Hindawi Publishing Corporation
EURASIP Journal on Applied Signal Processing
Volume 2006, Article ID 52919, Pages 1–12
DOI 10.1155/ASP/2006/52919

FPGA Implementation of an MUD Based on Cascade Filters for a WCDMA System

Quoc-Thai Ho, Daniel Massicotte, and Adel-Omar Dahmane

Laboratory of Signal and System Integration (LSSI), Department of Electrical and Computer Engineering, Université du Québec à Trois-Rivières, 3351 Boulevard des Forges, C.P. 500, Trois-Rivières, QC, Canada G9A 5H7

Received 2 October 2004; Revised 30 June 2005; Accepted 12 July 2005

The VLSI architecture targeted on FPGAs of a multiuser detector based on a cascade of adaptive filters for asynchronous WCDMA systems is presented. The algorithm is briefly described. This paper focuses mainly on real-time implementation. Also, it focuses on a design methodology exploiting the modern technology of programmable logic and overcoming the limitations of commercial tools. The dedicated architecture based on a regular structure of processors and a special structure of memory exploiting FPGA architecture maximizes the processing rate. The proposed architecture was validated using synthesized data in UMTS communication scenarios. The performance goal is to maximize the number of users of different WCDMA data traffics. This dedicated architecture can be used as an intellectual property (IP) core processing an MUD function in the system-on-programmable-chip (SOPC) of UMTS systems. The targeted FPGA components are Virtex-II and Virtex-II Pro families of Xilinx.

1. INTRODUCTION

The third generation (3G) of mobile wireless communication is adopted for high-throughput services and the effective utilization of spectral resources. This work focuses on Universal Mobile Universal Telecommunications systems (UMTS). In UMTS Systems, the wideband code-division multiple-access (WCDMA) scheme is adopted. The desired data throughputs for 3G UMTS systems are 144 kbps for vehicular, 384 kbps for pedestrian, and 2 Mbps for indoor environments [1, 2]. The receivers in 3G systems must take into account not only intersymbol interferences (ISI), but also more importantly multiple-access interferences (MAIs) which increase radically in the number of users and data rates. Multiuser detectors (MUDs) are applied to eliminate the MAI and become essential for an efficient 3G wireless network systems deployment [3]. The algorithmic aspect of MUD has become an important research issue over the last decade (e.g., [3–6]). Moreover, the real-time implementation aspect of MUDs is also well documented (e.g., [6–9]). The rapid prototyping targeted on field-programmable gate arrays (FPGAs) is also proposed [10–12]. These works demonstrate several limitations in practical systems in terms of timing and algorithm and hardware constraints (e.g., arithmetic complexity, memory access requirements, data flow) [5–7]. Moreover, no work was done to maximize the number of users on a chip (or a device in case of FPGAs). Maximizing the number of users makes it possible to increase the capacity of a cell and multiantenna processing.

Because minimum-mean-square-error (MMSE)-based receivers allow for a significant gain in performance, the adaptive two-stage linear cascade filter MUD (CF-MUD) based on MMSE receivers proposed in [13] offers a good tradeoff between performance and complexity. This algorithm presents a low-complexity and suitable regularity aspects for FPGA implementation. The CF-MUD is based on two blocks, signature and detection, which will be briefly described in Section 2. Each block acts as a filter in order to cancel the ISI and MAI. In previous works [14, 15], FPGA implementations of the signature block were presented. Based on the CF-MUD algorithm, this paper describes a complete design architecture targeted on the recent FPGA components—the Virtex-II and Virtex-II Pro of Xilinx including signature and detection blocks.

The rest of the paper is organized as follows. Section 2 presents a brief description of the system model and the adaptive MUD algorithm considered in this paper. Section 3 introduces the VLSI architecture of the present MUD targeted on the Virtex-II and Virtex-II Pro components. Section 4 describes the implementation methodology and Section 5 presents the results. Section 6 presents a few conclusions.

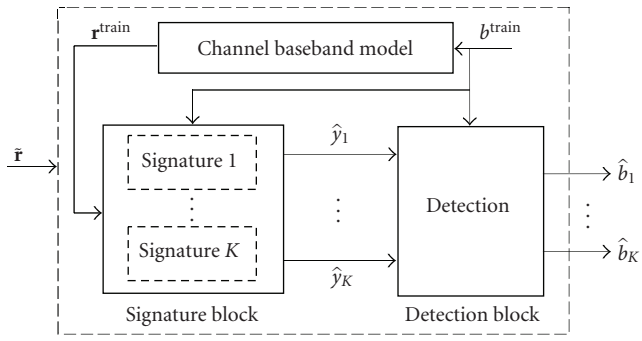


Figure 1: Principle of cascade filter MUD (CF-MUD).

2. BACKGROUND

2.1. DS-CDMA baseband model

In a direct-sequence CDMA (DS-CDMA) baseband system model, we consider K mobile users transmitting symbols from the alphabet $\Xi = \{-1, 1\}$. Each user's symbol is spread by a pseudonoise (PN) sequence of length N_c called the specific signature code. T denotes the symbol period and T_c denotes the chip period, where $N_c = T/T_c$ is an integer. User k's nth transmitted symbol is $b_k^{(n)}$.

The base transceiver station (BTS) received signal in baseband can then be written as follows:

$$\tilde{r}(t) = \sum_{n=0}^{N_b-1} \sum_{k=1}^{K} A_k b_k^{(n)} \sum_{l=1}^{L_k} h_{k,l}^{(n)} s_k^{(n)}(t - nT - \tau_{k,l}) + \eta(t), \tag{1}$$

where t denotes the time; L_k is the number of propagation paths; $h_{k,l}^{(n)}$ and $\tau_{k,l}$ are, respectively, the complex gain and the propagation delay of the path l for user k; N_b represents the number of the transmitted symbols, A_k is the transmitted amplitude of user k; $s_k^{(n)}$ is the specific signature of user k; and $\eta(t)$ is the additive white Gaussian noise (AWGN) with variance σ_η^2.

To increase the performance and capacity of communication systems, the ISI and MAI must be minimized. It is therefore essential to design MUD processing able to cancel these interferences. The following gives a brief description of the CF-MUD [13].

2.2. Cascade filter multiuser detector

The block diagram of the multiuser detector CF-MUD to be implemented on an FPGA is shown in Figure 1 [13].We can distinguish two blocks: signature and detection. Each block acts as an adaptive filter for canceling the ISI and MAI. The proposed linear adaptive MUD is based on the least-mean-square (LMS) adaptation method. This filter, however, needs data training sequences to adapt the filter coefficients. Compared to time-division multiple-access (TDMA) used in Global Systems for Mobile communications (GSM) systems, UMTS systems do not give access to preknown data with the exception of pilot bits—in order to adjust the filter coefficients. It is important to note that to assure the convergence, both block filters need more than the pilot bits available in fast-fading context. Preknown data training sequences $\tilde{\mathbf{r}}^{\text{train}}$ are internally generated based on channels parameters (amplitudes and delays) obtained from the channel-estimation technique.

The principle of CF-MUD is briefly described in Figure 2. The switch models the training phase and detection phase. The first block of the CF-MUD, the signature block, adapts the signatures of the users without prior knowledge of their PN codes. In the first step, we synchronized the received signal $\tilde{\mathbf{r}}(n)$ based on the estimated propagation delays for each user.

In the training phase, we used the following set of equations for user k ($k = 1, 2, \ldots, K$):

$$\hat{y}_k(n) = \Re\left(\hat{\mathbf{w}}_k(n)^H \tilde{\mathbf{r}}^{\text{train}}(n)\right), \quad \hat{\mathbf{w}}_k(0) = 0, \tag{2}$$

$$\alpha_k(n) = b_k^{\text{train}}(n) - \hat{y}_k(n), \tag{3}$$

$$\hat{\mathbf{w}}_k(n+1) = \hat{\mathbf{w}}_k(n) + \mu\, \tilde{\mathbf{r}}^{\text{train}}(n)\alpha_k(n)^*, \tag{4}$$

with $\hat{\mathbf{w}}_k(n) = [\hat{w}_{k,0}(n), \hat{w}_{k,1}(n), \ldots, \hat{w}_{k,N_c-1}(n)]^T$, and

$$\tilde{\mathbf{r}}(n) = \left[\tilde{r}(nT), \tilde{r}(nT - T_c), \tilde{r}(nT - 2T_c), \ldots, \tilde{r}(nT - (N_c - 1)T_c)\right]^T, \tag{5}$$

where $\dim(\hat{\mathbf{w}}_\mathbf{k}) = \dim(\tilde{\mathbf{r}}^{\text{train}}) = N_c \times 1$, $\Re(\bullet)$ defines the real part of complex value, $(\bullet)^H$ defines the Hermitian operation and $*$ the conjugate.

The following notations are used: $\hat{x}$ is the estimated value of x; $\hat{y}_k(n)$ is the adaptation output of user k; $\hat{\mathbf{w}}_k(n)$ is the vector of filter coefficients of user k; $b_k^{\text{train}}(n)$ is the synthetic transmitted training data sequence; $\tilde{\mathbf{r}}^{\text{train}}(n)$ is the synthetic received training data vector generated from the $b_k^{\text{train}}(n)$ transmitted through estimated channel parameters; $\alpha_k(n)$ is the adaptation error of the signature; and μ is the adaptation step of adaptive filters in the signature block.

The detection block aims to suppress the residual MAI and ISI based on the data of all users estimated using the output signal of the signature block. From all users, we formed a vector $\hat{\mathbf{y}}_\mathbf{T}(n)$ at the output of the signature block as follows:

$$\hat{\mathbf{y}}_\mathbf{T}(n) = \left[\hat{\mathbf{y}}_1(n-1), \ldots, \hat{\mathbf{y}}_K(n-1), \hat{\mathbf{y}}_1(n), \ldots, \hat{\mathbf{y}}_K(n), \hat{\mathbf{y}}_1(n+1), \ldots, \hat{\mathbf{y}}_K(n+1)\right]^T. \tag{6}$$

In the training phase, we used the following set of equations for user k (for $k = 1, 2, \ldots, K$):

$$\hat{o}_k(n) = \hat{\mathbf{v}}_{\mathbf{T}k}(n)^H \hat{\mathbf{y}}_\mathbf{T}(n), \quad \hat{\mathbf{v}}_{\mathbf{T}k}(0) = 0,$$
$$\beta_k(n) = b_k^{\text{train}}(n) - \hat{o}_k(n), \tag{7}$$
$$\hat{\mathbf{v}}_{\mathbf{T}k}(n+1) = \hat{\mathbf{v}}_{\mathbf{T}k}(n) + \nu \hat{\mathbf{y}}_\mathbf{T}(n)\beta_k(n)^*,$$

where $\hat{\mathbf{v}}_{\mathbf{T}k}(n) = [\hat{v}_{1,k}(n), \hat{v}_{2,k}(n), \ldots, \hat{v}_{3K,k}(n)]^T$, $\dim(\hat{\mathbf{v}}_{\mathbf{T}k}(n)) = \dim(\hat{\mathbf{y}}_\mathbf{T}(n)) = 3K \times 1$, $\hat{o}_k(n)$ is the adaptation output of user k corresponding to the output of the respective adaptive filter, $\hat{\mathbf{v}}_{\mathbf{T}k}(n)$ is the filter coefficient vector of user k, $\beta_k(n)$ is the adaptation error of detection, and ν is adaptation step of adaptive filters in the detection block.

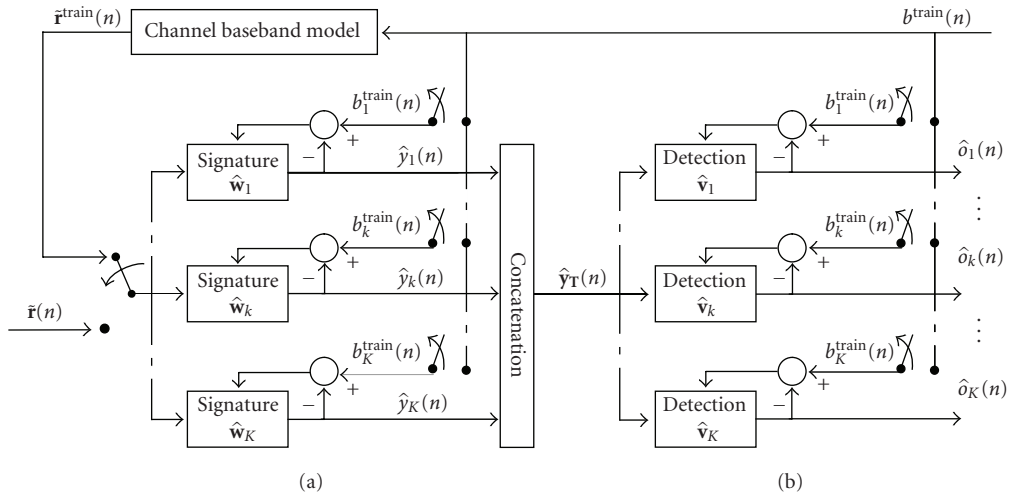


Figure 2: Principle of (a) signature block and (b) detection block for the kth user.

In the detection phase, the transmitted data of mobile users are estimated by the signature block from following equation:

$$\hat{y}_k(n) = \Re\left(\hat{\mathbf{w}}_k(n)^H \tilde{\mathbf{r}}(n)\right), \quad \text{for } k = 1, 2, \ldots, K. \tag{8}$$

Regarding the detection block, the transmitted data of users are estimated by the following equation:

$$\hat{o}_k(n) = \hat{\mathbf{v}}_{\mathbf{T}k}(n)^H \hat{\mathbf{y}}_{\mathbf{T}}(n), \quad \text{for } k = 1, 2, \ldots, K. \tag{9}$$

Finally, the estimated bits $\hat{b}_k(n)$ are found by simply taking the sign function of $\hat{o}_k(n)$,

$$\hat{b}_k(n) = \text{sign}\left(\hat{o}_k(n)\right). \tag{10}$$

When the adaptation process was completed, we applied (8), (9), and (10) to propagate the signal $\tilde{\mathbf{r}}(n)$ through the CF-MUD.

2.3. Performance evaluation of CF-MUD

Figure 3 depicts algorithmic performance in terms of the block error rate (BLER) of CF-MUD algorithm compared with the RAKE receiver and soft multistage parallel interference canceler (Soft-MPIC) in a WCDMA platform [3]. Simulation results were done for one antenna, in perfect channel estimation, Vehicular A channel defined by International Telecommunication Union (ITU) [16] 3 km/h mobile speed, 64 kbps data rate, and 15 users. We observed a gain of 1.9 dB to target a BLER of 10% for CF-MUD compared with Soft-MPIC and the RAKE receiver cannot reach the BLER of 10%. No decision feedback has been considered for CF-MUD and Soft-MPIC. Although MUD with decision feedback is considered superior than without the decision feedback creates a serious data dependency to parallelize the implementation on many devices.

Based on CF-MUD equations (2)–(10), the proposed FPGA-targeted architecture can be described as in Section 3.

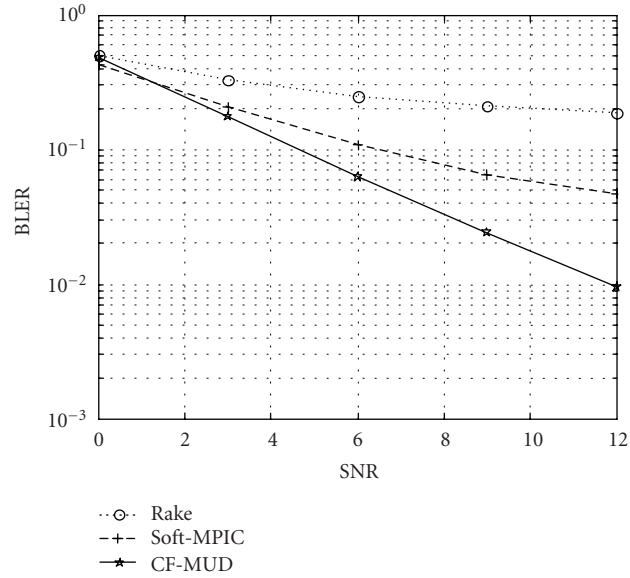


Figure 3: A performance evaluation of MUD methods in the WCDMA conditions with vehicular A channel at mobile speed 3 km/h, data rate 64 kbps (OVSF = 16), and 15 users in terms of BLER.

3. VLSI ARCHITECTURE TARGETED ON FPGA

The developed architecture should be reconfigurable to several baseband processing UMTS systems characterized by the number of users K and different communication scenarios in different mobile speeds. Thus, it can be reconfigured by respecting WCDMA, hardware, and algorithmic constraints. The main WCDMA constraints [2] are data rates, that is, orthogonal variable spreading factor (OVSF) of 64, 16, 8, or 4 corresponding, respectively, to 12.2 kbps (voice rate), 64 kbps, 144 kbps, and 384 kbps data rates; a time frame of 38400 chips in 10 milliseconds; and a mobile speed of 3 km/h to 100 km/h.

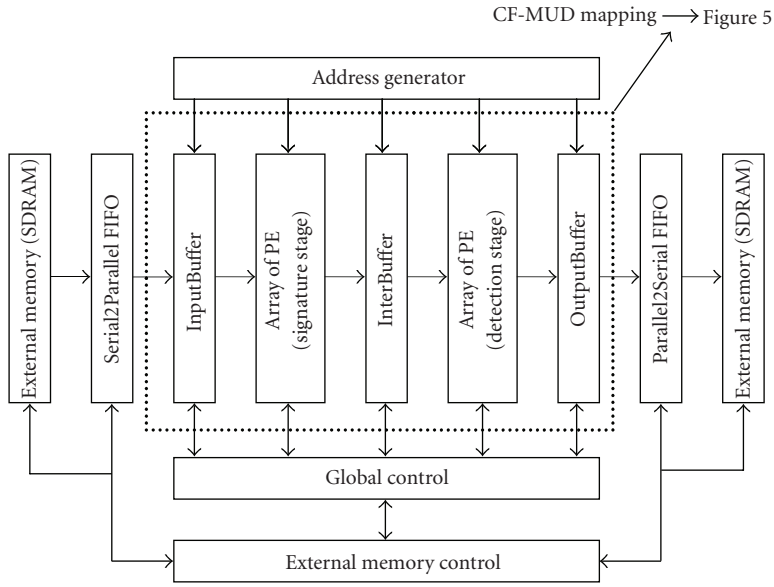


Figure 4: Simplified HW architecture of CF-MUD.

The main algorithmic constraints, with respect to MUD performance, consist of the number of adaptation iterations in the signature filter and detection filter, adaptation steps μ and ν, quantification scales to respect the arithmetic precision in fixed point.

The main hardware constraints take into account the limitations of targeted FPGAs in term of number of dedicated multipliers, number of block RAMs (BRAMs), and memory size of each BRAM [17].

These constraints were also used in our method of resource estimation before synthesis. The architecture must be able to respect real-time constraints bounded by time frame to detect all data frames, and by adaptation time to adapt all coefficients ($\hat{\mathbf{w}}$ and $\hat{\mathbf{v}}$) depending on the mobile speed.

The block diagram of the pipelined architecture is based on two stages of the modular array structure of processing elements (PEs) shown in Figure 4. Figure 5 illustrates the mapping of CF-MUD algorithm on array of PEs and internal memories (inside the FPGA). These PEs consist of optimized cores performing adaptive filtering defined by (2)–(4) which we called $\mathrm{PE}_{\mathrm{LMS}}$ including straightforward filtering defined by (2) which we called $\mathrm{PE}_{\mathrm{FIR}}$. The regularity of the CF-MUD makes it possible to time multiplex a number of users, that is, we used only one PE to process a number of users by time multiplexing selection. The time multiplexing, that is, number of users per PE, in the signature and detection blocks is defined by $T_{\mathrm{MUX}\,1}$ and $T_{\mathrm{MUX}\,2}$, respectively. Thus, the number of $\mathrm{PE}_{\mathrm{LMS}}$ and $\mathrm{PE}_{\mathrm{FIR}}$ inside each block is the same, and is represented by $N_{\mathrm{MUX}\,1}$ and $N_{\mathrm{MUX}\,2}$ for the signature and detection blocks, respectively. All PEs consider normalized-fixed complex-value signals and use the same time multiplexing.

The data and address paths are independent to permit maximum simultaneous direct access to data and address. Two different external memories SDRAM and two different memory buffers (*InputBuffer* and *OutputBuffer*) are used to allow independent access to input/output, and thus to maximize the multiple path access to external input/output. These memory buffers are implemented by the LUT (lookup-table) -based distributed memory of FPGAs. The memory buffers *InputBuffer* and *OutputBuffer* are multiport. The buffer *InternalBuffer* is used to memorize intermediate results from the signature filter and input to the detection filter. It is implemented by LUT-based distributed memories. The first-in first-out (FIFO) buffers *Serial2Parallel* and *Parallel2Serial* are used to minimize the utilization of input-output (IO) pins of FPGA and also to minimize the number of external memories. These buffers are implemented by LUT-based distributed memory of FPGAs as well. The PE of the architecture uses the semiglobal internal BRAM-based memories, that is, a certain number of PEs have access to the same memory. This number is defined by the possible time multiplexing determined from the architectural specification step.

We used an advanced scheduling based on time multiplexing by modifying the conventional methods, that is, As Soon As Possible (ASAP) and As Late As Possible (ALAP). This advanced scheme relies on the fact that ASAP gives low latency while ALAP gives high latency but uses less hardware resources [18]. Modifying jointly these two methods permits to balance the latency while exploiting the particular features of targeted FPGAs. The constraints of this scheduling involve using only two real dedicated multipliers and minimum number of multiplexers and other arithmetic operators (adders). This method exploits the symmetric structure of these FPGA components, especially the shared connection between BRAMs and the dedicated multipliers. Using two real multipliers to implement complex multiplication including four real multiplications permits to use this shared connection between dedicated multiplier and BRAM. Minimizing the number of multiplexers leads to a reduction in the critical path of circuit.

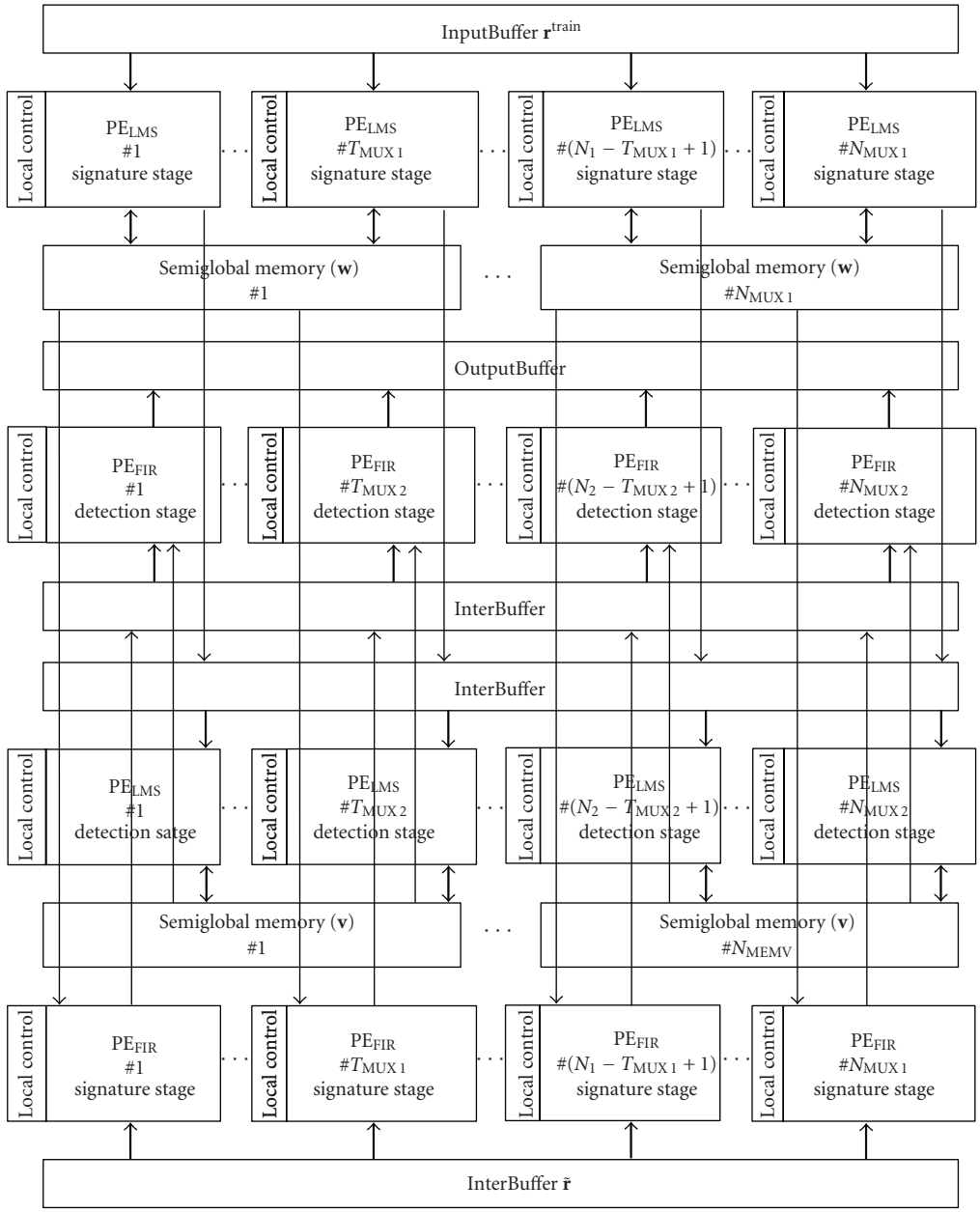


Figure 5: Mapping the CF-MUD on processing elements and internal memories.

The fine-grain pipeline of PEs, shown in Figure 6(a), uses dedicated 2-level pipelined multipliers available on the silicon die of Xilinx FPGA devices. To understand the PE functionality, consider the complex-number multiplication described by (2) as follows. The summation is up to N_T, which is N_C for signature filters and $3K$ for detection filters:

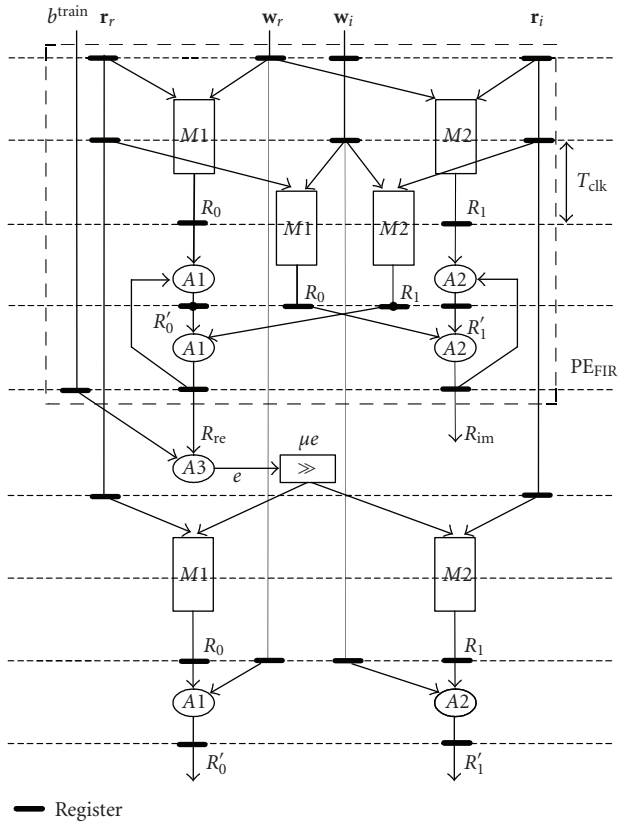
$$R_{\text{re}} = \sum_{i=0}^{N_T-1} \left(\Re\left(\tilde{r}_{k,i}^{\text{train}}\right)\Re\left(\hat{w}_{k,i}\right) - \Im\left(\tilde{r}_{k,i}^{\text{train}}\right)\Im\left(\hat{w}_{k,i}\right)\right),$$
$$R_{\text{im}} = \sum_{i=0}^{N_T-1} \left(\Re\left(\tilde{r}_{k,i}^{\text{train}}\right)\Im\left(\hat{w}_{k,i}\right) + \Im\left(\tilde{r}_{k,i}^{\text{train}}\right)\Re\left(\hat{w}_{k,i}\right)\right). \tag{11}$$

And to update the coefficients of (3) in (4),

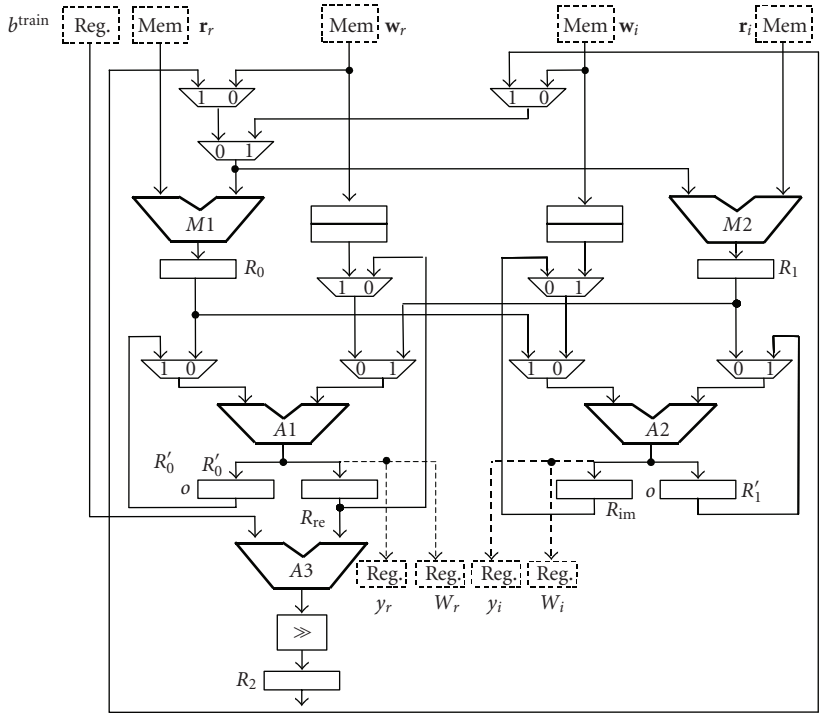
$$\Re\left(\hat{\mathbf{w}}_{k,i}(n+1)\right) = \Re\left(\hat{\mathbf{w}}_{k,i}(n)\right) + \mu\left(b_{k,i}^{\text{train}}(n) - R_{\text{re}}\right)\Re\left(\tilde{\mathbf{r}}_{k,i}(n)\right),$$
$$\Im\left(\hat{\mathbf{w}}_{k,i}(n+1)\right) = \Im\left(\hat{\mathbf{w}}_{k,i}(n)\right) + \mu\left(b_{k,i}^{\text{train}}(n) - R_{\text{im}}\right)\Im\left(\tilde{\mathbf{r}}_{k,i}(n)\right), \tag{12}$$

where $\Re(x)$ and $\Im(x)$ define the real and imaginary parts of x, and R_{re} and R_{im} represent the accumulation registers for real and imaginary parts.

Figure 6(b) illustrates the scheduling and register-transfer logic (RTL) mapping of PE_{LMS}, including PE_{FIR}, to implement the complex-number filter using two real-number multipliers, where Ax and Mx ($x = 1, 2, 3$) are, respectively,



(a)



(b)

Figure 6: Detailed description of a PE: (a) scheduling and (b) mapping of 2-level pipelined complex taps adaptive FIR-LMS filters.

the adder and the multiplier units. Unit $A1$ is an adder-subtracter that is used for addition or subtraction in the real part of (2). Unit $A3$ is subtracter operation that is used to calculate the error adaptation in (3). Saturation is used at the output of these operational units to maintain the length of the data bus. In this figure, the subscripts "r" and "i" represent the real and the imaginary parts of the variables, respectively. Registers R_{re} (R_{im}) and $R_0{}'$ ($R_1{}'$) correspond to $\Re(w_{k,i}(n))$ ($\Im(w_{k,i}(n))$) and $\Re(w_{k,i}(n+1))(\Im(w_{k,i}(n+1)))$, respectively. Registers R_0 (R_1) are used as pipelined registers allowing for two concurrent additions in multiplier-accumulator (MAC) and complex multiplications in (2), (4). Two registers are added before inputs of adders Ax to pipeline without hazard. The IO of PE can be registered or not. The fact that IO can be registered or not helps the processor to interface with other components of the system. The shift-to-right operation is represented by $\gg$. This shift operation allows to implement the hardware-free multiplication by adaptation step μ and ν whose value are of 2^{-n}.

The execution time of an adder is one clock cycle (T_{clk}) and that of a multiplier is 2 cycles. Regarding N complex taps filters, the throughput in terms of clock cycles of adaptation process is $(2N+5)$ and of detection process is $(2N+4)$. Thus, the throughput for the PE_{LMS} (including adaptation process and detection process) and PE_{FIR} (including detection process only) of are, respectively, $(3N+9)$ and $(2N+5)$. As a result, the throughputs of signature block and detection block are, respectively, $(3N_C+9)$, $(2N_C+5)$ and $(9K+9)$, $(6K+5)$.

The coarse-grain pipeline data-flow strategy in the system level of the architecture is detailed in Figures 7 and 8 for the adaptation and detection processes, respectively. The strategy depends on the processing time between signature block, detection block, and the adaptation and detection processes.

4. IMPLEMENTATION METHODOLOGY

This paper focuses on the hardware (HW) design flow of the MUD based on a library of the hard optimized IP cores; for example, complex-taps FIR filters used as PE for the adaptive MUD. It is necessary to estimate the timing performance and HW resources required by architectures from the architectural specifications satisfying these constraints. To reach the maximum number of users (K) for two family devices of Xilinx, a program based on nonlinear integer-programming model was developed. This nonlinear integer-programming is resolved by the branch-and-bound method [19]. The nonlinear integer-programming model makes it possible to estimate the performance requirements and the limitations of FPGA HW resources. This tool is used to maximize the time multiplexing (number of users in one PE) and timing performance (number of clock cycles) of the system, while respecting algorithmic constraints and HW resource limitations (number of multipliers and RAM block). It is also necessary to minimize the clock rate for power consumption. The program is helpful for choosing a type of suitable architecture in terms of pipeline strategy for the algorithmic specification of MUD. This tool can also be conversely used to estimate the necessary HW resources and timing performance.

For the specific developed architecture of the CF-MUD algorithm targeted on these FPGA devices (Virtex-II Pro and Virtex-II), the objective functions are to maximize the number of users K^{MAX} described by the nonlinear inequalities as follows:

$$K \le f\left(t, N_{\text{MEM}}, T_{\text{MUX}\,1}, T_{\text{MUX}\,2}, \text{OVSF}, N_{\text{chip}}, N_m, N_{A2}, N_{\text{cycle}}\right). \tag{13}$$

Respecting the following constraints,

$$T_{\text{MUX}\,1} \le g\left(t, N_{\text{MEM}}, \text{OVSF}, N_{\text{chip}}, N_{A1}, N_{\text{cycle}}\right) \tag{14}$$

and $T_{\text{MUX}\,2}$ is an integer satisfying the pipeline strategy of the HW architecture.

Where N_{MEM} is the number of data by BRAM, N_{chip} is the number of chip, N_m is the maximum number of dedicated multipliers available on silicon die of these FGPA components [17], N_{cycle} is the number of cycle (throughput) to solve the CF-MUD on FPGA (Section 3), and N_{A1} and N_{A2} are the number of adaptation iterations in the signature and detection block, respectively. We consider that the variables N_{A1}, N_{A2}, OVSF, and t are constraints. These above inequalities defined by straightforward functions $f(\bullet)$ and $g(\bullet)$, from (13) and (14), are built by taking constraints stated on Section 3 and the dedicated FPGA architecture.

Since verification is critical in the design flow, dynamic verification by simulations is used throughout. The results of fixed-point simulations high-level language (Matlab) provide a static functional reference for the HW verification of the architecture. The synthesized data are used for the verification in Matlab as well as in FPGA devices implementation.

5. RESULTS

HW architecture is targeted on the Virtex-II and Virtex-II Pro components of Xilinx to satisfy different algorithmic and WCDMA specifications in real time.

Tables 1 and 2 summarize the maximum number of simultaneous users (K^{MAX}) that can be processed in monorate on different devices of the Virtex-II and Virtex-II Pro families in different data based on the UMTS 3G standard. The data throughputs are fixed by the OVSF parameter such as 64, 16, 8, and 4 corresponding, respectively, to 12.2 kbps (voice rate), 64 kbps, 144 kbps, and 384 kbps (the last three throughputs are for data) [2]. We assumed three mobile speeds: slow fading (T_A = 40 milliseconds), medium fading (T_A = 10 milliseconds), and fast fading (T_A = 2 milliseconds), where T_A represents the allowed adaptation time of CF-MUD coefficients ($\hat{\mathbf{w}}$ and $\hat{\mathbf{v}}$) [20]. Considering the short code of 256 chips, the number of adaptation iterations is 100(256/OVSF) for each user k of the signature and detection block. We used the same number of adaptation iterations for hardware estimation.

While the allowed adaptation time constraint varies with the mobile speed, the allowed detection time is always limited by 10 milliseconds, which is the timing length of a frame

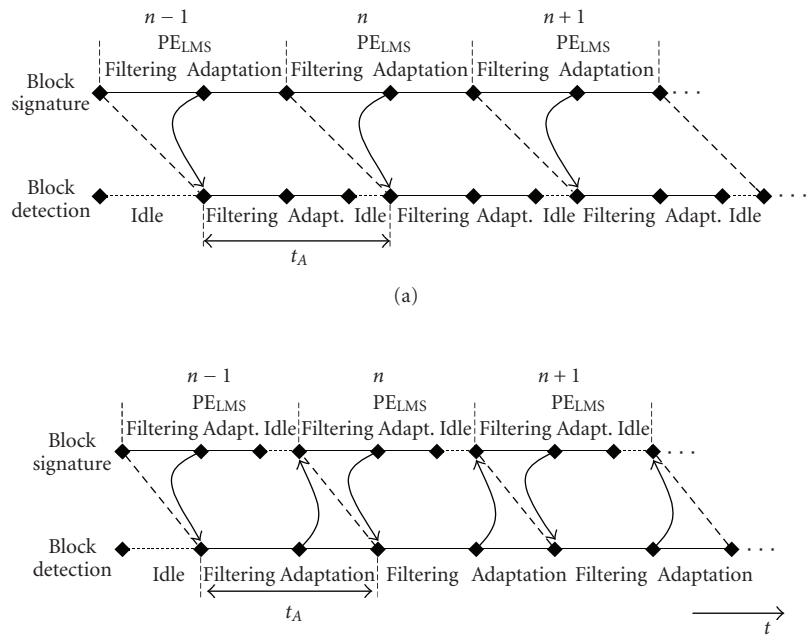


Figure 7: Pipeline strategy of adaptation process in case that the processing time of signature block is (a) superior and (b) inferior to the processing time of detection block.

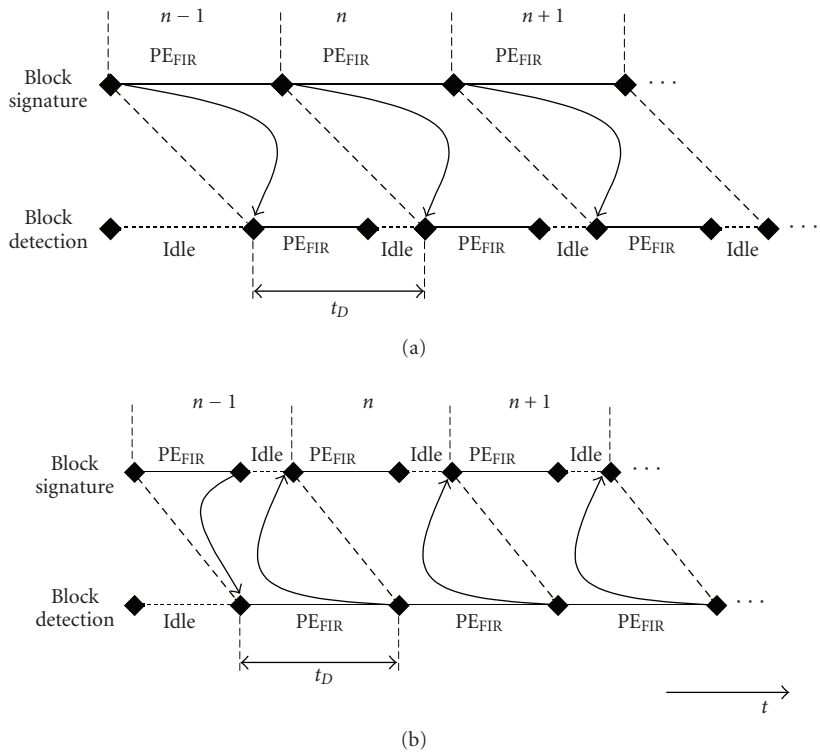


Figure 8: Pipeline strategy of detection process in case that the processing time of signature block is (a) superior and (b) inferior to the processing time of detection block.

Table 1: Maximum number of simultaneous users ($\hat{K}^{\mathrm{MAX}}$) detected and which can be integrated on different devices of Virtex-II Pro family.

Device	OVSF											
	Slow fading				Medium fading				Fast fading			
	64	16	8	4	64	16	8	4	64	16	8	4
XC2VP2	10	10	8	6	10	6	6	4	4	2	2	1
XC2VP4	22	20	16	14	20	14	10	6	10	4	2	2
XC2VP7	30	28	24	18	28	18	14	8	12	6	4	2
XC2VP20	52	48	36	28	48	28	16	16	22	12	4	2
XC2VP30	68	68	44	32	68	32	26	16	26	12	4	2
XC2VP40	84	82	64	38	82	38	32	16	32	12	4	2
XC2VP50	98	90	68	46	90	46	32	16	38	12	4	2
XC2VP70	112	108	68	64	108	64	32	16	54	12	4	2
XC2VP100	148	136	88	68	136	68	32	16	54	12	4	2
XC2VP125	170	136	110	68	136	68	32	16	54	12	4	2

Table 2: Maximum number of simultaneous users ($\hat{K}^{\mathrm{MAX}}$) detected and which can be integrated on different devices of Virtex-II family.

Device	OVSF											
	Slow fading				Medium fading				Fast fading			
	64	16	8	4	64	16	8	4	64	16	8	4
XCV40	2	2	2	2	2	2	2	2	1	1	0	0
XCV80	6	6	6	4	6	4	4	2	2	2	1	1
XCV250	18	18	16	12	18	12	10	6	8	4	2	2
XCV500	24	22	18	16	23	16	10	6	10	4	4	2
XCV1000	28	26	22	16	25	16	12	8	12	6	4	2
XCV1500	34	32	26	20	32	20	16	8	14	8	4	2
XCV2000	36	34	28	22	34	22	16	10	16	9	4	2
XCV3000	56	52	40	32	52	32	19	16	24	12	4	2
XCV4000	66	60	44	32	60	32	24	16	26	12	4	2
XCV6000	72	68	48	32	68	32	28	16	26	12	4	2
XCV8000	84	72	56	32	72	32	32	16	28	12	4	2

of 38400 chips in UTMS systems. To estimate the maximum number of users $\hat{K}^{\mathrm{MAX}}$, we assumed a 100 MHz clock frequency for all devices.

Tables 3 and 4 summarize the utilization ratio of resources on targeted devices corresponding to the estimated maximum number of users given in Tables 1 and 2, respectively. We observed that the utilization ratio of resources in case of fast-fading scenario is low (indicated in gray zones). This is because the adaptation time decreases an impose to fix $T_{\mathrm{MUX}\,1}$ and $T_{\mathrm{MUX}\,2}$ to equal 1. Thus, we are limited by few resources. But we can easily increase the number of users by only duplicating the same architecture on the device. Hence, we can easily increase K^{MAX} in fast-moving conditions.

Note that in these results, the users transmit simultaneously in the same sector. Normally, we should consider the number of user lower than the value of the OVSF. Thus, the number of user higher than the value of the OVSF should be distributed on the other sectors of the BTS. Under these conditions, the number of users by BTS (3 sectors) should be higher than the data indicated in Tables 1 and 2.

According to the pipeline strategy of developed architectures, the total time needed to process a data frame is restricted by the maximum execution time in the signature and detection blocks. In the signature block, the performance in terms of adaptation time (t_{A1}) and detection time (t_{D1}) is, respectively, defined by

$$t_{A1} = (3N_C + 9)N_{A1}\left(\frac{256}{\mathrm{OVSF}}\right)T_{\mathrm{MUX}\,1}T_{\mathrm{clk}},$$
$$t_{D1} = (2N_C + 5)\left(\frac{38400}{\mathrm{OVSF}}\right)T_{\mathrm{MUX}\,1}T_{\mathrm{clk}}. \tag{15}$$

In the detection block, we have

$$t_{A2} = (9K + 9)N_{A2}T_{\mathrm{MUX}\,2}T_{\mathrm{clk}},$$
$$t_{D2} = (6K + 5)\left(\frac{38400}{\mathrm{OVSF}}\right)T_{\mathrm{MUX}\,2}T_{\mathrm{clk}}. \tag{16}$$

With the pipeline strategy of architecture, the time processing in each cascade filter is, respectively, $\max(t_{A1}, t_{D1})$ and $\max(t_{A2}, t_{D2})$, and it needs to be inferior to T_A for adaptation depending on slow-, medium-, and fast-fading communication situations.

Table 5 summarizes the results of an experiment system for 16 users after routing and placing by the Xilinx physical tool (the ISE foundation) on the Virtex-II Pro component XC2VP30. The results for the data rate in fast-fading conditions are excluded for the system of 16 users because of the

Table 3: Utilization ratio of hardware (%) for $\hat{K}^{\mathrm{MAX}}$ of Table 1 on different devices of Virtex-II Pro family.

Device	OVSF											
	Slow fading				Medium fading				Fast fading			
	64	16	8	4	64	16	8	4	64	16	8	4
XC2VP2	93	97	98	88	97	88	100	89	79	83	83	39
XC2VP4	100	100	95	100	100	100	95	100	100	71	57	36
XC2VP7	96	95	98	95	95	95	95	97	98	95	68	23
XC2VP20	98	98	98	99	98	99	97	97	100	82	34	11
XC2VP30	90	100	88	97	100	97	99	94	76	70	22	7
XC2VP40	89	100	100	99	100	99	92	67	100	50	16	5.2
XC2VP50	100	92	98	99	92	99	85	55	98	41	13	4.3
XC2VP70	92	100	83	99	100	99	80	39	99	29	9.1	3.0
XC2VP100	100	92	99	92	92	92	59	29	97	22	6.7	2.2
XC2VP125	92	98	100	98	98	98	47	23	78	17	5.3	1.7

Table 4: Utilization ratio of hardware (%) for $\hat{K}^{\mathrm{MAX}}$ of Table 2 on different devices of Virtex-II family.

Device	OVSF											
	Slow fading				Medium fading				Fast fading			
	64	16	8	4	64	16	8	4	64	16	8	4
XCV40	78	80	84	93	79	93	85	90	54	67	0	0
XCV80	95	98	91	85	98	85	100	88	86	75	58	58
XCV250	98	96	100	90	96	90	97	97	89	83	67	42
XCV500	96	98	99	100	98	100	83	88	87	94	100	31
XCV1000	99	98	92	99	98	99	98	100	90	90	75	25
XCV1500	99	100	98	97	100	97	100	100	97	100	62	21
XCV2000	95	98	100	92	98	92	95	98	95	96	54	18
XCV3000	97	99	100	100	99	100	99	100	100	100	31	10
XCV4000	99	100	100	92	100	92	100	80	87	80	25	8.3
XCV6000	100	94	100	92	94	92	97	89	72	7	21	6.9
XCV8000	100	100	100	79	100	78	98	76	100	57	18	6.0

Table 5: Postplacing and routing results using Xilinx physical tools (ISE Foundation) targed on Xilinx Virtex-II Pro XC2VP30 device for a system of $K = 16$ users for slow- and medium-fading conditions.

	OVSF	T_{MUX}		Slices	BRAM	Multipliers	Clock rate (MHz)	Clock skew (ns)	t_A (ms)	t_D (ms)
		$T_{\mathrm{MUX}\,1}$	$T_{\mathrm{MUX}\,2}$							
Slow fading	64	4	4	6149/13696 (44%)	36/136 (32%)	32/136 (23%)	71	0.273	4.53	4.50
	16	4	4	4508/13696 (32%)	36/136 (32%)	32/136 (23%)	72	0.271	8.49	13.45
	8	3	2	6168/13696 (45%)	56/136 (41%)	52/136 (38%)	74	0.28	4.28	13.10
	4	2	2	7474/13696 (54%)	68/136 (50%)	64/136 (47%)	73	0.281	4.192	26.56
Medium fading	64	4	4	6155/13696 (44%)	36/136 (32%)	32/136 (23%)	75	0.279	4.34	4.31
	16	2	2	8466/13696 (61%)	68/136 (30%)	64/136 (47%)	83	0.281	3.68	5.82
	8	4	1	8493/13696 (61%)	84 (61%)	80 (58%)	49	0.708	8.62	9.89
	4	1	1	11940/13696 (87%)	132/136 (97%)	128 (94%)	46	1.181	3.33	20.00

limitation of the present architecture in terms of maximum numbers. Again, we can find a slight difference in terms of hardware resources (number of slices) between the results after synthesis in Table 5 and the results before synthesis by our resource-estimator tool in Table 1. This was explained in Section 4 by the absence of database for FPGA components. We consider only the number of multipliers and BRAMs in our integer nonlinear programming model. Moreover, even with knowledge of the database, the resource estimation before synthesis is still difficult [21]. Nevertheless, for the main resources, the number of multipliers and BRAMs are exactly the same as in Table 1.

6. CONCLUSIONS

The HW architectures of a multiuser detector based on a cascade of adaptive filters (CF-MUD) for WCDMA systems were developed. The CF-MUD based on FIR using an LMS adaptation process presented a good choice for targeting FPGA devices. We have exploited the implementation advantages of the algorithm and the particular features of Xilinx devices. The regularity and recursiveness of the CF-MUD algorithm offer the opportunity to maximize the utilization ratio in the resource of the FPGA device. Using real-time implementation and taking into account all UMTS constraints, we demonstrated a utilization ratio in the resource near to 100% to maximize the parallelism of the CF-MUD algorithm. These dedicated architectures can be used later as optimized IP cores performing MUD functions. The current HW architectures are purely glue logic. Future work will consist of exploiting software processing in the multirate CF-MUD as a whole respecting the constraint specifications of the 3G wireless communications.

ACKNOWLEDGMENTS

The authors are grateful for the financial support of the Natural Sciences and Engineering Research Council of Canada (NSERC). We also wish to thank Axiocom Inc. for its technical and financial assistance.

REFERENCES

[1] P. Chaudhury, W. Mohr, and S. Onoe, "The 3GPP proposal for IMT-2000," *IEEE Communications Magazine*, vol. 37, no. 12, pp. 72–81, 1999.

[2] 3rd Generation Partnership Project (3GPP), "Spreading and modulation (FDD)," Tech. Rep. TS 25.213 v4.1.0 (2001-06), 3GPP, Valbonne, France, 2001.

[3] S. Verdú, *Multiuser Detection*, Cambridge University Press, New York, NY, USA, 1998.

[4] A. O. Dahmane and D. Massicotte, "DS-CDMA receivers in Rayleigh fading multipath channels: direct vs. indirect methods," in *Proceedings of IASTED International Conference on Communications, Internet and Information Technology (CIIT '02)*, St. Thomas, Virgin Islands, USA, November 2002.

[5] A. O. Dahmane and D. Massicotte, "Wideband CDMA receivers for 3G wireless communications: algorithm and implementation study," in *Proceedings of IASTED International Conference on Wireless and Optical Communications (WOC '02)*, Banff, Alberta, Canada, July 2002.

[6] S. Moshavi, "Multi-user detection for DS-CDMA communications," *IEEE Communications Magazine*, vol. 34, no. 10, pp. 124–136, 1996.

[7] S. Rajagopal, S. Bhashyam, J. R. Cavallaro, and B. Aazhang, "Real-time algorithms and architectures for multiuser channel estimation and detection in wireless base-station receivers," *IEEE Transaction on Wireless Communications*, vol. 1, no. 3, pp. 468–479, 2002.

[8] O. Leung, C.-Y. Tsui, and R. S. Cheng, "VLSI implementation of rake receiver for IS-95 CDMA testbed using FPGA," in *Proceedings of IEEE Asia and South Pacific on Design Automation Conference (ASP-DAC '00)*, pp. 3–4, Yokohama, Japan, January 2000.

[9] G. Xu, S. Rajagopal, J. R. Cavallaro, and B. Aazhang, "VLSI implementation of the multistage detector for next generation wideband CDMA receivers," *The Journal of VLSI Signal Processing*, vol. 30, no. 1-3, pp. 21–33, 2002.

[10] Y. Guo, G. Xu, D. McCain, and J. R. Cavallaro, "Rapid scheduling of efficient VLSI architectures for next-generation HSDPA wireless system using Precision C synthesizer," in *Proceedings of 14th IEEE International Workshop on Rapid Systems Prototyping (RSP '03)*, pp. 179–185, San Diego, Calif, USA, June 2003.

[11] W. Schlecker, A. Engelhart, W. G. Teich, and H.-J. Pfleiderer, "FPGA hardware implementation of an iterative multiuser detection scheme," in *Proceedings of 10th Aachen Symposium on Signal Theory (ASST '01)*, pp. 293–298, Aachen,Germany, September 2001.

[12] B. A. Jones and J. R. Cavallaro, "A rapid prototyping environment for wireless communication embedded systems," *EURASIP Journal on Applied Signal Processing*, vol. 2003, no. 6, pp. 603–614, 2003, Special issue on rapid prototyping of DSP systems.

[13] D. Massicotte and A. O. Dahmane, "Cascade filter receiver for DS-CDMA communication systems," International Application Published Under the Patent Cooperation Treaty (PCT), May 2004, WO2004/040789.

[14] Q.-T. Ho and D. Massicotte, "FPGA implementation of adaptive multiuser detector for DS-CDMA systems," in *Proceedings of 14th International Conference on Field Programmable Logic and Application (FPL '04)*, pp. 959–964, Leuven, Belgium, August–September 2004.

[15] Q.-T. Ho and D. Massicotte, "A low complexity adaptive multiuser detector and FPGA implementation for wireless DS-WCDMA communication systems," in *Proceedings of Global Signal Processing Expo and Conference (GSPx '04)*, Santa Clara, Calif, USA, September 2004.

[16] The International Telecommunication Union (ITU), Geneva, Switzerland, available at: http://www.itu.org.

[17] Xilinx, San Jose, Calif, USA, available at: http://www.xilinx.com.

[18] G. De Micheli, *Synthesis and Optimization of Digital Circuits*, McGraw-Hill, New York, NY, USA, 1994.

[19] S. G. Nash and A. Sofer, *Linear and Nonlinear Programming*, McGraw-Hill, New York, NY, USA, 1996.

[20] S. Rajagopal, S. Rixner, and J. R. Cavallaro, "A programmable baseband processor design for software defined radios," in *Proceedings of 45th IEEE Midwest Symposium on Circuits and Systems (MWSCAS '02)*, vol. 3, pp. 413–416, Tulsa, Okla, USA, August 2002.

[21] C. Shi, J. Hwang, S. McMillan, A. Root, and V. Singh, "A system level resource estimation tool for FPGAs," in *Proceedings of 14th International Conference on Field Programmable Logic and Application (FPL '04)*, pp. 424–433, Leuven, Belgium, August–September 2004.

Quoc-Thai Ho received a B.S. degree in electrical and electronics engineering from the Ho Chi Minh City University of Technology, an M.S. degree in design of digital and analog integrated systems from the Institut National Polytechnique de Grenoble, and an M.S. degree in microelectronics from the École Doctorale de Grenoble in September 2000, October 2001, and June 2002, respectively. He is currently pursuing his Ph.D. in electrical engineering at the Université du Québec à Trois-Rivières where he joined the Laboratory of Signal and System Integration. His Ph.D. work consists of VLSI architectures of multiuser detectors for DS-WCDMA wireless communication systems of third generation. His actual research interests include VLSI implementation, design methodologies, FPGA-based rapid prototyping with applications to CDMA communication systems.

Daniel Massicotte received the B.S.A. and M.S.A. degrees in electrical engineering and industrial electronics in 1987 and 1990, respectively, from the Université du Québec à Trois-Rivières (UQTR), QC, Canada. He obtained the Ph.D. degree in electrical engineering in 1995 at the École Polytechnique de Montréal, QC, Canada. In 1994, he joined the Department of Electrical and Computer Engineering, Université du Québec à Trois-Rivières, where he is currently a Professor. He is currently the Head of the Laboratory of Signal and Systems Integration and Chief Technology Officer of Axiocom Inc. He received the Douglas R. Colton Medal for Research Excellence awarded by the Canadian Microelectronics Corporation, the PMC-Sierra High Speed Networking and Communication Award, and the Second place at the Year 2000 Complex Multimedia/Telecom IP Design Contest from Europractice in 1997, 1999, and 2000, respectively. His research interests include VLSI implementation and digital signal processing for the communications and measurement problems such as nonlinear equalization, multiuser detection, channel estimation, and signal reconstruction. He is the author and the coauthor of more than 60 technical papers. He is also Member of the Ordre des Ingénieurs du Québec, Groupe de Recherche en Électronique Industrielle (GREI), and Microsystems Strategic Alliance of Québec (ReSMiQ).

Adel-Omar Dahmane received the B.S. degree in electrical engineering from the Université des Sciences et de la Technologie Houary Boumédienne (USTHB), Algiers, Algeria, in 1997, the M.S. and Ph.D. degrees with honours in electrical engineering from Université du Québec à Trois-Rivières, Trois-Rivières (UQTR), QC, Canada, in 2000 and 2004, respectively. He was two times the Laureate of the Governor General of Canada's Academic Medal (gold medal—graduate level) and a Fellow of the Natural Sciences and Engineering Research Council of Canada (NSERC). From 2002 to 2004, he worked for Axiocom Inc. as a Director of research and development. In 2004, he joined the Université du Québec à Trois-Rivières as Professor in electrical and computer engineering. His current research interests include wireless communications, spread-spectrum systems, multiuser detection, MIMO, and VLSI implementation issues. He is a Member of the Research Group in Industrial Electronics at the UQTR.