J Sign Process Syst
DOI 10.1007/s11265-008-0323-2

Energy-efficient Hardware Architecture and VLSI Implementation of a Polyphase Channelizer with Applications to Subband Adaptive Filtering

Yongtao Wang · Hamid Mahmoodi · Lih-Yih Chiou · Hunsoo Choo · Jongsun Park · Woopyo Jeong · Kaushik Roy

Received: 25 September 2008 / Accepted: 11 November 2008

Abstract Polyphase channelizer is an important component of subband adaptive filtering systems. This paper presents an energy-efficient hardware architecture and VLSI implementation of polyphase channelizer, integrating algorithmic, architectural and circuit level design techniques. At algorithm level, low complexity polyphase channelizer architecture is derived using multirate signal processing approach. To reduce the computational complexity in polyphase filters, computation sharing differential coefficient (CSDC) method is effectively used as an architectural level technique. The main idea of CSDC is to combine the strength of augmented differential coefficient method and subexpression sharing. Efficient circuit-level techniques: low power commutator implementation, dual-VDD scheme and novel level-converting flip-flop (LCFF), are also used to further reduce the power dissipation. The proposed polyphase channelizer consumes 352 mW power with throughput of 480 million samples per second (MSPS). A test chip has been fabricated in 0.18 μm CMOS technology and its functionality is verified. Chip measurement results show that the dual-VDD implementation achieves a total power saving of 2.7 X.

Keywords Multirate system · Polyphase channelizer · Very large scale integration (VLSI) · Low power design · Hardware architecture

1 Introduction

Subband adaptive filtering systems are widely used for adaptive signal processing applications that require filters

This work was supported in part by the DARPA MSP program and Semiconductor Research Corporation (1122.001).

Y. Wang · H. Choo
Texas Instruments Inc.,
Dallas, TX 75243, USA

Y. Wang
e-mail: yw@ti.com

H. Choo
e-mail: hunsoo@ti.com

H. Mahmoodi
San Francisco State University,
San Francisco, CA 94132, USA
e-mail: mahmoodi@sfsu.edu

L.-Y. Chiou
National Cheng Kung University,
Tainan, Taiwan
e-mail: lihyih@mail.ncku.edu.tw

J. Park (✉)
School of Electrical Engineering, Korea University,
Anam-dong, Seongbuk-Gu,
Seoul 136-701, Korea
e-mail: jongsun@korea.ac.kr

W. Jeong
Samsung Electronics Co.,
Hwasung, Korea
e-mail: woopyo.jeong@samsung.com

K. Roy
School of Electrical and Computer Engineering,
Purdue University,
West Lafayette, IN 47907, USA
e-mail: kaushik@ecn.purdue.edu

Published online: 13 December 2008

with very long impulse response and/or suffer from slow convergence speed [1–6]. In such applications, subband adaptive filtering is a viable alternative to conventional least-mean-square (LMS) algorithm since it reduces computational complexity and offers improved convergence rate. In the basic configuration of a subband adaptive filtering system as shown in Fig. 1, both input signal x[n] and desired response d[n] are decomposed into subbands using polyphase channelizers and all the adaptive filtering operations are performed independently in those subbands. After separate adaptive filtering, the subbands signals are recombined by a polyphase combiner to produce the final output.

The basic structure of a polyphase channelizer is illustrated in Fig. 2, which is a multirate digital signal processing system [7]. In the polyphase channelizer, the input signal x[n] is multiplied with the complex exponentials $W_M^{-(k)n} = e^{j2\pi kn/M}, \forall 0 \le k \le M-1$, that is equivalent to a uniform shift in frequency domain. The resulting signals are passed through a low-pass filter with impulse response h[n], which is generally called the prototype filter. The output of the prototype filter is decimated by a factor N to generate each subband signal. Adaptive filtering algorithm and/or other signal processing can then be applied separately to the subband signals. Compared to direct wideband filtering approach, subband filtering greatly reduces both update rate and length of the adaptive filters resulting in lower computational complexity. Moreover, processing data in separate subbands shows better convergence speed in the case of LMS algorithm, since the adaptation step size in each subband can be matched to the energy of the subband input signal [1, 2].

In the polyphase channelizer shown in Fig. 2, when critical sampling is employed (i.e., N=M), the presence of aliasing requires the use of adaptive cross-filters between adjacent subbands [1] or gap filterbanks [4]. However, systems with cross-filters generally converge more slowly and have higher computation cost, while the distortion produced by gap filter banks may not be acceptable.

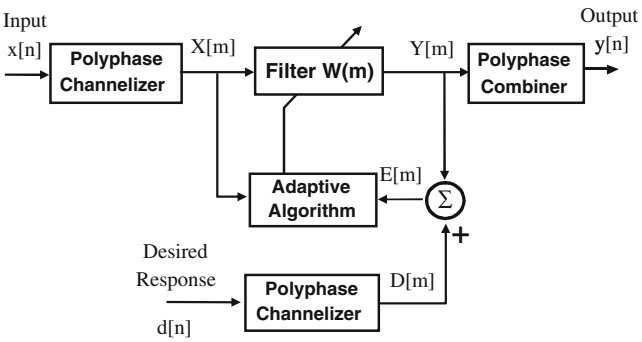


Figure 1 Basic configuration of a subband adaptive filtering system.

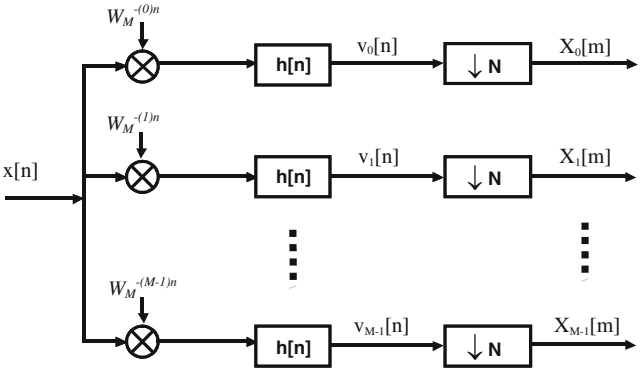


Figure 2 The basic structure of a polyphase channelizer.

Oversampled subband adaptive filtering systems with N< M offer simplified structure without employing cross-filters or gap filter banks and reduce the alias level in the subbands by allowing more spectral spacing between adjacent subbands. In order to reduce the computational complexity, the oversampling ratio M/N is usually chosen to be close to one. In our polyphase channelizer architecture, M=64 and N=48 are used.

Appropriate choice of the prototype filter h[n] is another important issue for the minimum mean-square error (MMSE) performance of subband adaptive filtering [6]. There are generally two criteria for choosing the prototype filters: sufficient stopband attenuation and perfect reconstruction. If the stopband attenuation of the prototype filter is high enough to sufficiently suppress the aliasing, the perfect reconstruction issue is simplified to the consideration of power-complementary [8]. To sufficiently reduce the alias level in the subbands, large prototype filter (e.g., with hundreds of taps) is necessary, which gives rise to large amount of power consumption, especially when the input data rate is high. A lot of research work has been conducted regarding the optimization over the choices of prototype filter and filter bank design [8–10]. However, little work has been done on efficient hardware architecture and VLSI implementation of the polyphase channelizer, which is the main focus of this work.

In order to achieve low power consumption while accommodating high input data rate, all design aspects from algorithm level to circuit need to be carefully analyzed and optimized. We present an energy-efficient hardware architecture and VLSI design techniques for implementing a polyphase channelizer with M=64 and N=48. The prototype filter has 768 taps and the input data rate is 480 million samples per second (MSPS). First, since the direct implementation of polyphase channelizer in Fig. 2 has large computational complexity, low complexity polyphase channelizer architecture is derived using current multirate signal processing techniques. In the low complexity architecture, all the polyphase filters can be expressed as transposed

direct form of FIR filters. Therefore, reduction of computational complexity in FIR filtering operation has a large impact on the power consumption of polyphase channelizer. Computation sharing differential coefficient (CSDC) method [11] is efficiently used to obtain low complexity parallel multiplierless implementation of FIR filters. The main idea of CSDC approach is to combine the strength of differential coefficient method [12] and subexpression sharing [13, 14], which leads to significant power savings in polyphase filters implementation.

In addition to the algorithmic/architectural level techniques, efficient circuit level techniques are also used for low power implementation of polyphase channelizer. The input data of polyphase channelizer are fed into the commutator using double data rate (DDR) format. However, using dual-edge triggered flip-flops incur larger area and power consumption than single-edge triggered flip-flop. We propose low power commutator design, which uses only positive-edge triggered flip-flops without degrading input data rates. To further reduce the power dissipation, since our proposed polyphase channelizer has two clock domains, efficient dual-VDD scheme and level-converting flip-flop (LCFF) are also presented as circuit level techniques.

The rest of the paper is organized as follows. In section 2, computationally efficient polyphase channelizer structure is derived using current multirate signal processing techniques. First part of section 3 presents the fixed-point modeling of polyphase channelizer based on tradeoffs between hardware complexity and system performance. Computation sharing differential coefficient (CSDC) approach for efficiently reducing power consumption in polyphase filters are also presented in section 3. In section 4, circuit-level techniques: energy efficient commutator, level-converting flip-flop (LCFF) and dual-VDD scheme, are explored to further reduce the power consumption of the proposed polyphase channelizer. VLSI implementation and the test chip results are presented in section 5, and section 6 concludes the paper.

2 Computationally Efficient Structure of Polyphase Channelizer

Although the basic structure shown in Fig. 2 is conceptually clear and useful, it is not computationally efficient, therefore not suitable for hardware implementation. In this section, we present a computationally efficient structure for polyphase channelizer using multirate signal processing approach.

As shown in Fig. 2, $X_k[m]$ is the output of k-th channel, $v_k[n]$ is the output of prototype filter h[n] for k-th channel, where $0 \le k \le M-1$. Vector $X[m] = [X_0[m], X_1[m], \ldots, X_{M-1}[m]]^T$ and vector $v[n] = [v_0[n], v_1[n], \ldots, v_{M-1}[n]]^T$. Since $X_k[m]$ is the decimation of $v_k[n]$ by a factor of N, $X_k[m]=v_k[mN]$, or $X[m]=v[mN]$. And $v_k[n] = \left(x[n]W_M^{-kn}\right) * h[n] = \sum_{l=-\infty}^{\infty} h[n-l] \times [l] W_M^{-kl}$, where * denotes the convolution operation. It follows that

$$\begin{aligned} X_k[m] &= \sum_{l=-\infty}^{\infty} h[mN-l] \times [l] W_M^{-kl} \\ &= \left(\sum_{l=-\infty}^{\infty} h[mN-l] \times [l] W_M^{k(mN-l)} \right) W_M^{-kmN} \\ &= \left(\sum_{l=-\infty}^{\infty} r_k[mN-l] \times [l] \right) W_M^{-kmN} \\ &= \left(r_k[n] * \times [n] \right) \downarrow_N W_M^{-kmN}, \end{aligned}$$

where $\downarrow N$ denotes decimation by a factor of N. And $r_k[n] = h[n]W_M^{kn}$, which in z-domain is $R_k(z) = H\left(zW_M^{-k}\right)$, where $H(z)$ denotes the z-transform of the prototype filter impulse response h[n], i.e., $H(z) = \sum_{n=-\infty}^{\infty} h[n] z^{-n}$, and $W_M^{-k} = e^{j2\pi k/M}$. The polyphase channelizer is thus transformed into the structure as shown in Fig. 3a.

The filters $r_k[n] = h[n]W_M^{kn}$ with $0 \le k \le$ M-1, and their following decimation operations form a new filter bank, which is highlighted by the dashed-line rectangle in Fig. 3a. The input of the filter bank is x[n] and the outputs of this new filter bank are $(r_k[n] * \times [n]) \downarrow_N$ with $0 \le k \le$ M-1.

Using polyphase decomposition and factorization, this new filter bank can be transformed into three serially connected processing blocks, i.e., a commutator with decimation factor of N, an M×N polyphase matrix and an

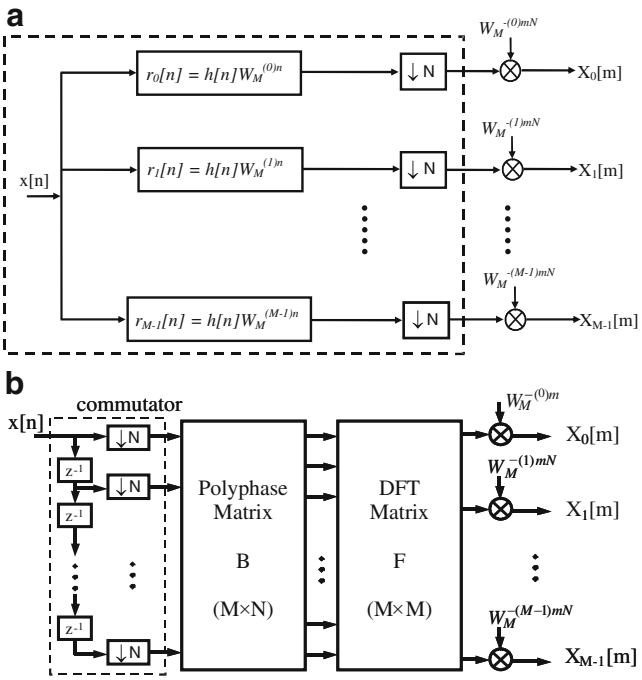


Figure 3 Computationally efficient polyphase channelizer.

M×M DFT matrix, which is shown in Fig. 3b [15]. In this low complexity architecture of polyphase channelizer, commutator is composed of a delay chain followed by decimators. The elements of M×N polyphase matrix B(z) are given by

$$[B(z)]_{ij} = \begin{cases} z^{-l} Q_{(j+lN)}(z^L), & if\,(i-j) \bmod g = 0 \\ 0, & if\,(i-j) \bmod g \neq 0 \end{cases}$$
$$(j+lN) \bmod M = i, g = \gcd(M,N), 0 \le i \le M-1, 0 \le j \le N-1,$$

where gcd(M,N) is the greatest common divisor of M and N. Here, K is the least common multiple of M and N, and J and L are the two integers satisfying K = JM = LN. In the M×N polyphase matrix, filter Q_l(z) is the l-th K-th order polyphase element of the prototype filter H(z), i.e., $H(z) = \sum_{l=0}^{K-1} z^{-l} Q_l(z^K)$ and $Q_l(z) = \sum_{n=-\infty}^{\infty} q_l(n) z^{-n}$ with $q_l[n]=h[Kn+l]$.

In this polyphase channelizer architecture, excluding the commutator, the whole system is operating at the rate N times lower than the input data rate, which significantly reduces the amount of required computations. Instead of using one large prototype filter h[n] for each subband channel, only one polyphase matrix B is needed for all M channels. DFT matrix can be implemented using the FFT structure [7]. For M=64, N=48, multiplication with the complex exponential W_M^{-kmN} is trivial since W_M^{-kmN} reduces to $(-j)^{km}$. Consequently, computational efficiency of the derived architecture is much higher than the basic architecture in Fig. 2, thereby achieving much higher energy efficiency.

Structure of the polyphase matrix B with M=64 and N=48 is shown in Fig. 4, where each non-zero element filter is represented by a dot. As mentioned above, filter $Q(z)$is the polyphase element of the prototype filter$H(z)$. There are totally K=192 nonzero polyphase element filters in B and the whole matrix B can be divided into 12 16×16 sub-matrices. The nonzero element filters of B are located on the main diagonal of these sub-matrices. For example, in the second sub-matrix on the first row, the non-zero element filters on its main diagonal are shown in Fig. 4 as $z^{-1}Q_{64-79}(z^4)$, i.e., the first non-zero element along the diagonal is $z^{-1}Q_{64}(z^4)$ filter, the second non-zero element along the diagonal is $z^{-1}Q_{65}(z^4)$ and so on. All other non-zero elements are similarly illustrated in Fig. 4.

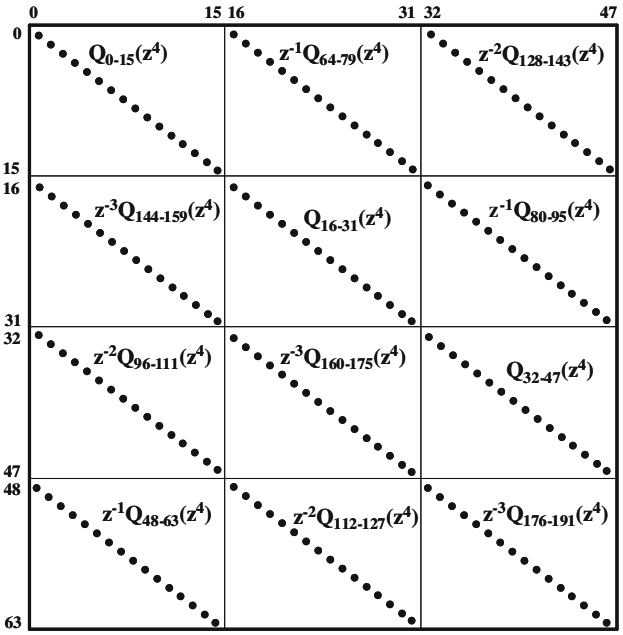


Figure 4 Structure of the polyphase matrix B with M=64 and N=48.

3 Fixed-Point Modeling and Low Complexity Architecture

3.1 Fixed-Point Modeling of Polyphase Channelizer

For the fixed point implementation of low complexity polyphase channelizer, floating-point data type needs to be converted into fixed-point and finite precision effect should be investigated. In the fixed-point modeling, there always exists a fundamental trade-off between hardware complexity and system performance. Considering the effect of quantization error on the polyphase channelizer system performance, fixed-point modeling of polyphase filters is presented in this subsection.

Integrated Side-Lobe Ratio (ISLR) is used as the metric for polyphase channelizer system performance. ISLR of the k-th channel (or subband) is defined as $ISLR_k = \frac{\text{Energy leaking into all other channels from the k - th channel}}{\text{Energy confined in the k - th channel}}$. In polyphase channelizer, the energy leakage in above equation causes alias distortion in each subband. In order to achieve a small minimum mean-square error (MMSE) of the subband adaptive filtering shown in Fig. 1, it is desirable to keep ISLR at a low level [6].

The input data of the polyphase channelizer come from a 12-bit ADC (analog-to-Digital Converter). We start by exploring the effect of the bit-length of the filter coefficients on ISLR. Relationship between bit-length of prototype filter coefficients and ISLR is shown in Fig. 5. When the bit-length is greater than 13, ISLR levels off and using more bits has little or no impact on ISLR. There exists a theoretical limit on achievable ISLR value for a given prototype filter. In the region where bit-length is smaller than 13, ISLR takes off and small reduction on bit-length will lead to large increase in ISLR. Hence the optimal point is the *knee* of the curve, which corresponds to a bit-length of 13 and an ISLR of −73.4 dB.

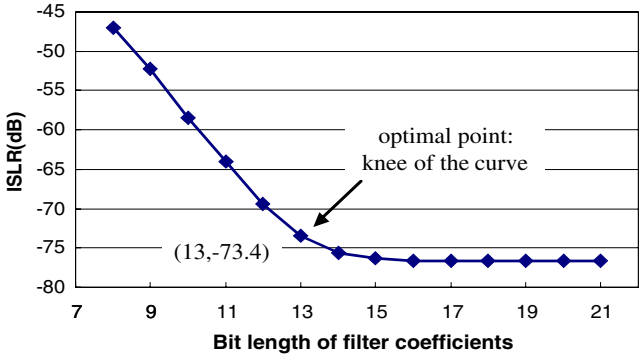


Figure 5 ISLR vs. Bit-length of filter coefficient.

3.2 Low Complexity Polyphase Filters Using CSDC

The polyphase matrix B(z) represents a *MIMO* (Multi-Input Multi-Output) system. The inputs to B(z) are denoted as vector $A(z)=[A_0(z), A_1(z),\ldots, A_{47}(z)]$ and outputs of U(z) are denoted as vector $U(z)=[U_0(z),U_1(z),\ldots,U_{63}(z)]$, which means $U(z)=B(z)A(z)$. Polyphase matrix B is shown in Fig. 6, where each filter Q_k represents the polyphase element of the prototype filter $H(z)$. More detailed inner structure of the polyphase matrix is shown in Fig. 7, providing views from an input port and an output port.

The implementation structure of the polyphase matrix B consists of two stages. The first stage is the *multiplication stage* in which all the inputs are multiplied by appropriate non-zeros elements in B(z). The second stage is the *summation stage* in which outputs of B(z) are calculated by summing up appropriate three outputs from the multiplication stage. For example, $A_0(z)$ is multiplied by $Q_0(z^4)$, $Q_{48}(z^4)z^{-1}$, $Q_{96}(z^4)z^{-2}$ and $Q_{144}(z^4)z^{-3}$ as shown in Fig. 7a, and $U_0(z)=Q_0(z^4)A_0(z)+z^{-1}Q_{64}(z^4)A_{16}(z)+z^{-2}Q_{128}(z^4)A_{32}(z)$, which is shown in Fig. 7b. Since the prototype filter has 768 taps, denoted as the set {h[0], h[1], …, h[767]}, each polyphase element of the prototype filter Q_k has four taps. From Figs. 6 and 7a, we note that every input of B(z), $A_k(z)$, is multiplied by four polyphase filters: Q_k,Q_{k+48},Q_{k+96},Q_{k+144}. These polyphase filters can be implemented using the transposed direct form structure as shown in Fig. 7a, where every input of B(z) is multiplied by a set of constants consisting of 16 filter coefficients. For example, as shown in Fig. 7a, $A_0(z)$ is multiplied by a set of constants consisting of coefficients h[0], h[192], h[384], h[576], h[48], h[240], h[432], h[624], h[96], h[288], h[480], h[672], h[144], h[336], h[528] and h[720]. Since we have 48 of 16 tap transposed direct from filters, computational complexity reduction on these multiplications has a large impact on the hardware implementation of the polyphase matrix B. For this purpose, we developed an efficient computational complexity reduction technique, called Computation Sharing Differential Coefficient (CSDC) method [11], which can be used to obtain low-complexity parallel multiplierless implementation of FIR filters and DSP tasks involving multiplications with a set of constants.

Let us consider the CSDC approach. Figure 8 shows the transposed direct form structure of M-tap FIR filter. In this figure, the set of FIR filter coefficients are represented as a vector $C = [c_0, c_1, \ldots C_{M-1}]$ and the outputs of the multiplication network are $P^{(n)} = \left[P_0^{(n)}, P_1^{(n)}, \ldots P_{M-1}^{(n)}\right]$. The input and output relation of the multiplication network can be expressed as $P_i^{(n)} = x(n)c_i$, where $0 \le i \le M-1$. To reduce the computational complexity of FIR filters, differential coefficients method [12] was proposed. In this approach, by considering the differential coefficient c_i-c_j, the computation of $P_i^{(n)} = x(n)c_i$ can be represented as $P_i^{(n)} = x(n)(c_i - c_j) + x(n)c_j$. When $x(n)(c_i$-$c_j)$ is very simple (e.g. $(c_i$-$c_j)$ is a power of two), other than computing $P_i^{(n)}$, we can simply reuse $P_j^{(n)}$ and sum up $P_j^{(n)}$ with $x(n)(c_i$-$c_j)$ to produce $P_i^{(n)}$. Compared to direct $P_i^{(n)}$ computation, since $x(n)(c_i$-$c_j)$ is much simpler, we can reduce the required amount of computations. In *augmented differential coefficients* approach, in addition to considering the difference between coefficients, $(c_i$-$c_j)$, sum of coefficients, (c_i+c_j), is also considered. In other words, $P_i^{(n)}$ can also be expressed as $P_i^{(n)} = x(n)(c_i + c_j) - x(n)c_j$, where we can achieve computation reduction if $x(n)(c_i+c_j)$ is much simpler. In the augmented differential coefficient approach, considering both the differences and sums of the filter coefficients greatly expands the design space, thus increasing the opportunities for more computational complexity reduction.

Subexpression sharing [13, 14] approaches are also used to reduce the computational complexity in FIR filter implementations. For example, consider FIR filter with three taps $c_0 = 0100101$, $c_1 = 10001001$, $c_2 = 00111001$. Without using computation sharing, seven additions are needed. However, we can easily notice that there is one common subexpression *1001* in these three coefficients. If we first compute $1001 \times x(n)$ and share the result of this computation among three coefficients, only 4 more additions are needed. By exploiting computation sharing, common computations are computed once and shared for

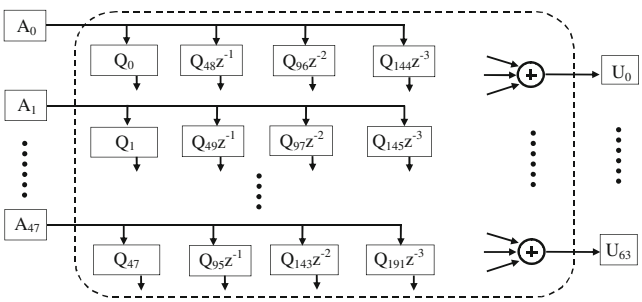


Figure 6 Implementation structure of the polyphase matrix B.

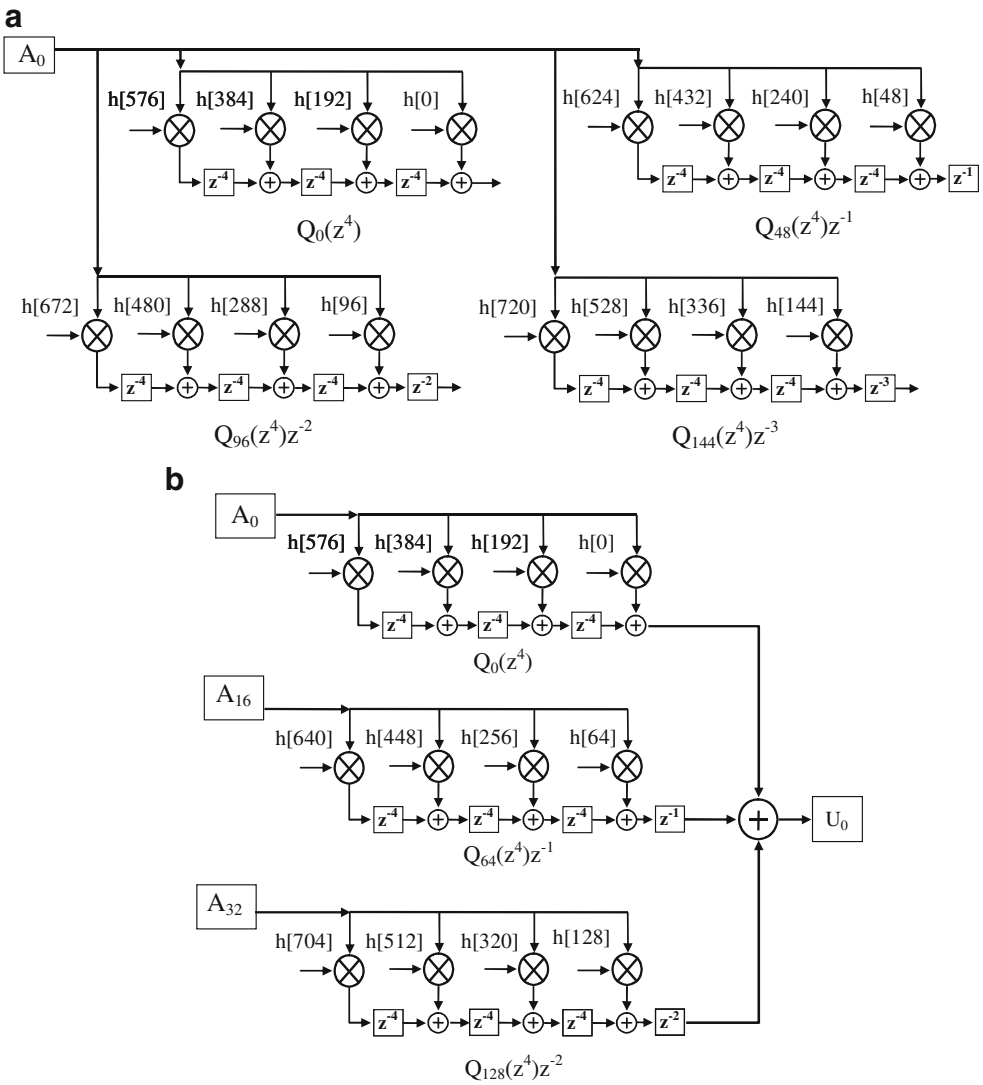


Figure 7 Illuminating the inner structure of polyphase matrix B.

all the filter coefficients, which significantly reduces total number of additions/subtractions.

The main idea of the CSDC method is to combine the strength of the augmented differential coefficient approach and subexpression sharing. As mentioned above, the augmented differential coefficient approach expands the design space. The expanded design space can be expressed as undirected and complete graph representation, where vertex set means all the filter coefficient and edges between two vertices means adder cost. The problem of minimizing the adder cost (the number of additions/subtractions) for a given filter is transformed into a problem of searching for *minimum spanning tree* with appropriate subexpression set. A heuristic search algorithm based on genetic algorithm is used to search for low-complexity solutions over the expanded design space in conjunction with exploring subexpression sharing. Comparison with several existing techniques based on the available data shows that our method yields comparable or better results for multiplierless FIR filter implementation. When applied to the polyphase filter in matrix B, CSDC achieved 57% complexity reduction in terms of the number of additions in comparison with the implementation in which all the coefficients are encoded in the canonical signed digit (CSD) format, which leads to significant area and power savings in our polyphase channelizer implementation.

Since M=64, for implementing the DFT matrix, we use the well-known radix-4 FFT structure, which has three stages. In the fixed-point modeling, the important nodes

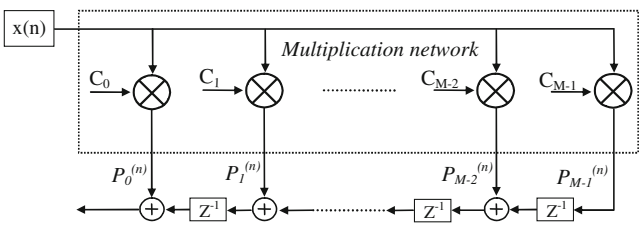


Figure 8 Transposed direct form of M-tap FIR filter.

under consideration include the output nodes of the polyphase matrix, output nodes of the first and second stages of the FFT and the final outputs. Outputs of the final stage of the FFT have the same bit-length as the final outputs since magnitude of the complex exponential, W_M^{-kmN}, is 1. Fixed-point modeling and extensive simulations in Matlab and Simulink [16] have been performed to determine the bit-length of DFT twiddle factors and the important nodes mentioned above.

Appropriate scaling is also employed to avoid overflow. Within the polyphase matrix, for the given prototype filter, the maximum possible gain is about 1.85, which is less than 2. Hence scaling by 0.5 is enough to avoid overflow. Scaling by 0.25 is applied to output of each stage of the FFT. Table 1 summarizes the bit-lengths used for final hardware implementation. The resulting ISLR is −66 dB.

4 Circuit Level Techniques

In this section, circuit-level design techniques, including an efficient commutator implementation, dual-VDD scheme, and a novel level-converting flip-flop (LCFF) are described. The input data of polyphase channelizer are fed into the commutator using double data rate (DDR) format. However, using dual-edge triggered flip-flops incur larger area and power consumption. We propose low power commutator design, which uses only positive-edge triggered flip-flops without degrading input data rates. To further reduce the power dissipation, since our proposed polyphase channelizer has two clock domains, efficient dual-VDD scheme and level-converting flip-flop (LCFF) are also presented.

5 An Efficient Commutator Circuit Implementation with Double Data Rate (DDR) Data Input

The input data (DATA) of polyphase channelizer are generated by an ADC (analog-to-Digital Converter) and are fed into the polyphase channelizer using a Double Data Rate (DDR) format with a DATA_VALID signal. The assertion of DATA_VALID signal indicates that the input data is valid. The timing diagram is illustrated in Fig. 9.

As shown in Fig. 3b, the commutator is composed of a delay chain, which consists of 47 serially connected delay elements, and 48 decimators. The outputs of the commutator are denoted as $x_1[m]$, $x_2[m]$, …, $x_{47}[m]$, $x_{48}[m]$, from top to bottom. Since data input from ADC uses Double Data Rate (DDR), one straightforward implementation is to use dual-edge triggered flip-flops. However, using dual-edge triggered flip-flops would incur larger area and power consumption than single-edge triggered flip-flops.

Table 1 Bit-lengths used for hardware implementation.

Filter coefficients	Outputs of polyphase matrix B	DFT twiddle factors	Outputs of 1st and 2nd FFT stage	Final outputs
13	16	16	16	16

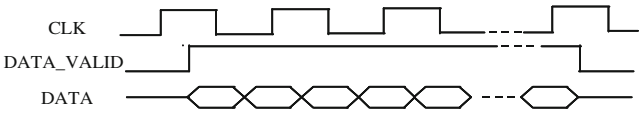
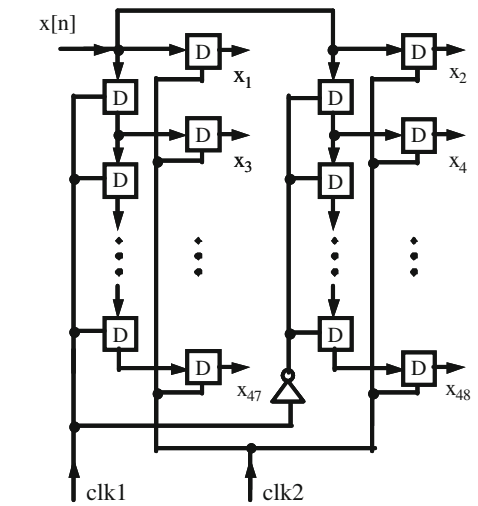


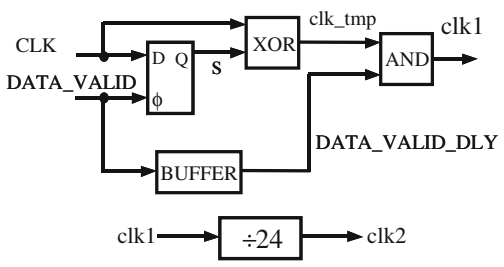
Figure 9 Timing diagram of DDR data input.

We developed an efficient commutator implementation, which uses only positive-edge triggered flip-flops, and a clock generation circuit as shown in Fig. 10a, b, respectively. As shown in Fig. 10a, the input data sequence is first broken into two sequences: an odd data sequence and an even data sequence. Then these two data sequences are sampled by the flip-flops driven by clock signal clk2. In order to make the proposed circuit work, generating appropriate clock signals clk1 and clk2 is critical.

When DATA_VALID goes from low to high, first sampling edge of clk1 shall be a rising edge. Otherwise,



a Commutator Circuit



b Clock generation circuit

Figure 10 Efficient commutator implementation.

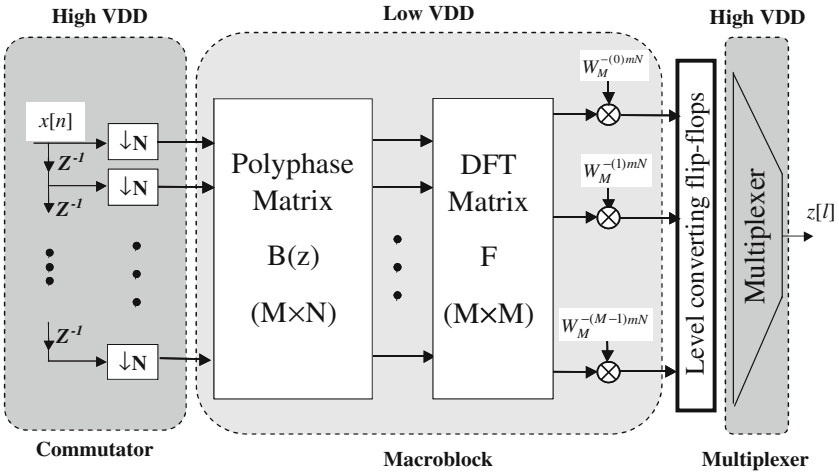


Figure 11 Dual-VDD, level converting and multiplexing.

input data will not be correctly sampled. Clock signal CLK cannot meet this requirement since the edge of CLK right after the rising edge of DATA_VALID can be either a rising edge or a falling edge (although in Fig. 9, it is drawn that the first edge of CLK right after the rising edge of DATA_VALID is a falling edge). The clock generation circuitry is shown in Fig. 10b. In the generation of clk1, first we use the rising edge of signal DATA_VALID to sample the CLK signal and generate signal **s** (Note that **s** is initially reset to low). If **s** is high, the first sampling edge of CLK must be a falling edge. Otherwise, the first sampling edge of CLK must be a rising edge. Propagating signal **s** and CLK through an XOR gate generates the clk_tmp signal. As a result, the first sampling edge of CLK generates a rising edge on clk_tmp, which can be used to sample the first input data sample. To remove possible glitches on clk_tmp, DATA_VALID is delayed by a buffer (BUFFER), generating the DATA_VALID_DLY signal. Finally, DATA_VALID_DLY and clk_tmp go through an AND gate such that the glitches in clk_tmp do not propagate through to clk1, thus producing a glitch-free clock signal clk1. It is necessary to carefully adjust the delay of the buffer (BUFFER) to make sure that clock signal clk1 is generated as desired. Dividing clk1 by 24 produces the clock signal clk2. Consequently, in conjunction with the clock generation circuit, the commutator is efficiently implemented without using dual-edge triggered flip-flops. Given the fact that the commutator is operating at a data rate 48 times as high as the rest of the polyphase channelizer, this efficient implementation of the commutator leads to considerable power saving.

5.1 Dual-VDD Scheme and Level-Converting Flip-Flop

As pointed out in section 2, the whole system except the commutator operates at a rate 1/48 of the input data rate. Therefore, we can apply the nominal supply voltage to the commutator while the supply voltage of the rest of the system (i.e. polyphase matrix, FFT and multiplications with the complex exponentials W_M^{-kmN}) can be scaled down. Due to the quadratic dependence between the switching power and supply voltage, such a dual-VDD scheme can lead to significant power savings. The associated overhead is the level conversion that is required to raise the output signal level to the high supply voltage at the interface from the low-VDD block to the high-VDD block. The application of the dual-VDD scheme is illustrated in Fig. 11.

To export the computation results of the subbands, multiplexing is usually employed as shown in Fig. 11. The multiplexer operates at the nominal supply voltage. The multiplexer (MUX) usually consists of two stages. At the first stage all the channel outputs are sampled and latched into flip-flops. At the second stage the latched

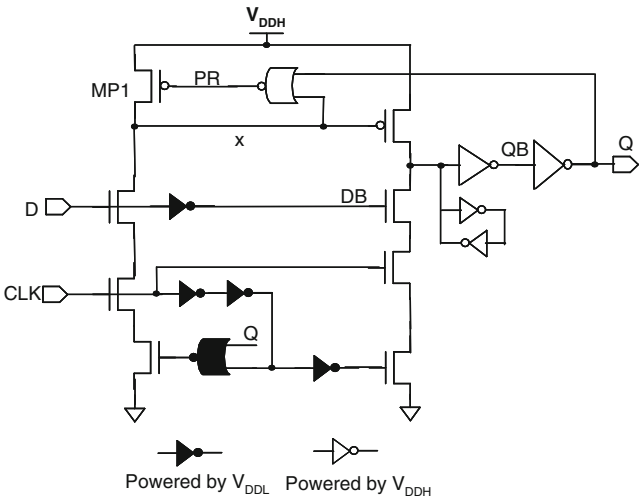


Figure 12 Level-converting flip-flop (LCFF).

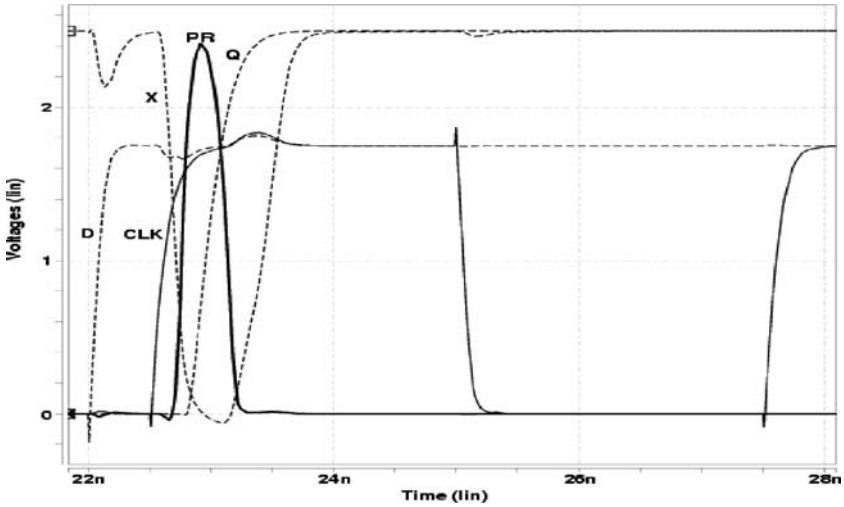


Figure 13 Simulation waveforms of the LCFF.

data are serially sent off the chip. Conventionally, the level converters and the flip-flops in the first stage of the multiplexer are designed and optimized separately. In this work, we developed a new Level-Converting Flip-Flop (LCFF) called Self-Precharging Filp-Flop (SPFF) [17], which merges a level converter and a flip-flip, leading to reduced area and power consumption and higher performance.

The schematic of the proposed level-converting flip-flop is shown in Fig. 12. It is composed of two stages. The first stage is a sampling circuit detecting voltage at the input during a pulse window implicitly generated on the rising edge of the clock. During the sampling window, the state of the input is captured to the dynamic node (X) and then stored to the second stage, which is a cross-coupled inverter latch. Conditional capturing capability has been incorporated by getting a feedback from the output through the NOR gate that drives the lowest NMOS transistor in the sampling paths. In this way, redundant transitions are removed from the dynamic node resulting in statistical power saving based on the data switching activity. The amount of power saving achieved by this internal clock gating is larger than the incurred power overhead for relatively low data switching activities. However, in high data switching activities the conditional capturing may not be of benefit since there is less chance to gate the clock and prevent redundant internal switching. The order of the transistor stack in the sampling path is based on the arrival time of the signals. The data input, which is the latest arriving signal, drives the transistor closest to the dynamic node. This ordering increases the performance of the flip-flop and allows more negative setup time. Negative setup

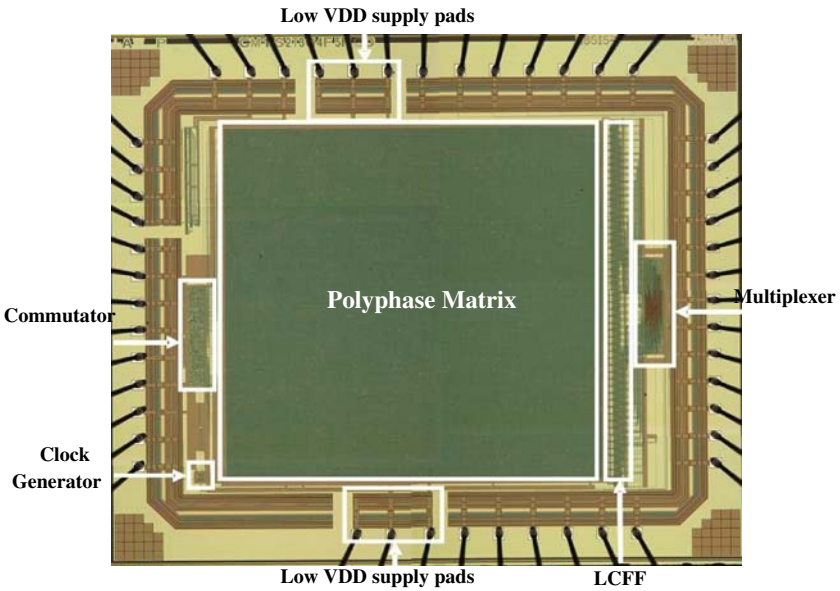


Figure 14 Die photo of the test chip.

time provides soft clock edge property [18], which is powerful in eliminating clock skew and jitter from timing budget in critical paths.

The precharging transistor (MP1) is derived by a self-resetting circuit providing the self-precharging capability to this flip-flop. If the dynamic node (X) is discharged, the output goes high and the precharge transistor (MP1) is turned on and recharges the dynamic node. During the rest of the cycle, the state of the dynamic node is kept charged by the NOR gate and the PMOS precharging transistor which act like an inverter and a keeper when the dynamic node is high. If the output goes high due to the discharge of the dynamic node, the feedback from the output to the sampling path turns off the sampling path so that the self-preharging operation does not cause any short circuit power consumption. In this flip-flop data and clock can have any voltage swing and the level conversion occurs on the dynamic node. Another benefit of the self-precharging technique is that it reduces the clock load and saves some clock power. Moreover, the switching activity of the self-precharging circuit is dependent on the data switching activity. Therefore, in moderate and low data switching activities the power overhead of the self-prechrging circuit is mitigated by the saving from the clock power.

Figure 13 shows the simulated waveforms of the flip-flops, which are obtained by HSPICE simulations of the flip-flops using typical models of a 0.25 μm CMOS technology at 25°C with V_{DDH}=2.5 V and V_{DDL}=1.75 V and the output load of 30 fF. As observed, the delay of the self-precharging (PR) is long enough so that it happens after latching the input data. Based on simulation results, the proposed flip-flop exhibits up to 60% delay reduction and 35% improvement in power delay product as compared to conventional level converting flip-flops proposed in [18].

6 VLSI Implementation and Test Chip Results

The proposed polyphas channelizer is implemented using well-automated design flow from algorithmic optimization and fixed-point modeling in Maltab/Simulink [16], VHDL coding and logic synthesis to physical design. LCFF is designed using full custom design method. The rest of the design, including the commutator, the clock generation circuit, polyphase matrix, FFT and multiplexer are coded in VHDL, synthesized using Synopsys tools [19] and their layouts are separately generated in Silicon Ensemble [20] using Artisan standard cell library [21]. Since our design incorporates dual VDD and combines custom and semi-custom blocks, the final layout is generated by assembling the layouts of all the constituent blocks in IC Craftsman [20]. We used TSMC 6-metal-alyer 0.18 μm CMOS technology. We performed full-chip simulation to estimate the power dissipation of the system. Based on the simulation results, when the whole system is operating at the nominal supply voltage of 1.8 V (i.e. without employing dual-VDD scheme), the total power consumption is about 844 mW with a throughput of 480 MSPS. However, using the proposed dual-VDD scheme, the lower VDD can be as low as 0.9 V and this leads to a power consumption of 352 mW, which corresponds to a power saving of 2.4X. The total layout area of the design is about 64 mm^2.

In order to reduce the fabrication cost while validating the proposed hardware architecture and VLSI design techniques, the design was simplified by reducing the wordlength of input data to 4 bits. Bit-lengths of filter coefficients and DFT twiddle factors remain unchanged while bit-lengths of other nodes are reduced by 8. The test chip includes the commutator and its clock generation circuitry, the polyphase matrix, LCFF block, and multiplexer.

The test chip was fabricated using the TSMC 6-metal-layer 0.18 μm CMOS technology. Die photo of the test chip

Table 2 Features of polyphase channelizer chip.

Process		TSMC 0.18 μm
Voltage	nominal VDD	1.8 V
	VDDL	0.9 V
Clock frequency		240 Mhz
Power (at 240 Mhz)	normal operation	57 mW
	using dual-VDD	21.1 mW
die area		10 mm^2

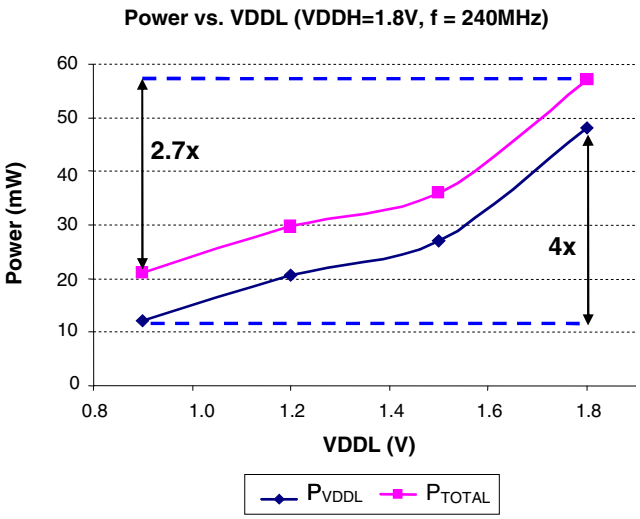


Figure 15 Power consumption of the test chip at different VDDL.

is shown in Fig. 14. The total area of the test chip is about 10 mm^2. The nominal supply voltage for the core is 1.8 V while 3.3 V supply is used for the I/O cells. In order to explore the full potential of the dual-VDD scheme, a separate set of power and ground pads were used to provide the low VDD supply to the low-VDD portion of the test chip as shown in Fig. 14. The test chip is packaged in a 52-pin ceramic Leadless Chip Carrier (LCC) package.

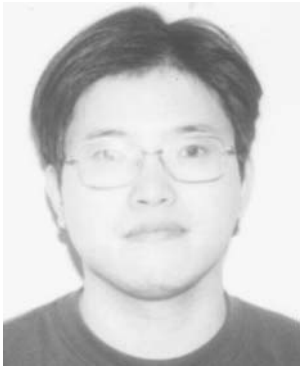
Table 2 shows the features of our polyphase channelizer test chip. Functionality verification and power measurements of the test chip were done at different low VDD (VDDL) values. A Tektronix logic analyzer was used for input pattern generation and output monitoring. Power consumption was measured by applying sequences of random input data. The test chip was functional at VDDL from 1.8 V to 0.9 V at input data rate of 480 MSPS, i.e., frequency of the input clock signal is 240 MHz. The power consumption of the test chip at different VDDL is also shown in Fig. 15. The top curve shows the total power consumption of the chip, which is denoted by P_{TOTAL}. Without employing dual-VDD (i.e, the whole chip operates at 1.8 V), the total power consumption is 57 mW. By reducing VDDL to 0.9 V, the total chip power consumption is 21.1 mW, reduced by a factor of 2.7. The bottom curve shows the power consumption of the low-VDD portion of the test chip, denoted by P_{VDDL}, which closely follows a quadratic dependence on VDDL.

7 Conclusions

We presented an energy-efficient hardware architecture and VLSI implementation of a polyphase channelizer, which is an important component of subband adaptive filtering system. Optimizations at the algorithmic, architectural and circuit level are integrated to achieve low power consumption while accommodating a high system throughput. As algorithmic and architectural techniques, multirate signal processing and computation sharing differential coefficient (CSDC) method are effectively used. Efficient circuit-level techniques such as low power commutator implementation, dual-VDD scheme and novel level-converting flip-flop (LCFF), are also used to further reduce the power dissipation. Simulation results of the full bit-length implementation of the proposed polyphase channelizer show a power consumption of 352 mW with system throughput of 480 MSPS. A reduced bit-length version of the design was fabricated in a test chip using TSMC 0.18 um process for functional verification of the proposed hardware architecture and VLSI design techniques. Chip measurement results show a power saving of 2.7X using the proposed dual-VDD implementation.

References

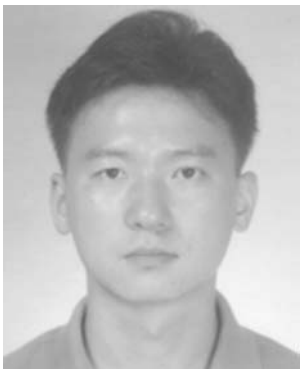
1. Gilloire, A., & Vetterli, M. (1992). Adaptive filtering in subbands with critical sampling: Analysis, experiments and applications to acoustic echo cancellation. *IEEE Trans Signal Process*, *40*, 1862–1875. doi:10.1109/78.149989.
2. Shynk, J. J. (1992). Frequency-domain and multirate adaptive filtering. *IEEE Signal Process Mag*, *9*, 14–37. doi:10.1109/79.109205.
3. Weiss, S., et al. (1998). Adaptive equalization in oversampled subbands. *Electron Lett*, *34*(15), 1452–1453. doi:10.1049/el:19981085.
4. Tanrikulu, O., et al. (1997). Residual echo signal in critically sampled subband acoustic echo cancellers based on IIR and FIR filter banks. *IEEE Trans Signal Process*, *45*(4), 901–912. doi:10.1109/78.564178.
5. Song, W. S., et al. (2000). High-performance low-power polyphase channelizer chip-set. *Asilomar Conference on Signals, Systems and Computers*, *2*, 1691–1694.
6. Weiss, S., et al. (2001). Steady-state performance limitations of subband adaptive filters. *IEEE Trans Signal Process*, *49*(9), 1982–1991. doi:10.1109/78.942627.
7. Proakis, J. G., & Manolakis, D. G. (1996). *Digital signal processing: principles, algorithms and applications*, Third edition, Prentice Hall Inc.
8. Vaidyanathan, P. P. (1993). *Multirate Systems and Filter Banks*. Prentice Hall Inc.
9. Harteneck., M., Weiss, S., & Stewart, R. W. (1999). Design of the near perfect reconstruction oversampled filterbanks for subband adaptive filters. *IEEE Trans. On Circuits and Systems–II: Analog and Digital Signal Processing*, *46*(8). August.
10. Eneman, K., & Moonen, M. (1997). Filter bank constraints for subband and frequency-domain adaptive filters. *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics*, 19–22, Oct.
11. Wang, Y., & Roy, K. (2005). "CSDC: a new complexity reduction technique for multiplierless implementation of digital FIR filters". *IEEE Trans Circuits and Systems I: Fundamental Theory and Applications, 52*(9). September.
12. Sankaraya, N., Roy, K., & Bhattacharya, D. (1997). Algorithms for low power and high speed FIR filter realization using differential coefficients. *IEEE Trans Circuits Syst. II Analog Digit Signal Process*, *44*(6), 488–497. doi:10.1109/82.592582.
13. Hartley, R. I. (1996). Subexpression sharing in filtering using canonic signed digit multipliers. *IEEE Trans Circuits Syst II Analog Digit Signal Process*, *43*(10), 677–688. doi:10.1109/82.539000.
14. Pasko, R., et al. (1999). A new algorithm for elimination of common subexpressions. *IEEE Trans Computer-Aided Design of Integrated Circuits and Systems*, *18*(1), 58–68 Jan.
15. Cvetkovic, Z., & Vetterli, M. (1998). Tight Weyle-Heisenberg frames in $\ell^2(Z)$. *IEEE Trans Signal Process*, *46*(5). May.
16. The Mathworks, Inc.: Matlab and simulink. [Online]. Available: http://www.mathworks.com
17. Mahmoodi-Meimand, H., Roy, K. (2002). Self precharging flip-flop (SPFF): a new level-converting flip-flop. *European Solid-State Circuits Conference*, 407–410. Sep.
18. Partovi, H. (2001). Clocked storage elements. In A. Chandrakasan, W. J. Bowhill, & F. Fox (Eds.),*Piscataway design of high-performance microprocessor circuits* (pp. 207–234). NJ, USA: IEEE ch. 11.
19. Synopsys, Inc.: [Online]. Available: http://www.synopsys.com.
20. Cadence design systems, Inc. [Online]. Available: http://www.cadence.com.
21. Artisan components, Inc. [Online]. Available: http://www.artisan.com.



Yongtao Wang received his Ph.D. degree from the School of Electrical and Computer Engineering at Purdue University in 2005. His research interests include low-power/high-performance VLSI architectures for digital signal processing (DSP) and wireless communication applications, adaptive/multi-rate signal processing, sigma-delta modulator design, digital baseband processing for OFDM and MIMO systems, low-power/high-speed digital integrated circuit design, system/architecture for RF CMOS System-on-Chip (SoC) design, mixed-signal design with an emphasis on utilizing DSP means to combat RF/analog impairments. He interned at Texas Instruments during the summer of 2004 where he developed a novel method for designing sigma-delta modulators with arbitrary transfer functions. He joined the Wireless Terminals Business Unit of Texas Instruments as a systems engineer in August 2005 and has since worked on CMOS System-on-Chip products for various wireless standards such as GSM/GPRS/EDGE, WCDMA, WLAN and so on. He has published nine conference/journal papers and has four patents pending. He is a member of IEEE and Semiconductor Research Corporation (SRC). He has served as a reviewer for several international conferences and IEEE journals.

Hamid Mahmoodi received his B.S. degree in Electrical Engineering from Iran University of Science and Technology, Tehran, Iran, in 1998 and his M.S. degree in Electrical and Computer Engineering from the University of Tehran, Iran, in 2000. He received his Ph.D. degree in Electrical and Computer Engineering from Purdue University, West Lafayette, IN, in 2005. He is currently an assistant professor of Electrical and Computer Engineering at the School of Engineering at San Francisco State University. His research interests include low-power, robust, and high-performance circuit design in nano-scale technologies. He has many publications in journals and conferences and several patents. He was a co-recipient of the 2006 IEEE Circuits and Systems Society VLSI Transactions Best Paper Award and the Best Paper Award of the 2004 International Conference on Computer Design. He is a technical program committee member of IEEE Custom Integrated Circuits Conference and International Symposium on Quality Electronics Design.

Lih-Yih Chiou received his B.S.E.E. degree from the National Cheng-Kung University, Tainan, Taiwan, his M.S. degree from University of Louisiana, Lafayette, LA, and Ph.D. degree from Purdue University, West Lafayette, IN, in 1988, 1993, and 2003, respectively. In 2003, he joined the Electrical Engineering Faculty at National Cheng Kung University, Tainan, Taiwan. His research interests include low-power VLSI design and CAD for VLSI, electronic system-level design for SoC and reconfigurable computing.



Hunsoo Choo was born and raised in Seoul, Korea. He received Bachelors Degree in Electrical Engineering from Yonsei University at Seoul, Korea. Later, he traveled to US in pursuit of higher education. He received his M.S. and Ph.D. degrees, both in Electrical and Computer Engineering department of Purdue University in 2000 and 2005, respectively. Since 2005, he has been working at Texas Instrument, Dallas, TX as a RF system designer. His research interest includes CMOS mixed signal VLSI design, computer-aided design for

mixed signal integrated circuits and low-power and low-complexity digital circuit design.

Jongsun Park received his B.S. degree in Electronics Engineering from Korea University, Seoul, Korea, in 1998 and his M.S. and Ph.D. degrees in Electrical and Computer Engineering from Purdue University, West Lafayette, IN, in 2000 and 2005, respectively. From 2005 to 2008, he was with the Signal Processing Technology Group, Marvell Semiconductor Inc., Santa Clara, CA. He was also with the Digital Radio Processor System Design Group, Texas Instruments, Dallas, TX in summer of 2002. He joined the Electrical Engineering faculty of the Korea University, Seoul, Korea, in 2008. His research interests focus on variation-tolerant, low-power and high-performance VLSI architectures and circuit designs for digital signal processing and digital communications.

Woopyo Jeong received his B.S. and M.S. degrees in Electrical Engineering from Yonsei University, Seoul, Korea, in 1991 and 1993, respectively. In 1993, he joined Samsung Electronics Co., Ltd. in Korea, where he had been engaged in research and development for DRAM. He rejoined Samsung Electronics Co., after he had received his Ph.D. degree in Electrical Engineering from Purdue University, in 2004. He is currently working at mobile DRAM Design team of Samsung Electronics Co., LTD. in Korea. His research interests include high performance and low power circuit design.

Kaushik Roy received his B.Tech. degree in Electronics and Electrical Communications Engineering from the Indian Institute of Technology, Kharagpur, India, and Ph.D. degree from the Electrical and Computer Engineering Department of the University of Illinois at Urbana-Champaign in 1990. He was with the Semiconductor Process and Design Center of Texas Instruments, Dallas, where he worked on FPGA architecture development and low-power circuit design. He joined the Electrical and Computer Engineering faculty at Purdue University, West Lafayette, IN, in 1993, where he is currently a Professor and holds the Roscoe H. George Chair of Electrical & Computer Engineering. His research interests include VLSI design/CAD for nano-scale Silicon and non-Silicon technologies, low-power electronics for portable computing and wireless communications, VLSI testing and verification, and reconfigurable computing. Dr. Roy has published more than 450 papers in refereed journals and conferences, holds 8 patents, and is co-author of two books on Low Power CMOS VLSI Design (John Wiley & McGraw Hill). Dr. Roy received the National Science Foundation Career Development Award in 1995, IBM faculty partnership award, ATT/Lucent Foundation award, 2005 SRC Technical Excellence Award, SRC Inventors Award, Purdue College of Engineering Research Excellence Award, and best paper awards at 1997 International Test Conference, IEEE 2000 International Symposium on Quality of IC Design, 2003 IEEE Latin American Test Workshop, 2003 IEEE Nano, 2004 IEEE International Conference on Computer Design, 2006 IEEE/ACM International Symposium on Low Power Electronics & Design, and 2005 IEEE Circuits and system society Outstanding Young Author Award (Chris Kim), 2006 IEEE Transactions on VLSI Systems best paper award. Dr. Roy is Purdue University Faculty Scholar. Dr. Roy was a Research Visionary Board Member of Motorola Labs (2002). He has been in the editorial board of IEEE Design and Test, IEEE Transactions on Circuits and Systems, and IEEE Transactions on VLSI Systems. He was Guest Editor for Special Issue on Low-Power VLSI in the IEEE Design and Test (1994) and IEEE Transactions on VLSI Systems (June 2000), IEE Proceedings – Computers and Digital Techniques (July 2002). Dr. Roy is a fellow of IEEE.