

[25] M. C. Reed and J. Asenstorfer, "A novel variance estimator for turbo- code decoding," in *Proc. Int. Conf. Telecommunications*, 1997, pp. 173–178.

[26] K. Manochehri, S. Pourmozafari, and B. Sadeghian, "Very fast multi operand addition method by bitwise subtraction," in *5th Int. Conf. Information Technology: New Generations*, 2008, pp. 1240–1241.

[27] D. Truhachev, C. Schlegel, and L. Krzymien, "Low-complexity ca- pacity achieving two-stage demodulation/decoding for random matrix channels," in *Information Theory Workshop*, Lake Tahoe, CA, 2007, pp. 584–589.

[28] M. Martina, M. Nicola, and G. Masera, "Hardware design of a low complexity, parallel interleaver for WiMax duo-binary turbo decoding," *IEEE Commun. Lett.*, vol. 12, no. 11, pp. 846–848, Nov. 2008.

[29] R. Asghar, D. Wu, J. Eilert, and D. Liu, "Memory conflict analysis and implementation of a reconfigurable interleaver architecture supporting unified parallel turbo decoding," *J. Signal Process. Syst.*, vol. 60, no. 1, pp. 15–29, Jul. 2009.

[30] N. S. Correal, R. M. Buehrer, and B. D. Woerner, "A DSP-based DS-CDMA multiuser receiver employing partial parallel interfer- ence cancellation," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 4, pp. 613–630, Apr. 1999.

[31] M. D. Nisar, "Design and implementation of a DS-CDMA RAKE re- ceiver on TMS320 C6711 DSK and its performance analysis," in *Proc. Int. Multitopic Conf. (INMIC)*, Dec. 2004, pp. 271–277.

[32] X. Reves, A. Gelonch, and F. Casadevall, "Software radio implemen- tation of a DS-CDMA indoor subsystem based on FPGA devices," in Proc. 2001 IEEE Int. Symp. Personal, Indoor and Mobile Radio Com- munications, Sep. 2001, vol. 1, pp. D86–D90.

[33] M. Ahmed-Ouameur and D. Massicotte, "A real-time DSP-based it-erative (turbo) multiuser detection employing interference cancellor- MMSE," in Proc. IEEE North-East Workshop on Circuits and Systems, Jun. 2006, pp. 65–68.