Dynamic Algorithm Transformations (DAT)—A Systematic Approach to Low-Power Reconfigurable Signal Processing

Manish Goel and Naresh R. Shanbhag, *Member, IEEE*

***Abstract*— In this paper, dynamic algorithm transformations (DAT's) for designing low-power reconfigurable signal-processing systems are presented. These transformations minimize energy dissipation while maintaining a specified level of mean squared error or signal-to-noise ratio. This is achieved by modeling the nonstationarities in the input as temporal/spatial transitions between states in the input state–space. The reconfigurable hardware fabric is characterized by its configuration state–space. The configurable parameters are taken to be the filter taps, coefficient and data precisions, and supply voltage $V_{\rm dd}$. An energy-optimal reconfiguration strategy is derived as a mapping from the input to the configuration state–space. In this strategy, taps are powered down starting with the tap with the smallest value of $[w_k^2/\mathcal{E}_m(w_k)]$ (where w_k and $\mathcal{E}_m(w_k)$ are, respectively, the coefficient and energy dissipation of the kth tap). Optimal values for precisions and supply voltage $V_{\rm dd}$ are subsequently computed from the roundoff error and critical path delay requirements, respectively. The DAT-based adaptive filter is employed as a near-end crosstalk (NEXT) canceller in a 155.52-Mb/s asynchronous transfer mode–local area network transceiver over category-3 wiring. Simulation results indicate that the energy savings range from -2% to 87% as the cable length varies from 110 to 40 m, respectively, with an average savings of 69%. An average savings of 62% is achieved for the case where the supply voltage $V_{\rm dd}$ is kept fixed.**

***Index Terms*— Algorithm transformations, low-power, reconfigurable computing, signal processing.**

I. Introduction

THE recent growth of portable wireless networked communication systems has made it essential that maximum functionality be provided for prolonged periods under severe constraints on battery weight and life. This fact has made low-power digital signal processing (DSP) an important research area. Energy minimization techniques have been proposed at all levels of the design hierarchy beginning with algorithms and architectures and ending with circuits and technological innovations. Existing techniques include those at the algorithmic level (such as strength reduction [1]–[3] and variable-length vector quantizer (VQ) [4]), architectural level (such as pipelining [5], [6] and parallel processing [6]), logic (logic minimization [7], [8] and precomputation [9]), circuit (reduced voltage swing [10] and adiabatic logic [11]) and technological level [12]. Algorithm transformation techniques [13] such as look-ahead [6], relaxed look-ahead [14], algebraic transformations [15], and retiming [16] have been employed in high-speed and, more recently, low-power DSP system design. We refer to these algorithm transformations as static algorithm transformations (SAT's) because these are applied during the algorithm design phase, assuming a worst-case scenario, and their implementation is time-invariant.

In recent years, reconfigurable signal processing has emerged as an alternative approach to low-power DSP. In [17], reconfigurability is employed to map a wide class of signal-processing algorithms (**SPA**'s) to an appropriate architectural template. Related work includes approximate signal processing [18], [19] where just the right amount of computational resources, needed at a specific instant/period to meet the algorithm performance requirements, is allocated. Compiler-based run-time software optimization techniques are explored in [20], [21]. Field-programmable gate-array (FPGA)-based devices and their reconfiguration strategies are discussed in [22]–[24]. Hybrid architectures based on FPGA's and general-purpose DSP's is the topic of research in [25] and [26].

Our approach to the design of reconfigurable DSP is to add just the right degree of flexibility (as demanded by the application) to ASIC's resulting in application-specific reconfigurable integrated circuits (ASRIC's). The ASRIC approach is suitable for mobile multimedia systems of the future as it maintains the energy and throughput efficiency of ASIC's. In this paper, we present dynamic algorithm transformations (DAT's) as a systematic approach to low-power ASRIC's. The reconfiguration strategies are derived via DAT's that optimize energy dissipation while maintaining a specified level of algorithmic performance measure such as mean squared error (MSE) or signal-to-noise ratio (SNR). The DAT techniques are based upon the principle that the input is usually nonstationary and, hence, it is better (from an energy perspective) to adapt the algorithm and architecture to the input. In contrast, present-day systems (referred to as "worst-case designs") seek out and design for the worst-case scenario. For example, a broadband modem is typically designed for the longest cable length, the maximum cable temperature, and the worst-case near-end crosstalk (NEXT) interferer. The worst-case design requires high complexity and, hence, high energy consumption that remains the same even if the cable length, in practice, is small. A DAT-based modem will exploit this spatial variability between one location to another to reconfigure itself to save

Manuscript received July 1, 1998; revised November 15, 1998. This work was supported by the National Science Foundation under CAREER Award MIP 96-23737, and by the Defense Advanced Research Projects Agency under Contract DABT63-97-C-0025.

The authors are with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA (e-mail: mgoel@uivlsi.csl.uiuc.edu; shanbhag@uivlsi.csl.uiuc.edu).

Publisher Item Identifier S 1063-8210(99)04569-2.

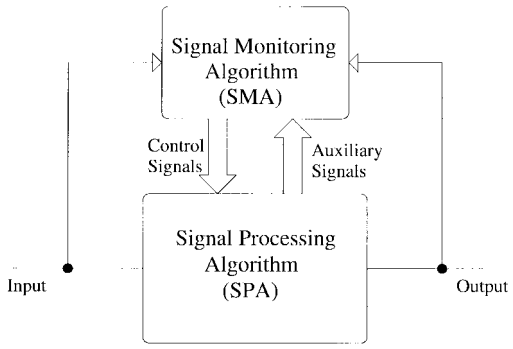


Fig. 1. Reconfigurable DSP system.

energy. Similarly, wireless channels exhibit extensive temporal variabilities due to fading that can be exploited by a DAT-based receiver.

The main contribution of this paper is to formalize the design of reconfigurable DSP systems. This is done by modeling: 1) the temporal/spatial variabilities in the input as state transitions in an input state–space and 2) reconfiguration of the hardware fabric as transitions in a configuration state–space. Given an input state, an energy-optimal configuration state is derived systematically as a solution to an optimization problem, which has energy as the objective function and a constraint on the SNR. The proposed design approach is independent of the hardware platform. From an implementation perspective, a reconfigurable DSP system has the **SPA** implemented in a reconfigurable hardware (see Fig. 1) (such as FPGA, certain DSP's, or ASIC's), while the input state and state transitions are monitored by a signal monitoring algorithm (**SMA**) block (or a controller).

In the past, the DAT-based approach has been successfully applied to low-energy equalizers for 51.84-Mb/s very-high-speed digital subscriber loop (VDSL) [27] and low-energy finite-impulse response (FIR) filters. In this paper, we apply DAT to a NEXT canceller for 155.52-Mb/s asynchronous transfer mode (ATM)–local area network (LAN) [28]. Simulation results indicate that the energy savings range from -2% to 87% as the cable length varies from 110 to 40 m, respectively, with an average energy savings of 69% as compared to the worst-case design corresponding to a cable length of 110 m.

The remainder of this paper is organized as follows. Section II provides preliminaries regarding adaptive filters and the variable supply voltage scheme. In Section III, we describe reconfigurable datapath and their energy dissipation models. The main result is described in Section IV, where we present DAT's for adaptive filters. Finally, in Section V, we employ the DAT-based adaptive filter as a NEXT canceller in 155.52-Mb/s ATM-LAN over unshielded twisted-pair (UTP) category-3 copper wiring and present simulation results.

II. Preliminaries

In this section, we describe preliminaries regarding adaptive filters [29] and the variable supply voltage scheme [19] that would be necessary for the development of DAT.

A. Adaptive Filtering

Adaptive filters are based upon a stochastic model of the input signal $x(n)$, where n is the time instant. The output of a fixed-coefficient N-tap filter processing the input $x(n)$ is given by

$$y(n) = \sum_{k=1}^{N} w_k x(n-k+1) \tag{1}$$

where $w_1, w_2, \cdots, w_N$ are the filter coefficients. The filter coefficients w_k can be chosen to minimize the error $e(n)$ given by

$$e(n) = d(n) - y(n) \tag{2}$$

where $d(n)$ is the desired signal. Typically, it is the MSE value that is minimized, where the MSE $\mathcal{J}(n)$ is defined as

$$\mathcal{J}(n) = E\Big[|e(n)|^2\Big] \tag{3}$$

where $E[\cdot]$ is the expectation operator. If the error $e(n)$ is assumed to be an ergodic process, then the sample average of $|e(n)|^2$ can be approximated by its time average, which is given by

$$\mathcal{J}(n) \approx \frac{1}{L} \sum_{j=0}^{L-1} |e(n-j)|^2 \tag{4}$$

where L is the window over which the squared error is averaged. It can be shown [29] that if the input signal is a wide-sense stationary (WSS) white process (samples of $x(n)$ are uncorrelated), then the minimum MSE $\mathcal{J}_{\min}$ is given by

$$\mathcal{J}_{\min} = \sigma_d^2 - \sum_{k=1}^{N} |w_{k,\text{opt}}|^2 \sigma_x^2 \tag{5}$$

where $\sigma_d^2 = E[d^2(n)]$ is the desired signal power, $\sigma_x^2 = E[x^2(n)]$ is the input signal power, and $w_{k,\text{opt}}$ are the optimum coefficients. For stationary inputs, the optimum coefficients can be computed [29] by solving a system of linear equations. However, if the input signal is nonstationary, then the coefficients w_k need to be updated via an adaptive algorithm such as the least mean square (LMS) algorithm [30], defined as follows:

$$y(n) = \sum_{k=1}^{N} w_k(n-1) x(n-k+1) \tag{6}$$

$$w_k(n) = w_k(n-1) + \mu e^*(n) x(n-k+1) \tag{7}$$

where $e^*(n)$ is the complex conjugate of $e(n)$, defined in (2), $d(n)$ is the desired signal, $y(n)$ is the filter output, and μ is the step size. If the step-size μ is sufficiently small, then the coefficients $w_k(n)$ will approach $w_{k,\text{opt}}$ as n approaches infinity. In that case, $\mathcal{J}(n)$ in (4) will approach the optimum value $\mathcal{J}_{\min}$ in (5). The LMS algorithm is commonly employed in numerous signal processing and communications applications due to its inherent simplicity.

A direct implementation of the LMS algorithm is shown in Fig. 2, where each tap consists of two multipliers and two adders. The filter (**F**) block implements (6) and weight-update (**WUD**) block implements (7). The critical path delay T_{cp} for the architecture in Fig. 2 is given by

$$T_{\text{cp}} = 2T_m + (N+1)T_{\text{sum}} + B_{\text{ADD}} T_{\text{carry}} \tag{8}$$

where T_m is the propagation delay for the multiplier, T_{sum} and T_{carry} are the propagation delays for the sum and carry outputs, respectively of a 1-b full adder, and B_{ADD} is the precision of

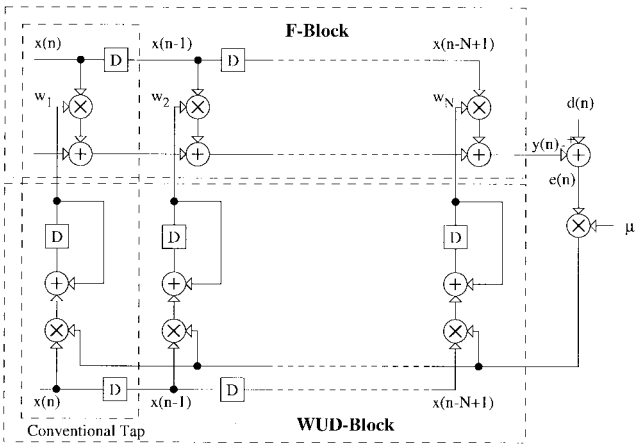


Fig. 2. LMS adaptive filter architecture.

the adder in the **F**-block. Note that we assume a ripple–carry adder architecture and that μ is a power-of-two. Typically, we choose $T_{\rm cp} \leq T_s$ so that the sample rate requirements are met. In many applications, the **WUD** block is switched off (to conserve energy) after the optimum coefficients have been obtained, i.e., after convergence. In that case, the critical path delay $T_{\rm cp}$ is due to the adders and multipliers in the **F** block and is given by

$$T_{\rm cp} = T_m + NT_{\rm sum} + B_{\rm ADD}T_{\rm carry}. \tag{9}$$

Note that, if some of the taps in the **F** block are powered down, then $T_{\rm cp}$ in (9) will be smaller. Further energy savings can be obtained by lowering the supply voltage via schemes such as in [19], [31], and [32], as described below.

B. Variable Supply Voltage for Low Power

If T_s is the sample period and $T_{\rm cp,max}$ is the critical path delay when all the taps are powered up, then we choose

$$T_{\rm cp,max} = T_s. \tag{10}$$

If some of the taps are powered down, then $T_{\rm cp} < T_{\rm cp,max}$. The slack in critical path delay can be exploited to obtain energy savings. The propagation delay for CMOS circuits has an inverse relationship with the supply voltage, while the power dissipation has a quadratic relationship with the supply voltage. Therefore, the reduced critical path delay due to reconfiguration can be exploited to reduce the supply voltage in order to save energy. In fact, the slack is equivalent to the processing rate r [19], which is defined as

$$r = \frac{T_{\rm cp}}{T_{\rm cp,max}} = \frac{T_{\rm cp}}{T_s} \tag{11}$$

where $0 \leq r \leq 1$. Thus, $r = 1$ when $T_{\rm cp} = T_{\rm cp,max} = T_s$, indicating that the processing rate of the architecture is maximum. It can be shown [19] that the supply voltage can be lowered to $V_{\rm dd}(r) \leq V_{\rm dd,max}$, where $V_{\rm dd}(r)$ is given by

$$V_{\rm dd}(r) = V_t + \frac{r}{2}V_o + \sqrt{\frac{r^2}{4}V_o^2 + rV_tV_o} \tag{12}$$

and $V_o = (V_{\rm dd,max} - V_t)^2/V_{\rm dd,max}$, V_t is the threshold voltage and $V_{\rm dd,max} = V_{\rm dd}(1)$. In [19], the value of r is obtained experimentally by buffering the input samples in a first-in first-out (FIFO) buffer and monitoring the number of unprocessed inputs in the buffer. A specific supply voltage $V_{\rm dd}$ value is then chosen according to (12). In contrast to [19], we determine r algorithmically from (11), where the critical path delay $T_{\rm cp}$ is varied (by powering down the specific taps) according to the requirements imposed by the input environment. It should be mentioned that (12) is employed to obtain a coarse $V_{\rm dd}$ value. Fine adjustments [31] to track the process and temperature variations can be done subsequently. The reader is referred to [31] and [32] for details and additional references.

Thus, energy savings are obtained by powering down taps, reducing precisions and reducing the supply voltage $V_{\rm dd}(r)$. In order to enable these energy savings, we need an underlying reconfigurable hardware fabric and the energy dissipation models for the hardware, which are described below.

III. Reconfigurable Datapath: The SPA Block

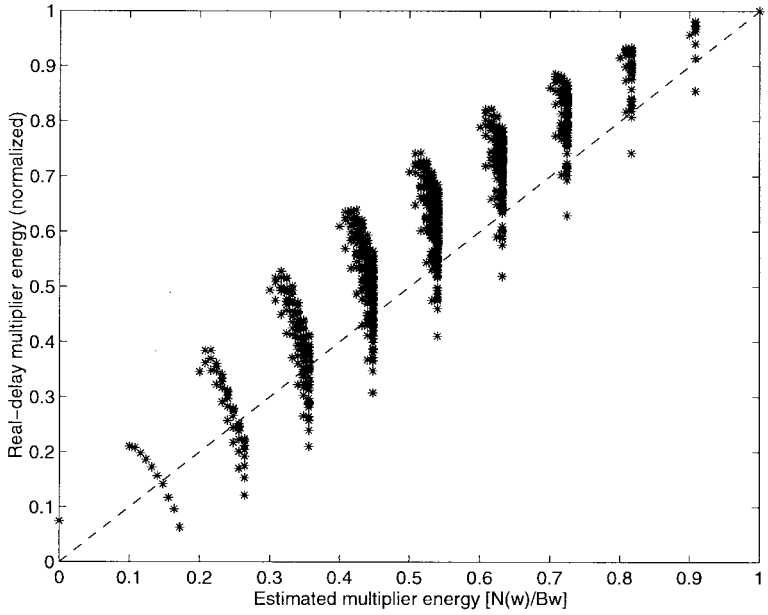
In this section, we present reconfigurable hardware and energy models for arithmetic units and filters. The reconfigurable hardware architecture and energy models are formulated so that energy-optimum reconfiguration strategies (described in Section IV) can be computed in real time. We will focus on energy models for the multipliers as these consume a large percentage of the total energy. It is well known that energy dissipation is a function of the input statistics in CMOS circuits. For a direct-form FIR filter, the input $x(n-k+1)$ into the kth-tap multiplier is a delayed copy of $x(n)$. Thus, the statistics of the data input are the same for all taps. Therefore, we present an energy dissipation model of a multiplier, which is a function of the coefficient input only.

A. Multiplier Energy Models

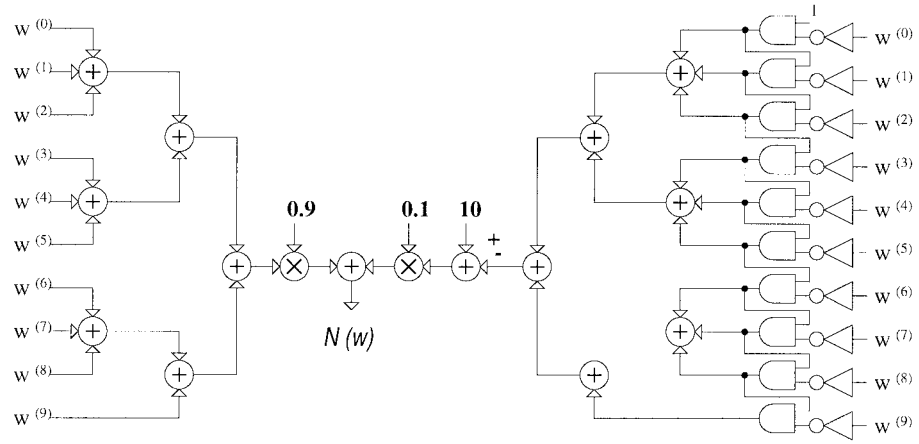
We will assume that a B_x-b signal $x(n)$ is being multiplied by a B_w-b constant coefficient w. The constant coefficient assumption is valid for adaptive filters if we assume that the **WUD**-block is powered-down after convergence. The multiplier energy model is based on two estimates of the transition activity in an array multiplier [33]. The first estimate is given by number of ones in a binary representation of the coefficient w. The justification for this estimate is that if the jth bit in the coefficient is zero, then the transition activity in the jth row of the array multiplier is reduced. In [34], a similar expression was employed in estimating the energy dissipation of a shift–add multiplier. Let $w = -w^{(0)} + \sum_{j=1}^{B_w-1} w^{(j)}2^{-j}$ be the two's complement representation of the coefficient w. The number of ones $\mathcal{N}_1(w)$ is then given by

$$\mathcal{N}_1(w) = \sum_{j=0}^{B_w-1} w^{(i)}. \tag{13}$$

A linear model based on $\mathcal{N}_1(w)$ was regressed against *real-delay* energy consumption values obtained via the gate-level simulation tool MED [35] with typical delay values for a 0.18-μm 2.5-V CMOS technology. It was found that $\mathcal{N}_1(w)$ underestimates transition activity in the array multiplier by an error of approximately 14%. This is due to the fact that, in



(a)



(b)

Fig. 3. Estimate of multiplier energy based on (15). (a) Model accuracy. (b) Hardware implementation for real-time evaluation.

this model, the transition activity in a row due to the signal propagation from the previous row is ignored.

The second estimate is based on the number of zeros in the least significant bit (LSB) positions in the two's complement representation of w. For example, $w = [0100]$ has two zeros at the LSB positions. It can be shown that, in an array multiplier, the rows corresponding to these zeros have very little transition activity. Therefore, the energy model is given by the difference of the coefficient precision and the number of LSB zeros. This number $\mathcal{N}_2(w)$ is computed as

$$\mathcal{N}_2(w) = B_w - \sum_{j=0}^{B_w-1} \prod_{i=j}^{B_w-1} \left(1 - w^{(i)}\right) \tag{14}$$

where $(1 - w^{(i)})$ is the bit-wise complement of $w^{(i)}$. Regression of a linear model based on $\mathcal{N}_2(w)$ with real-delay energy values based on MED [35] indicated that (14) overestimates the multiplier energy by 75%.

As the two estimates (13) and (14) underestimate and overestimate the transition activity, respectively, an accurate energy model can be obtained by taking their weighted sum as follows:

$$\mathcal{N}(w) = 0.9\mathcal{N}_1(w) + 0.1\mathcal{N}_2(w), \quad \mathcal{E}_m(w) = \mathcal{E}_{\max}\frac{\mathcal{N}(w)}{B_w} \tag{15}$$

where $\mathcal{N}_1(w)$ and $\mathcal{N}_2(w)$ are obtained from (13) and (14), respectively. In Fig. 3(a), we show regression of the model in (15) against the real-delay energy consumption values. It was found that the model in (15) is accurate with less than 9% error as compared to a real-delay gate-level simulation.

An architecture-level implementation of the hardware that evaluates the energy model in (15) is shown in Fig. 3(b), where $w^{(i)}$ is the ith b of coefficient w. If such energy models for the **SPA** hardware are not available, then a look-up table (ROM) can be used for storing all possible energy values. Note that the models based on closed-form expressions such as (15) are useful in determining the energy-optimum configurations, but are not used to estimate the energy savings.

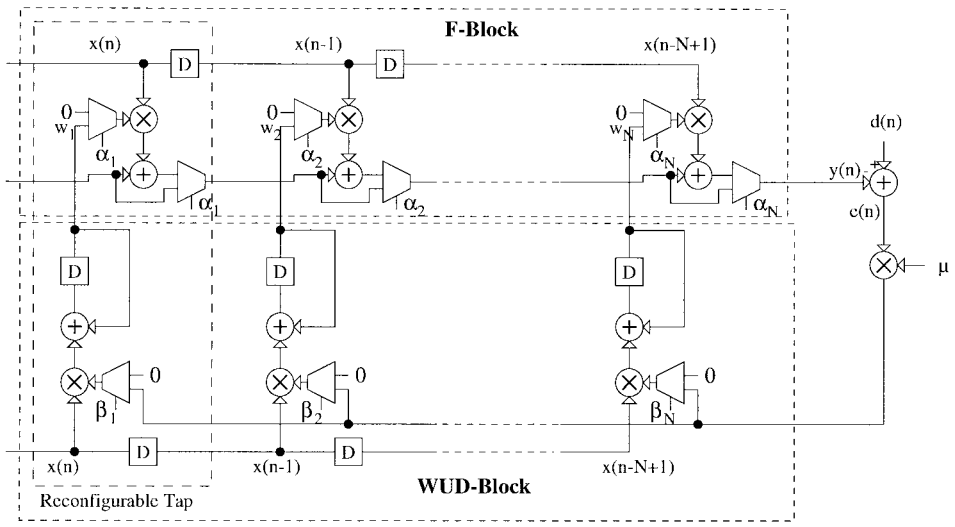


Fig. 4. Reconfigurable adaptive-filter architecture.

B. Reconfigurable Adaptive Filters

In Fig. 4, we present a reconfigurable architecture for an adaptive filter, where we have modified the architecture in Fig. 2 by introducing control signals α_k and β_k, which power up/down the kth tap. For example, setting $\alpha_k = 0$ forces a zero at the input to the **F**-block multiplier of the kth tap and bypasses the **F**-block adder. Additional energy savings are possible by zeroing out or freezing $x(n-k)$. This is not done because it was found via a real-delay gate-level simulation that the energy dissipation in such a case is less than 5% of the average multiplier energy. In addition, it is assumed that reconfigurations in Fig. 4 occur once every L samples, where L is large (e.g., $L = 8192$ for the NEXT-canceller application). Therefore, energy dissipation due to the transition in the multiplier coefficient input from w_k to 0 is ignored. For the applications, where fast reconfigurations are required, it might be better to instead hold the inputs to the previous value and force the output to zero. This can be done by replacing multiplexers at the multiplier inputs by pass transistors. Such an approach will also be useful for folded architectures (e.g., in fractionally spaced equalizers). If needed, additional multiplexers can be employed to power down the adder in **F**-block. Thus, the kth tap in the **F**-block can be powered down by setting $\alpha_k = 0$. Similarly, $\beta_k = 0$ powers down the multiplier in the kth tap of the **WUD** block.

Additional energy savings can be achieved by changing the precisions of the input signal and coefficients. In [36], the coefficient precision was varied by providing a gain at the output of the filter. By changing the gain, the adaptive filter can be made to converge to new coefficients with smaller precision. In this paper, we adjust the precision by forcing the LSB's to a value of "0." The advantage of this scheme is its quick convergence and simplicity of implementation. Forcing the LSB's to "0" does not vary the coefficients by a large amount. Thus, the adaptive filter tracks the new optimum coefficients (at optimum precision) very quickly. In Section IV, we will prove that the coefficient precision is a logarithmic function of the number of powered-up taps, and the precision requirement do not change by more than 2 b. Therefore, the reconfiguration of precision can be implemented with a very small hardware overhead consisting of two AND gates for each tap.

The energy dissipation of an N-tap reconfigurable adaptive filter employed in the **SPA** block is given by

$$\mathcal{E}_{\mathbf{SPA}} = \mathcal{E}_{\mathbf{F}} + \mathcal{E}_{\mathbf{WUD}} \tag{16}$$

where $\mathcal{E}_{\mathbf{F}}$ and $\mathcal{E}_{\mathbf{WUD}}$ are the energy dissipations for the **F** and **WUD** blocks, respectively. It can be seen that $\mathcal{E}_{\mathbf{F}}$ is given by

$$\mathcal{E}_{\mathbf{F}} = \sum_{k=1}^{N} \alpha_k \mathcal{E}_m(w_k) \tag{17}$$

where $\mathcal{E}_m(w_k)$ is the energy dissipation of the multiplier in the **F**-block and kth tap. Similarly, the energy dissipation of the **WUD** block is given by

$$\mathcal{E}_{\mathbf{WUD}} = \sum_{k=1}^{N} \beta_k \mathcal{E}_{m,\mathrm{wud}} \tag{18}$$

where $\mathcal{E}_{m,\mathrm{wud}}$ is the energy dissipation of a weight–update block multiplier. It is worth mentioning that two inputs to all the multipliers in the **WUD**-block (see Fig. 2) have the same statistics and, therefore, all **WUD**-block multipliers are assumed to consume the same energy. In case of a sign–LMS algorithm and powers-of-two LMS algorithm, $\mathcal{E}_{m,\mathrm{wud}}$ can be replaced by the energy consumption of a shifter. As there is no need to update a tap if it is powered down $(\alpha_k = 0)$ or if the filter has converged, then in these two cases, we will force $\beta_k = 0$. Therefore, after convergence, the critical path delay T_{cp} in the **F**-block is given by

$$T_{\mathrm{cp}} = T_m + \sum_{k=1}^{N} \alpha_k T_{\mathrm{sum}} + B_{\mathrm{ADD}} T_{\mathrm{carry}} + N T_{\mathrm{mux}} \tag{19}$$

where T_{mux} is the computation delay of a two-to-one multiplexer (2×1 mux). The critical path delay in (19) will be maximum when all the taps in adaptive filter are powered

up. The maximum critical path delay $T_{\rm cp,max}$ is obtained by substituting $\alpha_k = 1$ $(k = 1, 2, \cdots, N)$ in (19) and is given by

$$T_{\rm cp,max} = T_m + N(T_{\rm sum} + T_{\rm mux}) + B_{\rm ADD}T_{\rm carry}. \quad (20)$$

However, when some of the taps in the **F**-block are powered down (i.e., $\alpha_k = 0$ for some k) then $T_{\rm cp}$ in (19) will be smaller than $T_{\rm cp,max}$ in (20). The reduced critical path delay for these cases can be exploited to save energy further by lowering the supply voltage by an appropriate amount, as described in Section II-B. It is worth noting that the effectiveness of critical path change is dependent on the relative values of $T_{\rm sum}$ to other terms in (20).

C. Complex Adaptive Filters

In this section, we consider complex adaptive filters with complex-valued coefficients $w_k = c_k + jd_k$ (where c_k and d_k are real and imaginary parts of w_k) processing the complex-valued data $x(n) = x_r(n) + jx_i(n)$ (where $x_r(n)$ and $x_i(n)$ are real and imaginary parts of $x(n)$). These filters are commonly employed in many communications systems including the NEXT canceller, to be described in Section V. In particular, we will employ the low-power strength-reduced **(SR)** [3] architecture shown in Fig. 5. This architecture can be derived by viewing complex filtering as polynomial multiplication and then applying the strength-reduction transformation [15] at the algorithmic level. The reconfigurable **SR** adaptive filter architecture can be derived from Figs. 4 and 5. The energy dissipation of the **SR** adaptive filter is given by

$$\mathcal{E}_{\rm SR} = \sum_{k=1}^{N} \alpha_k \Big[\mathcal{E}_m(c_k + d_k) + \mathcal{E}_m(d_k) + \mathcal{E}_m(c_k - d_k)\Big] + 3\sum_{k=1}^{N} \beta_k \mathcal{E}_{m,\rm wud} \quad (21)$$

where $\alpha_k = 0$ and $\beta_k = 0$ indicates that the multipliers in the kth tap of the **F** and **WUD** blocks are powered down.

In Section IV, we develop energy-optimum reconfiguration strategies for the reconfigurable adaptive filters presented in this section.

IV. DAT

In this section, we present DAT for low-power reconfigurable signal processing, specifically for adaptive filters. The motivation for DAT is that the worst-case scenario is usually not the nominal scenario. Hence, significant energy efficiencies can be gained by having an **SMA** block (see Fig. 1) that monitors the input state and then reconfigures the **SPA** block. This naturally leads to the definition of the input and configuration state–spaces discussed in Sections IV-A and IV-B, respectively. These definitions are then employed in formulating an energy optimization problem discussed in Section IV-C. The solution to this problem is derived in Section IV-D, which results in an energy-optimal reconfiguration strategy. In Section IV-E, we compute the energy savings due to DAT. The DAT technique is employed in a system identification example in Section IV-F.

A. Input State–Space

We model the nonstationarities in the input via the definition of an *input state–space* as follows.

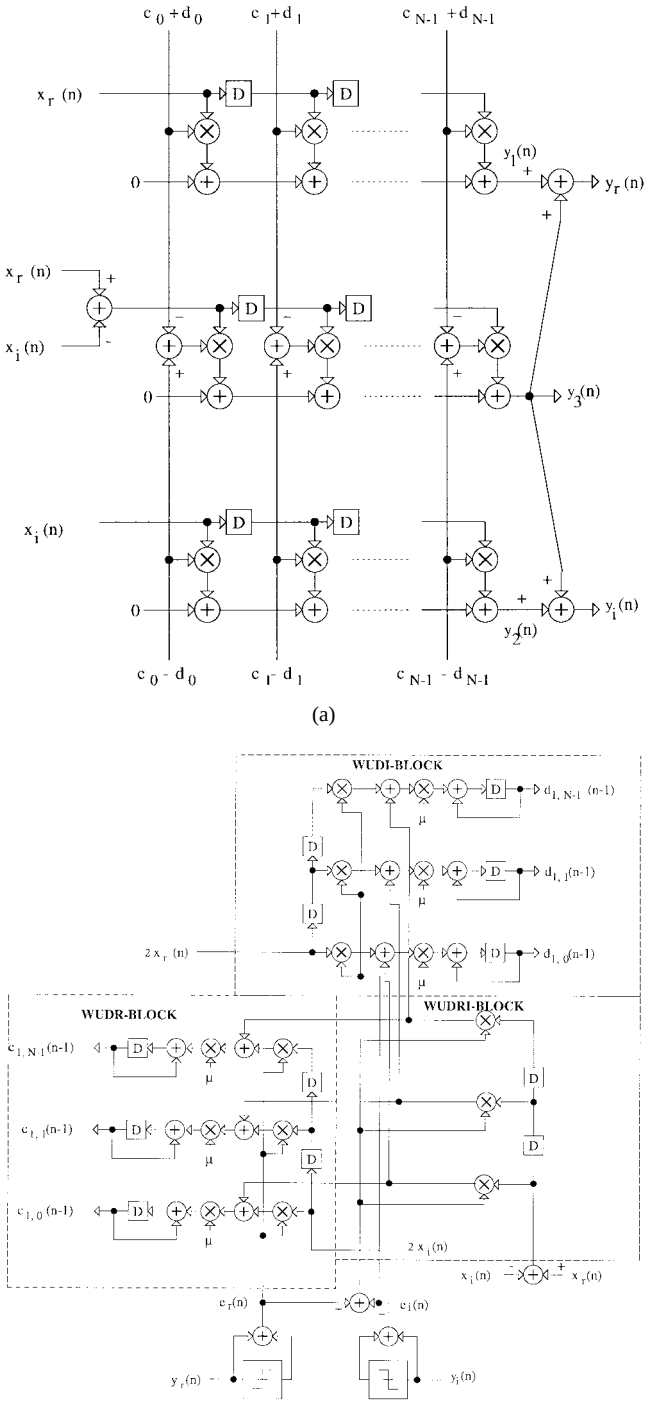


Fig. 5. Complex adaptive filter. (a) **F**-block of **SR**. (b) **WUD**-block of **SR**.

Definition 1: The **input state–space** $\mathcal{S} \triangleq \{\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_{N_s}\}$, where $\mathbf{s}_i$ is a vector of input-dependent parameters. The input state at time instant n, $\mathbf{s}(n)$, will be in state $\mathbf{s}_i$ (i.e., $\mathbf{s}(n) = \mathbf{s}_i$) with a probability $p(\mathbf{s}_i)$.

As an example, consider the case of a 155.52-Mb/s ATM–LAN networked office building with 100 workstations, out of which 80 workstations are approximately 70 m from the closet and the remaining are distributed equally between 40 and 100 m. In this case, the state–space $\mathcal{S}$ will have three elements, i.e., $\mathbf{s}_1$, $\mathbf{s}_2$, and $\mathbf{s}_3$ and the probability of occurrence $p(\mathbf{s}_1) = 0.1$, $p(\mathbf{s}_2) = 0.8$, and $p(\mathbf{s}_3) = 0.1$. The elements of

vector $\mathbf{s}_i$ would be the parameters of interest in the design of an ATM–LAN receiver. For example, one parameter of interest is the peak-to-average ratio (PAR), which is defined as the ratio of the peak and the root-mean-squared values of the input signal. Thus, when the input state $\mathbf{s}(n) = \mathbf{s}_i$, the corresponding PAR (denoted as PAR_i) is useful in determining the input precision $B_{x,\mathrm{opt}}$. Thus, $\mathbf{s}_i = [\sigma_{x_i}^2, \mathrm{PAR}_i, \mathrm{SNR}_{\mathbf{in}_i}]$, where $\sigma_{x_i}^2$, PAR_i, and $\mathrm{SNR}_{\mathbf{in}_i}$ are the input signal energy, input PAR, and input SNR, respectively. Hence, the state–space components are given by

$$\mathbf{s}_1 = [1.9 \text{ dB}, 10.3 \text{ dB}, 12.1 \text{ dB}]$$
$$\mathbf{s}_2 = [3.5 \text{ dB}, 9.9 \text{ dB}, 14.9 \text{ dB}]$$
$$\mathbf{s}_3 = [7.1 \text{ dB}, 9.1 \text{ dB}, 18.2 \text{ dB}].$$

In Section V, we will employ $\mathbf{s}_i = [\sigma_{x_i}^2]$ to model different lengths of the cable for 155.52-Mb/s ATM–LAN. Usually, $\mathbf{s}(n)$ is monitored over a window of L samples.

B. Configuration State–Space

A reconfigurable hardware fabric is characterized by its *configuration state–space*, which is defined below.

Definition 2: The **configuration state–space** $\mathcal{C} \triangleq \{\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_{N_c}\}$, where $\mathbf{c}_i$ is a vector of reconfiguration control signals. The hardware fabric will be in configuration $\mathbf{c}(n)$ at a given time instant n where $\mathbf{c}(n) \in \mathcal{C}$.

For an N-tap reconfigurable adaptive filter (see Fig. 4), the configuration vector can be defined as

$$\mathbf{c} = \left[\underline{\alpha}, \underline{\beta}, \mathbf{B}_w, \mathbf{B}_x, \mathbf{B}_V\right]$$

where $\underline{\alpha}$ and $\underline{\beta}$ are N-b control words containing *specific values* of the control signals α_k and β_k, respectively, in Fig. 4. Similarly, $\mathbf{B}_w$ and $\mathbf{B}_x$ are control words indicating the coefficient precision and data precisions, respectively. Finally, $\mathbf{B}_V$ is the control word indicating the value of the supply voltage V_{dd}. For example, if $N = 8$, $B_w = 8$ bits, $B_x = 4$ bits, and V_{dd} can have values from 1.5 to 2.5 V with a step of 0.5 V, then $\underline{\alpha}$, $\underline{\beta}$, $\mathbf{B}_w$, $\mathbf{B}_x$, and $\mathbf{B}_V$, are control words with eight, eight, three, two, and two bits, respectively. Thus, $\mathbf{c}$ is a control word with 23 bits and the number of configurations $N_c = 2^{23}$.

The above example indicates how easily the number of configurations explodes with an increase in reconfigurable parameters. We are interested in determining the energy-optimum configuration $\mathbf{c}_{\mathrm{opt}}(\mathbf{s}_i)$ for every input state $\mathbf{s}_i$ from a total of N_c possible configurations while satisfying an MSE constraint. Due to the large number of possible configurations N_c, it becomes important to develop a systematic approach to obtaining $\mathbf{c}_{\mathrm{opt}}$. In order to develop such an approach, we formally define the energy-optimum configuration $\mathbf{c}_{\mathrm{opt}}(\mathbf{s}_i)$ as follows.

Definition 3: The **energy-optimum configuration** $\mathbf{c}_{\mathrm{opt}}(\mathbf{s}_i) \in \mathcal{C}$ for a given input state $\mathbf{s}_i \in \mathcal{S}$ is defined as

$$\mathbf{c}_{\mathrm{opt}}(s_i) = \arg\min_{\mathbf{c}\in\mathcal{C}} \mathcal{E}_{\mathbf{SPA}}(\mathbf{c}), \qquad \text{s.t. } \mathcal{J}_{\mathbf{SPA}}(\mathbf{c}, \mathbf{s}_i) \le \mathcal{J}_o \tag{22}$$

where $\mathcal{E}_{\mathbf{SPA}}(\mathbf{c})$ is energy consumed by the **SPA** block in configuration $\mathbf{c}$, $\mathcal{J}_o$ is the specified MSE and $\mathcal{J}_{\mathbf{SPA}}(\mathbf{c}, \mathbf{s}_i)$ is the MSE achieved by the **SPA** block when the input is in state $\mathbf{s}_i$ and the **SPA** block is in configuration $\mathbf{c}$.

If the **SPA** block consists of the adaptive filter in Fig. 4, then $\mathcal{E}_{\mathbf{SPA}}(\mathbf{c})$ is given by (16) (for a real adaptive filter) or (21) (for a complex adaptive filter) and $\mathcal{J}_{\mathbf{SPA}}(\mathbf{c}, \mathbf{s}_i)$ is given by

$$\mathcal{J}_{\mathbf{SPA}}(\mathbf{c}, \mathbf{s}_i) = \sigma_d^2 - \sum_{k=1}^{N} \alpha_k |w_k|^2 \sigma_x^2 \tag{23}$$

where σ_d^2, σ_x^2, and $|w_k|^2$ are as defined in Section II-A. Thus, in (22), the objective function is tied to the hardware fabric and the constraint is dependent upon the application at hand. Next, we formulate the energy optimization problem whose solution will result in a energy-optimum reconfiguration strategy.

C. Energy Optimization Problem

In its most general form, the energy optimization problem can be written as

$$\min_{\mathbf{c}\in\mathcal{C}} \sum_{i=1}^{N_s} \mathcal{E}_{\mathbf{SPA}}(\mathbf{c}) p(\mathbf{s}_i), \qquad \text{s.t. } \mathcal{J}_{\mathbf{SPA}}(\mathbf{c}, \mathbf{s}_i) \le \mathcal{J}_o, \forall \mathbf{s}_i \in \mathcal{S}. \tag{24}$$

Note that the optimization problem in (24) is independent of the hardware platform, i.e., one could potentially have a platform based on an FPGA, programmable digital signal processor (software DAT) or a multiprocessor [17]. This is because the hardware-specific parameters can be incorporated via the definition of $\mathcal{E}_{\mathbf{SPA}}(\mathbf{c})$ and the configuration state–space $\mathcal{C}$. In this paper, we have focused on a dedicated implementation of a reconfigurable DSP. We simplify the problem in (24) by solving it independently for each input state $\mathbf{s}_i$. The energy optimization problem for the reconfigurable adaptive filter in Fig. 4 is given by

$$\min_{\alpha_k\in\{0,1\}} \sum_{k=1}^{N} \alpha_k \mathcal{E}_m(w_k), \qquad \text{s.t. } \sigma_d^2 - \sum_{k=1}^{N} \alpha_k |w_k|^2 \sigma_x^2 \le \mathcal{J}_o. \tag{25}$$

Note that we do not include the **WUD**-block reconfiguration signals β_k's in the optimization problem because we assume that $\beta_k = 0$ after the adaptive filter has converged, i.e., the **WUD**-block is powered down. Next, we determine the solution to (25) via the Lagrange multiplier method [37] to obtain a practical reconfiguration strategy.

D. Energy-Optimum Reconfiguration Strategy

A block-level diagram of a DAT-based adaptive filter is shown in Fig. 6. The **SPA** block has the architecture shown in Fig. 4 and the **SMA** block computes the energy-optimum reconfiguration strategy (to be derived in this section) for the **SPA** block.

The first sub-block in the **SMA** block detects the value of the input state $\mathbf{s}(n)$. The energy-optimum values of the configurable parameters α_k, β_k $(k = 1, 2, \cdots, N)$, B_w, B_x, and V_{dd} are then computed in other sub-blocks as a solution to the energy optimization problem (25) presented in Section IV-C. In order to keep the reconfiguration strategy simple, we

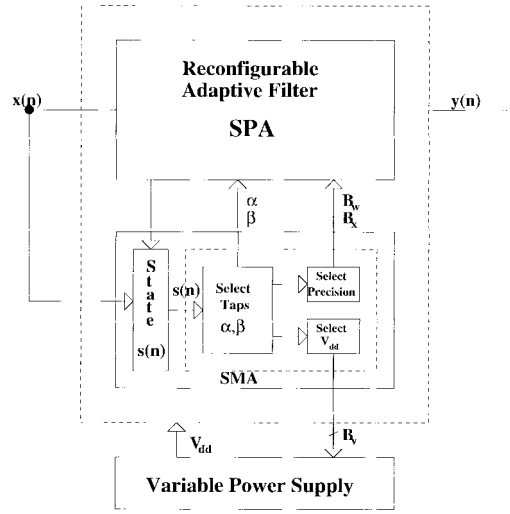


Fig. 6. DAT-based reconfigurable adaptive signal-processing system.

propose to solve the optimization problem in (25) through the following three steps.

Step 1) Determine the optimum values of control signals $\alpha_{k,\mathrm{opt}}$ and $\beta_{k,\mathrm{opt}}$ via the Lagrange multiplier method [37].

Step 2) Determine the optimum precisions $B_{w,\mathrm{opt}}$ and $B_{x,\mathrm{opt}}$ via (31) and (36), respectively.

Step 3) Determine the optimum supply voltage $V_{\mathrm{dd,opt}}$ from (11) and (12) and (19) and (20).

Note that it is possible to stop at any of the steps indicated above to obtain increasingly suboptimal low-energy solutions. We next describe the above three steps in more detail.

1) Energy-Optimum Choice of α_k and β_k (Step 1): We define a dual optimization problem as follows:

$$\min_{\alpha_k \in \{0,1\}} \mathcal{L}(\alpha_1, \alpha_2, \cdots, \alpha_N, \lambda^*) \tag{26}$$

where λ^* is a constant referred as the Lagrange multiplier [37] and

$$\mathcal{L}(\alpha_1, \alpha_2, \cdots, \alpha_N, \lambda) = \sum_{k=1}^{N} \alpha_k \mathcal{E}_m(w_k) + \lambda \left(\sigma_d^2 - \mathcal{J}_o - \sum_{k=1}^{N} \alpha_k |w_k|^2 \sigma_x^2 \right) \tag{27}$$

where $\lambda \geq 0$ is a constant multiplier, w_k is the kth coefficient, and $\mathcal{E}_m(w_k)$ is the energy of a multiplier that has w_k at one of the inputs.

It can be shown [38] that the solution to (26) is given by

$$\alpha_{k,\mathrm{opt}} = \begin{cases} 1, & \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} \geq \tilde{\lambda}^* \\ 0, & \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} < \tilde{\lambda}^* \end{cases} \tag{28}$$

where $\tilde{\lambda}^*$ is a constant. Equation (28) indicates that it is better to power down the taps with small values of the *energy-normalized metric* E_k, where E_k is defined as

$$E_k = |w_k|^2 / \mathcal{E}_m(w_k). \tag{29}$$

Intuitively, small values of E_k imply that the kth tap contributes less to the SNR (as w_k is small), but consumes more energy ($\mathcal{E}_m(w_k)$ is large). In practice, we do not need to compute the constant $\tilde{\lambda}^*$ in (28) if we employ the reconfiguration strategy of powering down the taps, starting with those with the smallest value of E_k, until the MSE constraint is violated. It is worth mentioning that the solution to Step 1 depends upon the multiplier architecture. The optimum value for β_k $(k = 1, 2, \cdots, N)$ is chosen as zero if either $\alpha_{k,\mathrm{opt}} = 0$ or the filter has converged.

If $\mathcal{E}_m(w_k)$ is assumed to be independent of w_k, then the energy-optimum reconfiguration strategy would be to switch off taps with the smallest coefficients. Thus, the strategy proposed in [18] for low-pass filters (LPF's) falls out as a special case of (28). Note that the reconfiguration in (28) is different from the case where only the end taps are powered down [36]. Finally, our strategy has been derived by solving an optimization problem and, hence, is guaranteed to be energy optimal under the SNR constraint. In Section V, we will demonstrate the application of this strategy to a NEXT canceller in a 155.52-Mb/s ATM–LAN transceiver.

2) Energy-Optimum Choice of Precisions (Step 2): Let the coefficients w_k in the **F**-block be represented by B_w bits. Assuming a linear stochastic model for the fixed-point error in the coefficients w_k, the MSE for an N-tap fixed-point FIR filter $\mathcal{J}_{\mathrm{fx}}$ is given by [39]

$$\mathcal{J}_{\mathrm{fx}} = \mathcal{J}_{\mathrm{fl}} + \frac{N \sigma_x^2 2^{-2B_w}}{12} \tag{30}$$

where the second term in (30) is the coefficient quantization error due to the fixed-point implementation and $\mathcal{J}_{\mathrm{fl}}$ is the MSE for the floating-point algorithm.

From (30), we see that in order to maintain a fixed quantization error, the required coefficient precision decreases with the filter length. Therefore, coefficient precision can be reduced when the taps get powered down in the reconfigurable datapath. From (28) and (30), the optimum coefficient precision $B_{w,\mathrm{opt}}$ is obtained as follows:

$$B_{w,\mathrm{opt}} = B_w + \frac{1}{2} \log_2 \left(\frac{1}{N} \sum_{k=1}^{N} \alpha_{k,\mathrm{opt}} \right) \tag{31}$$

where B_w is the maximum precision required when all the taps are powered up. The reduction in precision [see (31)] with the number of taps is very small. In fact, one bit reduction in the precision is achieved for each four times reduction in filter length.

Similarly, the input precision $B_{x,\mathrm{opt}}$ can be determined from the expression for the signal-to-quantization noise ratio at the input (SQNR). Let $x(n)$ be the input signal with the maximum value $x_{\max}$ and mean squared value σ_x^2. Assuming that we employ B_x bits to quantize $x(n)$ and that the quantization noise $q_x(n)$ is a uniformly distributed signal over the interval $[-\Delta/2, \Delta/2)$ where $\Delta = -2x_{\max}/2^B$, we obtain the quantization noise power $E[|q_x(n)|^2]$ as follows:

$$E\left[|q_x(n)|^2\right] = \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} |q_x|^2 \, dq_x = \frac{\Delta^2}{12} = \frac{x_{\max}^2}{3 \times 2^{2B}}. \tag{32}$$

From (32), the SQNR (dB) can be obtained as

$$\begin{aligned}\text{SQNR (dB)} &= 10\log_{10}\left[\frac{\sigma_x^2}{E\left[|q_x(n)|^2\right]}\right] \\ &= 10\log_{10}\left[3\times 2^{2B}\frac{\sigma_x^2}{x_{\max}^2}\right] \end{aligned} \tag{33}$$

which can be further simplified to obtain

$$\text{SQNR (dB)} = 6B_x + 4.8 - \text{PAR (dB)} \tag{34}$$

where PAR (dB) is the PAR at the input, and is defined as

$$\text{PAR (dB)} = 20\log_{10}\left[\frac{x_{\max}}{\sigma_x}\right]. \tag{35}$$

For an equalizer, we can assume that an automatic gain control (AGC) block normalizes the input signal so that the input signal range $[-x_{\max}, x_{\max})$ matches that of the analog-to-digital converter (ADC). The root mean square (rms) value σ_x can then be computed by taking the square root of the time average of the squared input signal. The optimum precision can be computed from (34) as follows:

$$B_{x,\text{opt}} = B_x + \frac{\text{PAR (dB)} - \text{PAR}_{\max}\text{ (dB)}}{6} \tag{36}$$

where $\text{PAR}_{\max}$ is the worst-case PAR and B_x is the maximum input precision. We keep the data precision fixed in the NEXT canceller for a 155.52-Mb/s ATM–LAN, as it was found that these do not change.

3) Energy-Optimum Choice of Supply Voltage V_{dd} (Step 3): In reconfigurable systems where variable supply voltage generators are available (such as in [31] and [32]), an energy-optimum value of V_{dd} (if computed) can be employed to provide a coarse initial estimate to the tracking loop. In this section, we demonstrate how the energy-optimum value for V_{dd} can be computed.

The critical path delay of an adaptive filter is obtained by substituting $\alpha_k = \alpha_{k,\text{opt}}$, $k = 1, 2, \cdots, N$ in (19). Next, we obtain the normalized processing rate r by substituting (19) in (11) as follows:

$$r = \frac{T_m + NT_{\text{mux}} + B_{\text{ADD}}T_{\text{carry}} + \sum_{k=1}^{N}\alpha_{k,\text{opt}}T_{\text{sum}}}{T_s} \tag{37}$$

where T_s is the sample period. The optimum supply voltage V_{dd} can now be obtained by substituting r into (12).

Thus, we have presented a practical reconfiguration strategy to determine the configuration parameters $\alpha_{k,\text{opt}}$, $\beta_{k,\text{opt}}$, $B_{x,\text{opt}}$, $B_{w,\text{opt}}$, and $V_{\text{dd,opt}}$ for a DAT-based adaptive filter, as shown in Fig. 6. In the following section, we compute energy savings due to a DAT-based system over a traditional worst-case design.

E. Energy Savings Via DAT

The average energy dissipation of the **SPA** block $\mathcal{E}_{\mathbf{SPA},\text{ave}}$ is given by

$$\mathcal{E}_{\mathbf{SPA},\text{ave}} = \sum_{j=1}^{N_s}\mathcal{E}_{\mathbf{SPA}}\Big(\mathbf{c}_{\text{opt}}(\mathbf{s}_j)\Big)p(\mathbf{s}_j) \tag{38}$$

where $\mathbf{c}_{\text{opt}}(\mathbf{s}_j)$ is the energy-optimum configuration corresponding to state $\mathbf{s}_j$ obtained as a solution to (24).

The **SMA** block is activated after L samples and that too only if there is a transition in the state. For L sufficiently large, the energy dissipation of the **SMA** block ($\mathcal{E}_{\text{SMA,ave}}$) will be negligible. The total average energy dissipation per sample of a DAT-based system is obtained as

$$\mathcal{E}_{\text{DAT,ave}} = \mathcal{E}_{\mathbf{SPA},\text{ave}} + \mathcal{E}_{\text{SMA,ave}} \tag{39}$$

where, in general, $\mathcal{E}_{\text{SMA,ave}}$ will be negligible as compared to $\mathcal{E}_{\mathbf{SPA},\text{ave}}$. The average energy savings ($\mathcal{E}_{\text{sav}}$) due to a DAT-based system is given as

$$\mathcal{E}_{\text{sav}} = \frac{\mathcal{E}_{\text{WC}} - \mathcal{E}_{\text{DAT,ave}}}{\mathcal{E}_{\text{WC}}} \times 100\% \tag{40}$$

where $\mathcal{E}_{\text{WC}}$ is the energy dissipation of the worst-case design. In fact, $\mathcal{E}_{\text{WC}}$ equals the maximum value that $\mathcal{E}_{\mathbf{SPA}}(\mathbf{c}_{\text{opt}}(\mathbf{s}_j))$ in (38) can assume over all possible states $\mathbf{s}_j$.

F. Example: System Identification

In this subsection, we apply DAT to a system identification example in order to demonstrate the concepts presented thus far. The problem is to estimate the impulse response of an unknown system via an adaptive filter, as shown in Fig. 7(a). The unknown system can represent an echo path in a voice-band modem or a crosstalk path in high-speed data modem, where the adaptive filters are usually employed to identify the unknown echo/crosstalk signal and then cancel it. As the number of taps required by the adaptive filter varies with the unknown system, hence, DAT-based approaches can achieve significant energy savings over the traditional designs based upon worst-case assumptions.

We assume that the unknown system can be in five distinct states (i.e., $N_s = 5$) with impulse responses for each of the states, as shown in Fig. 7(b). Such impulse responses can be encountered in a wireless channel with multipath fading. We assume that probability of occurrence of these states is given by $p(\mathbf{s}_1) = 0.1$, $p(\mathbf{s}_2) = 0.2$, $p(\mathbf{s}_3) = 0.4$, $p(\mathbf{s}_4) = 0.2$, and $p(\mathbf{s}_5) = 0.1$. The input signal $x(n)$ is uncorrelated with variance $\sigma_x^2 = 1$, and we assume a noise of variance $\sigma_n^2 = 0.0001$ (i.e., SNR = 40 dB) at the output of the unknown system. As the signal $e(n)$ in Fig. 7(a) represents the residual echo/crosstalk, we are, therefore, interested in minimizing $\mathcal{J}(n) = E[e^2(n)]$. Hence, we set the desired MSE, $\mathcal{J}_o = 0.01$.

For the DAT-based adaptive filter, assume an input precision of $B_x = 8$ bits, coefficient precision of $B_{w,\max} = 8$ bits, maximum number of taps equal to $N = 8$ taps, and maximum supply voltage of $V_{\text{dd,max}} = 5$ V. Also, assume the computational delays of the hardware blocks to be $T_m = 4$ ns, $T_{\text{mux}} = 0.1$ ns, $T_{\text{sum}} = 0.7$ ns, $T_{\text{carry}} = 0.6$ ns, and a sampling rate of 50 MHz. Further assume that the hardware platform permits the powering-up of taps via control signals $\alpha_k (k = 1, 2, \cdots, 8)$, reconfiguring the coefficient precision B_w and supply voltage V_{dd}. Table I shows the optimum configurations achieved by employing the strategy presented in Section IV-D. When the input state varies from $\mathbf{s}_1$ to $\mathbf{s}_5$, the number of powered-up taps decreases from 8 to 2, the

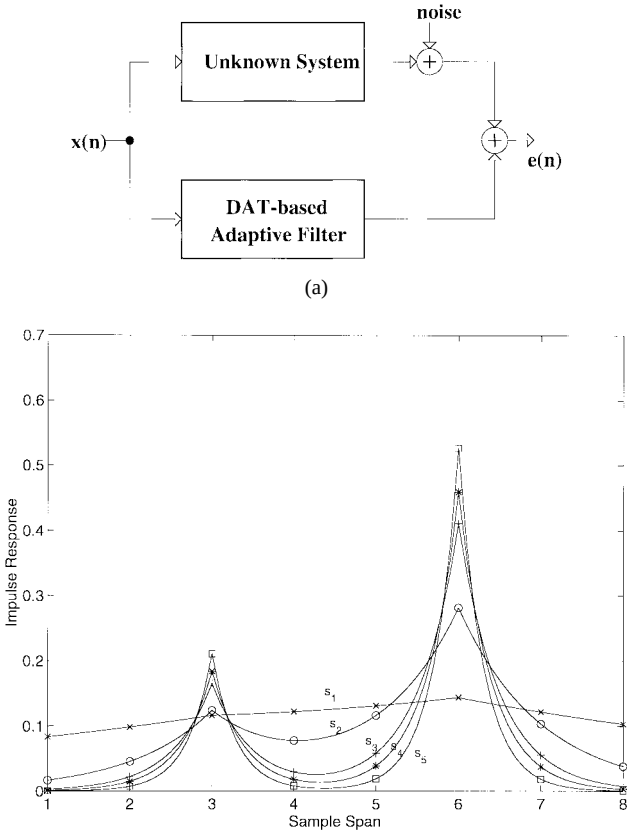


Fig. 7. System identification. (a) Block diagram. (b) Impulse responses of the unknown systems.

TABLE I
RESULTS FOR SYSTEM IDENTIFICATION

$\mathbf{s}_i$	$p(\mathbf{s}_i)$	α_{opt}	$B_{w,opt}$	$V_{dd,opt}$	$\mathcal{E}_{sav}$
$\mathbf{s}_1$	0.1	[1,1,1,1,1,1,1,1]	8 bits	5.0V	0%
$\mathbf{s}_2$	0.2	[0,1,1,1,1,1,1,1]	8 bits	4.9V	16%
$\mathbf{s}_3$	0.4	[0,0,1,1,1,1,1,0]	8 bits	4.6V	44%
$\mathbf{s}_4$	0.2	[0,0,1,0,1,1,1,0]	8 bits	4.4V	53%
$\mathbf{s}_5$	0.1	[0,0,1,0,0,1,0,0]	7 bits	4.2V	74%

coefficient precision varies from 8 to 7 bits, and the supply voltage $V_{\rm dd}$ varies from 5 to 4.2 V. Also note that $\alpha_{\rm opt}$ for state $\mathbf{s}_4$ and $\mathbf{s}_5$ in Table I, is given by $[0,0,1,0,1,1,1,0]$ and $[0,0,1,0,0,1,0,0]$, respectively, where 0 indicates that a tap is powered down, and 1 indicates that a tap is powered up. Thus, for states $\mathbf{s}_4$ and $\mathbf{s}_5$, the energy-optimum configuration requires powering down internal taps and, hence, could not have been obtained via existing approaches [18], [36].

The energy dissipation of the **SPA** block in each of the five states is also shown in Table I. The energy dissipation for the worst-case design $\mathcal{E}_{\rm WC}$ corresponds to state $\mathbf{s}_1$, which requires the maximum number of powered-up taps, the maximum precision, and the maximum supply voltage $V_{\rm dd}$. The energy savings, shown in Table I, are computed by employing (40). Energy savings range from 0% to 74% as the input state changes from $\mathbf{s}_1$ to $\mathbf{s}_5$. The average energy savings are computed via (40) and are shown in Table I. An average of 39% energy savings are achieved for this example. As can be seen, the average energy savings due to DAT depend upon the relative energy dissipation and probability of occurrence for the states corresponding to the nominal and the worst cases. Large energy savings can be expected for situations where $\mathcal{E}_{\rm WC} \gg \mathcal{E}_{\rm DAT,ave}$, and the state corresponding to the worst case is not very likely. In Section V, we employ the DAT-based adaptive filter as a NEXT canceller for a 155.52-Mb/s ATM–LAN.

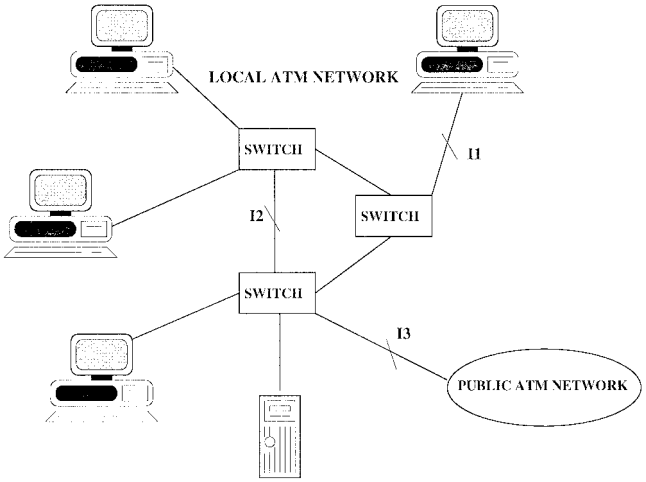


Fig. 8. ATM–LAN environment.

V. APPLICATION TO 155.52-MBITS/S ATM–LAN

In this section, we will study the performance of the proposed DAT-based system in a high-speed digital communication system. We will employ the DAT-based adaptive filter as a NEXT canceller for a data rate of 155.52 Mb/s over a UTP wiring [40].

Fig. 8 shows a vendor's view of an asynchronous transfer mode ATM-based LAN. The environment of interest for the UTP category-three (UTP-3) user network interface (UNI) consists of "I1" and "I2" interfaces (see Fig. 8). The wiring distribution system runs either from the closet to desktop or between hubs in the closets. The wiring employed consists mostly of either a TIA/EIA-568 UTP-3 four-pair cable or the DIW 10 Base-T 25-pair bundle. The propagation loss for these channels increases rapidly with an increase in the frequency of operation. Therefore, bandwidth efficient transmission schemes become necessary to support such high data rates over these channels. The carrierless amplitude phase (CAP) transmission scheme is such a scheme and is the standard [41] for 155.52-Mb/s ATM–LAN over UTP-3 wiring.

In the LAN environment, the two major causes of performance degradation for transceivers operating over UTP wiring are: propagation loss and crosstalk generated between adjacent wire pairs. The propagation loss that is assumed in system design is the worst-case loss given in the TIA/EIA-568 draft standard for category-3 cable [42]. This loss can be approximated by the following expression:

$$L_P(f) = 2.320\sqrt{f} + 0.238f \tag{41}$$

where the propagation loss $L_P(f)$ is expressed in decibels per 100 m and the frequency f is expressed in megahertz.

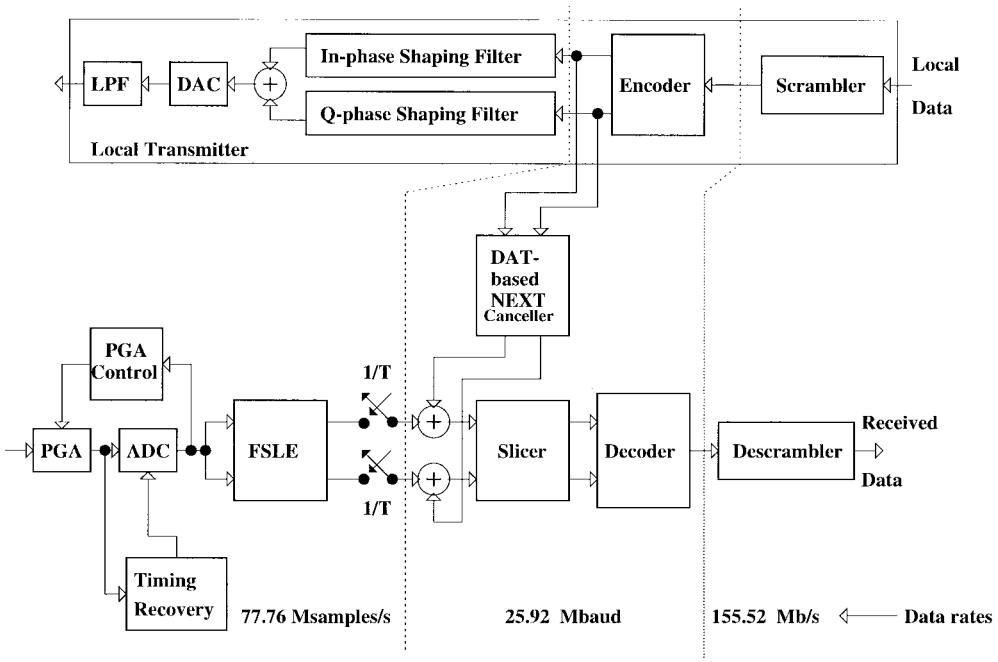


Fig. 9. 155.52-Mb/s ATM–LAN transceiver.

The worst-case NEXT loss model for a single interferer is also given in the TIA/EIA Draft Standard [42]. The squared magnitude of the NEXT transfer function corresponding to this loss can be expressed as

$$L_N(f) = 43 - 15\log f \tag{42}$$

where the frequency f is in megahertz, and $L_N(f)$ is expressed in decibels. In Section V-A, we briefly describe the CAP transceiver for a 155.52-Mb/s ATM–LAN. The interested reader is referred to [40] for further details. The simulation setup is described in Section V-B, while the simulation results are presented in Section V-C.

A. 155.52-Mb/s ATM–LAN Transceiver

The block diagram of a digital CAP transceiver is shown in Fig. 9. The bit stream to be transmitted is first passed through a scrambler. The scrambled bits are then fed into an encoder, which maps blocks of m bits onto one of $k = 2^m$ different complex symbols $a(n) = a_r(n) + ja_i(n)$ for a k-CAP line code. In this study, we have employed $k = 64$ because this value is necessary to limit the transmit spectrum to 30 MHz; a limit set by the Federal Communication Commission (FCC). The symbols $a_r(n)$ and $a_i(n)$ are processed by digital shaping filters. This requires that the shaping filters be operated at a sampling frequency f_s, which is at least twice the maximum frequency component of the transmit spectrum. The outputs of the filters are subtracted and the result is passed through a digital-to-analog (D/A) converter, which is followed by an interpolating LPF. It can be seen that most of the signal processing at the transmitter (including transmit shaping) is done in the digital domain, which permits a robust VLSI implementation.

In the receiver (see Fig. 9), the analog signal is first amplified by a programmable gain amplifier (PGA) whose gain is controlled by a digital PGA control block. The output of PGA is passed to an analog-to-digital (A/D) converter operating at 77.76 MHz, which converts the analog signal into a digital signal. The sampling instant of the A/D is controlled by a digital timing recovery block. The digital output of the A/D is processed by a fractionally spaced linear equalizer (FSLE). In addition, the local transmitted symbols are passed through a complex adaptive NEXT canceller, which tries to cancel the effect of the NEXT in the received signal. The algorithmic performance measure in this case is the SNR at the slicer $(\mathrm{SNR_{sl}})$, which is equal to the ratio of signal constellation power (which equals 42 for 64-CAP) to the MSE across the slicer. Hence, we have the following relation:

$$\mathrm{SNR_{sl}\,(dB)} = 10\log_{10}(42) - \mathcal{J}_{\mathrm{sl}}\,(\mathrm{dB}). \tag{43}$$

Henceforth, we employ $\mathrm{SNR_{sl}}$ as the algorithmic/system performance measure. For 64-CAP, a specification of $\mathrm{SNR_{sl}} = 29.45$ dB is sufficient to obtain a probability of error less than 10^{-10}.

The complexity requirements for the NEXT canceller increase as the cable length increases. Traditionally, the NEXT canceller is designed for the worst-case scenario, i.e., the longest cable length. However, for shorter cable lengths, the complexity of the NEXT canceller can be reduced and, thus, substantial energy savings can be achieved. In this experiment, we employ a DAT-based NEXT canceller to enable the energy savings possible due to the different cable lengths.

B. Simulation Setup

We assume a spatial variation in the length of the UTP-3 cable from 110 to 40 m (see Fig. 8) with the lengths having a Gaussian distribution with a mean of 75 m, as indicated in Table II. Usually, an estimate of the probability distribution of the cable lengths can be obtained from surveys, which currently are not available. Thus, the state–space $\mathcal{S}$ has eight states. In our simulations, we emulate the spatial variation of

TABLE II
RESULTS FOR DAT-BASED NEXT CANCELLER

$\mathbf{s}_i$	Cable length (m)	$p(\mathbf{s}_i)$	$\sum \alpha_{k,opt}$ (taps)	$B_{w,opt}$ (bits)	$V_{dd,opt}$ (V)	$\mathcal{E}_{sav}$ (%)
$\mathbf{s}_1$	110	0.05	30	12	2.5	−2
$\mathbf{s}_2$	100	0.10	22	12	2.4	32
$\mathbf{s}_3$	90	0.15	12	12	2.2	67
$\mathbf{s}_4$	80	0.20	8	12	2.1	78
$\mathbf{s}_5$	70	0.20	7	11	2.1	80
$\mathbf{s}_6$	60	0.15	6	11	2.1	82
$\mathbf{s}_7$	50	0.10	5	11	2.1	87
$\mathbf{s}_8$	40	0.05	4	11	2.0	89

cable length by varying the cable length in time. This exercise also demonstrates the performance of the DAT-based NEXT canceller in the presence of temporal variations in the cable length due to temperature changes.

The received signal power depends upon the attenuation of the channel and, hence, the cable length. Furthermore, the input SNR is also a function of the received signal power and this determines the performance of the receiver. We define the input states $\mathbf{s}_i$ (corresponding to the eight cable lengths in Table II) as

$$\mathbf{s}_i = \left[\sigma_{x_i}^2\right]$$

where $\sigma_{x_i}^2$ is the received signal power for the ith cable length. The state set $\mathcal{S}$ for the 155.52 Mb/s ATM–LAN can be found from (41) as follows:

$$\begin{aligned}\mathcal{S} &= [\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4, \mathbf{s}_5, \mathbf{s}_6, \mathbf{s}_7, \mathbf{s}_8] \\ &= [1.6 \text{ dB}, 1.9 \text{ dB}, 2.3 \text{ dB}, 2.8 \text{ dB}, 3.5 \text{ dB}, 4.4 \text{ dB}, \ 5.5 \text{ dB}, \\ &\quad 7.1 \text{ dB}].\end{aligned}$$

The changes in the input state can be detected by monitoring $\mathrm{SNR}_{\mathbf{sl}}$. In particular, we compute $\mathcal{J}_{\mathrm{sl}}$ by averaging $|e(n)|^2$ over 1024 symbols and substitute its value into (43) to obtain the value of $\mathrm{SNR}_{\mathbf{sl}}$.

As the NEXT canceller is a complex adaptive filter, we employ the **SR** architecture proposed in [3]. The **SR** architecture enables 21% energy savings without any loss in SNR. However, the energy savings in this paper do not include those due to **SR** transformation. This is because the reference architecture for computing the energy savings is also based on the **SR** architecture. Assume that $\mathrm{SNR}_o = 31$ dB (this is 1.55 dB more than the minimum of 29.45 dB) is the desired performance level. Furthermore, if $\mathrm{SNR}_{\mathbf{sl}} \notin [\mathrm{SNR}_o, \mathrm{SNR}_o + \delta]$, then we assume that the input state has changed substantially so that a new **SPA** configuration for the NEXT canceller needs to be computed. We choose $\delta = 3$ dB to remove undesired glitches in steady state. This will guarantee that the SNR is always better than 29.45 dB, thus keeping the bit error rate below 10^{-10}.

We will assume $T_m = 4$ ns, $T_{\mathrm{mux}} = 0.1$ ns, and $T_{\mathrm{sum}} = 0.7$ ns, $T_{\mathrm{carry}} = 0.6$ ns. The sample period is $T_s = 38$ ns. It is assumed that $V_{\mathrm{dd,max}} = 2.5$ V and the **SMA** block always operates at 2.5 V. The worst-case design corresponds to number of powered up taps $N = 30$, coefficient precision $B_w = 12$ bits, data precision $B_x = 4$ bits, **F**-block adder precision $B_{\mathrm{ADD}} = 16$, and supply voltage $V_{\mathrm{dd,max}} = 2.5$ V. The data precision B_x is kept constant at four bits because the input to the NEXT canceller belongs to the 64-CAP signal set $\{-7, -5, -3, -1, 1, 3, 5, 7\}$, which can be represented with four bits.

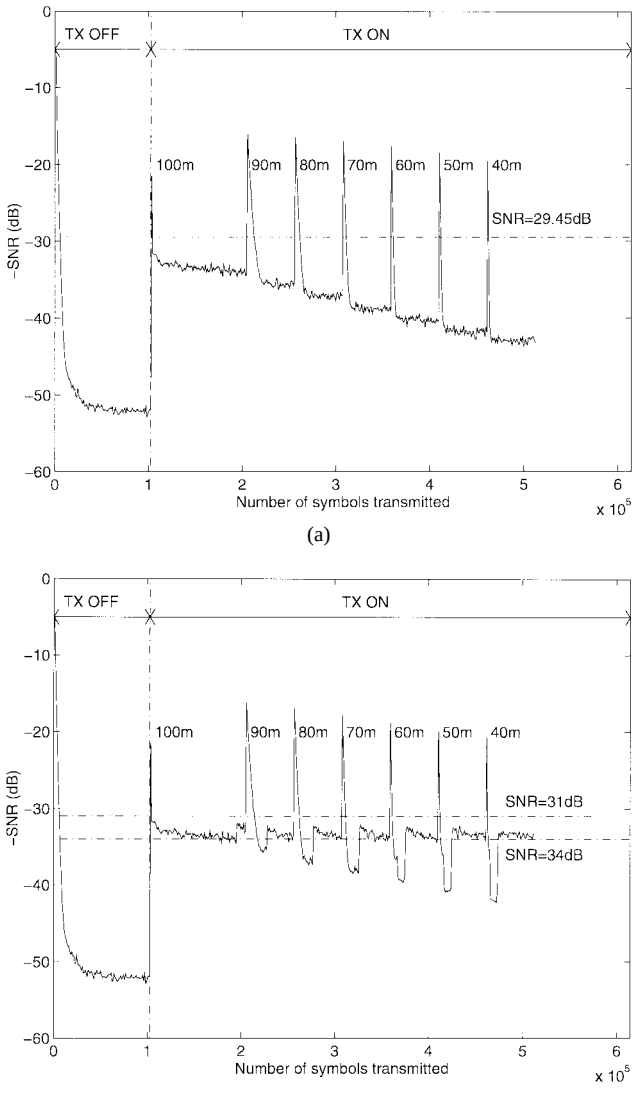


Fig. 10. Convergence curves for NEXT canceller based on: (a) worst case and (b) DAT-based designs.

For the energy consumption, it is assumed that the standard cells based on 0.18-μm 2.5-V CMOS technology are being employed. The energy consumption models for the arithmetic blocks are obtained by real-delay simulations via the gate-level simulation tool MED [35] and employed to compute the energy savings due to DAT. However, the **SMA** block employs the simple energy models (also supported by real-delay simulations), described in Section III-A, in order to compute the energy-optimum configuration, as presented in Section IV-D. We next present the simulation results for the DAT-based NEXT canceller.

C. Simulation Results

Consider the NEXT canceller designed for the worst case. The local transmitter (see Fig. 9) is switched on after 102 400

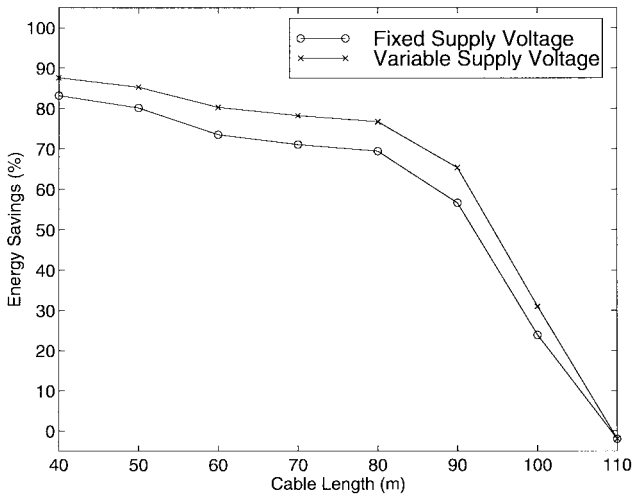


Fig. 11. Energy savings for the DAT-based NEXT canceller.

symbols. This introduces the NEXT interferer into the receiver which the NEXT canceller attempts to cancel. In Fig. 10(a), the convergence plot of $\mathrm{SNR_{sl}}$ shows that the performance of the worst-case design varies from 32 to 42 dB as the length is varied from 100 to 40 m. The value of $\mathrm{SNR_{sl}} = 29.45$ dB ensures a probability of error less than 10^{-10}. Hence, it is possible to trade off the excess performance for shorter cable lengths in order to achieve energy savings. We employ the DAT-based NEXT canceller to enable these energy savings. Recall that for a DAT-based NEXT canceller, an SNR window of 31–34 dB was specified. This means that if $\mathrm{SNR_{sl}}$ is less than 31 dB, then some of the taps are powered up to enhance the performance. Similarly, if $\mathrm{SNR_{sl}}$ is greater than 34 dB, then some of the taps are powered down to achieve energy savings. Thus, DAT-based NEXT canceller jointly optimizes energy dissipation and $\mathrm{SNR_{sl}}$.

In Fig. 10(b), the algorithmic performance measure $\mathrm{SNR_{sl}}$ for a DAT-based NEXT canceller is plotted. It can be seen that $\mathrm{SNR_{sl}}$ for the DAT-based system always lies in the window 31–34 dB during steady-state, which guarantees adequate system performance. Whenever the channel length changes, there is a sudden decrease in $\mathrm{SNR_{sl}}$ [refer to the peaks in Fig. 10(b)]. In that case, all the taps are turned on and the adaptive filter coefficients converge to their optimum settings. After convergence, the **SMA** block monitors the $\mathrm{SNR_{sl}}$ and determines the energy-optimum configuration for the NEXT canceller according to the strategy described in Section IV-D.

The final configuration for each state (shown in Table II) indicates that the number of powered-up taps range from 4 to 30 for cable length variations from 40 to 110 m. Similarly, the coefficient precision vary from 11 to 12 bits. As there is a $4\times$ reduction in the number of powered-up taps in going from state $\mathbf{s}_1$ to $\mathbf{s}_8$, we obtain a reduction in coefficient precision by 1 b. Also, the supply voltage V_{dd} varies from 2.0 to 2.5 V for cable lengths ranging from 40 to 110 m. In Fig. 11, we plot energy savings [see (40))] for a DAT-based NEXT canceller employing array multipliers when the cable length varies from 110 to 40 m. The energy savings include the energy consumption in the **SMA** block. For the variable supply voltage V_{dd} case, the energy savings range from -2% to 89% for cable length variations from 110 to 40 m, respectively. Negative energy savings are due to the reconfiguration overhead that the traditional worst-case designs do not have. The energy savings range from -2% to 87% when the supply voltage V_{dd} is fixed and cable length is varied from 110 to 40 m. It was found that for the state probability distribution $p(\mathbf{s}_i)$ given in Table II, the average energy savings with the fixed supply voltage are 62%. Similarly, for the variable supply voltage case, the average energy savings were found to be 69%. Thus, for this application, only 7% additional energy savings over the fixed supply voltage case are achieved due to the variable supply voltage scheme. This is because the critical path delay varies only marginally with reduction in the number of powered-up taps. Larger energy savings can be expected in the architectures, where the critical path is a strong function of the powered-up taps. Thus, it can be seen that the DAT-based approach is quite attractive from the viewpoint of energy savings for the 155.52-Mb/s ATM–LAN application.

VI. Conclusions and Future Work

We have proposed DAT's as a formal approach to the design of low-power reconfigurable DSP systems and have demonstrated its use in the design of a NEXT canceller for a 155.52-Mb/s ATM–LAN. The main contribution of the DAT-based approach is a systematic determination of energy-optimal reconfiguration strategies for any platform or application. Application of DAT techniques require a proper understanding of the system requirements and the constraints imposed by the reconfigurable hardware fabric. Thus, the DAT approach embodies the growing trend of jointly addressing system and circuit design issues in order to obtain increasingly superior solutions.

DAT's have a broad range of applicability other than the ones presented in this paper. For example, DAT techniques can be applied for developing low-energy software for programmable DSP's, determining energy-optimal reconfiguration strategies for FPGA's, design of low-power wireless transceivers, lattice-based adaptive equalizers, forward error-correction (FEC) codecs, and computer-aided design (CAD) tools that enable the design of complex DAT-based DSP and communication systems.

References

[1] A. Chandrakasan, M. Potkonjak, R. Mehra, J. Rabaey, and R. W. Brodersen, "Minimizing power using transformations," *IEEE Trans. Computer-Aided Design*, vol. 14, pp. 12–31, Jan. 1995.

[2] D. A. Parker and K. K. Parhi, "Low-area/power parallel FIR digital filter implementations," *J. VLSI Signal Processing*, vol. 17, pp. 75–92, Sept. 1997.

[3] N. R. Shanbhag and M. Goel, "Low-power adaptive filter architectures and their application to 51.84 Mb/s ATM–LAN," *IEEE Trans. Signal Processing*, vol. 45, no. 5, pp. 1276–1290, May 1997.

[4] E. Tsern and T. H. Meng, "A low-power videorate pyramid VQ decoder," *IEEE J. Solid-State Circuits*, vol. 31, pp. 1789–1794, Nov. 1996.

[5] H. H. Loomis and B. Sinha, "High speed recursive digital filter realization," *Circuit, Syst., Signal Processing*, vol. 3, no. 3, pp. 267–294, 1984.

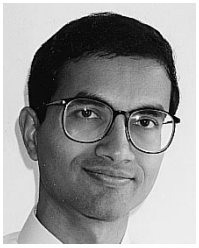
[6] K. K. Parhi and D. G. Messerschmitt, "Pipeline interleaving and parallelism in recursive digital filters—Part I & II," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1099–1134, July 1989.

[7] A. Shen, A. Ghosh, S. Devdas, and K. Keutzer, "On average power dissipation and random pattern testability of CMOS combinational logic networks," in *IEEE Int. Conf. Computer-Aided Design*, 1992, pp. 402–407.
[8] S. Iman and M. Pedram, "An approach for multilevel logic optimization targeting low-power," *IEEE Trans. Computer-Aided Design*, vol. 15, pp. 889–901, Aug. 1996.
[9] M. Alidina, J. Monterio, S. Devdas, A. Ghosh, and M. Papaefthymiou, "Precomputation-based sequence logic optimization for low-power," *IEEE Trans. VLSI Syst.*, vol. 2, pp. 398–407, Dec. 1994.
[10] Y. Nakagome *et al.*, "Sub-1 V swing internal bus architecture for future low-power ULSI's," *IEEE J. Solid-State Circuits*, vol. 28, pp. 414–419, Apr. 1993.
[11] W. C. Athas, L. J. Svensson, J. G. Koller, N. Tzartzanis, and E. Y.-C. Chou, "Low-power digital systems based on adiabatic switching principles," *IEEE Trans. VLSI Syst.*, vol. 2, pp. 398–407, Dec. 1994.
[12] B. Davari, R. H. Dennard, and G. G. Shahidi, "CMOS scaling for high-performance and low-power—The next ten years," *Proc. IEEE*, vol. 83, pp. 595–606, Apr. 1995.
[13] K. K. Parhi, "Algorithm transformation techniques for concurrent processors," *Proc. IEEE*, vol. 77, pp. 1879–1895, Dec. 1989.
[14] N. R. Shanbhag and K. K. Parhi, *Pipelined Adaptive Digital Filters*. Norwell, MA: Kluwer, 1994.
[15] M. Potkonjak and J. Rabaey, "Fast implementation of recursive programs using transformations," in *Proc. ICASSP*, San Francisco, CA, Mar. 1992, pp. 569–572.
[16] C. Leiserson and J. Saxe, "Optimizing synchronous systems," *J. VLSI Comput. Syst.*, vol. 1, pp. 41–67, 1983.
[17] A. Abnous and J. M. Rabaey, "Ultra-low-power domain-specific multimedia processors," in *Proc. IEEE VLSI Signal Processing Workshop*, Oct. 1996, pp. 461–470.
[18] J. T. Ludwig, S. H. Nawab, and A. P. Chandrakasan, "Low-power digital filtering using approximate processing," *IEEE J. Solid-State Circuits*, vol. 31, pp. 395–400, Mar. 1996.
[19] V. Gutnik and A. P. Chandrakasan, "Embedded power supply for low-power DSP," *IEEE Trans. VLSI Syst.*, vol. 12, pp. 425–435, Dec. 1997.
[20] D. Keppel, "A portable interface for on-the-fly instruction space modification," in *Proc. 4th Int. Conf. ASPLOS-IV*, Apr. 1991, pp. 86–95.
[21] C. Consel and F. Noël, "A general approach for run-time specialization and application to *C*," in *Proc. 23rd ACM SIGPLAN-SIGACT Symp. POPL*, Jan. 1996, pp. 145–156.
[22] J. Villasenor and B. Hutchings, "The flexibility of configurable computing," *IEEE Signal Processing Mag.*, pp. 67–84, Sept. 1998.
[23] S. Trimberger, D. Carberry, A. Johnson, and J. Wong, "A time-multiplexed FPGA," in *Proc. IEEE Workshop FPGA's Custom Comput. Machines*, J. Arnold and K. L. Pocek, Eds., Napa, CA, Apr. 1997, pp. 22–28.
[24] N. Shirazi, P. M. Athanas, and A. L. Abbott, "Implementation of a 2-D fast Fourier transform on an FPGA—Based custom computing machine," in *Field-Programmable Logic Applicat. 5th Int. Workshop Field-Programmable Logic Applicat.*, W. Moore and W. Luk, Eds., Oxford, U.K., Sept. 1995, pp. 282–292.
[25] J. R. Hauser and J. Wawrzynek, "GARP: A MIPS processor with a reconfigurable coprocessor," in *Proc. IEEE Workshop FPGA's Custom Comput. Machines*, J. Arnold and K. L. Pocek, Eds., Napa, CA, Apr. 1997, pp. 12–21.
[26] C. Rupp, M. Landguth, T. Garverick, E. Gomersall, H. Holt, J. Arnold, and M. Gokhale, "The NAPA adaptive processing architecture," in *IEEE Symp. Field-Programmable Custom Comput. Machines*, Napa, CA, Apr. 1998, pp. 23–37.
[27] M. Goel and N. R. Shanbhag, "Low-power equalizers for 51.84 Mb/s very high-speed digital subscriber loop (VDSL) modems," in *IEEE Workshop Signal Processing Syst.: Design and Implementation*, Boston, MA, Oct. 1998, pp. 317–326.
[28] ———, "Dynamic algorithm transformations (DAT) for low-power adaptive signal processing," in *Int. Symp. Low-Power Electron. and Design*, Monterey, CA, Aug. 1997, pp. 161–166.
[29] S. Haykin, *Adaptive Filter Theory*. Englewood Cliff, NJ: Prentice-Hall, 1991.
[30] B. Widrow *et al.*, "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proc. IEEE*, vol. 64, pp. 1151–1162, Aug. 1976.
[31] V. Gutnik, "Variable supply voltage for low power DSP," Master's thesis, Dept. Elect. Eng. Comput. Sci., Massachusetts Inst. Technol., Cambridge, May 1996.
[32] R. E. Gonzalez, "Low-power processor design," Ph.D. dissertation, Dept. Elect. Eng. Comput. Sci., Stanford Univ., Stanford, CA, June 1997.
[33] N. Weste and K. Eshraghian, *Principles of CMOS VLSI Design: A Systems Perspective*. Reading, MA: Addison-Wesley, 1988.
[34] N. Sankarayya, K. Roy, and D. Bhattacharya, "Algorithms for low power and high speed fir filter realization using differential coefficients," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 488–497, June 1997.
[35] M. G. Xakellis and F. N. Najm, "Statistical estimation of the switching activity in digital circuits," in *Design Automation Conf.*, June 1994, pp. 728–733.
[36] C. J. Nicol, P. Larsson, K. Azadet, and J. H. O'Neill, "A low power 128-tap digital adaptive equalizer for broadband modems," *IEEE J. Solid-State Circuits*, vol. 32, pp. 1777–1789, Nov. 1997.
[37] D. P. Bertsekas, *Nonlinear Programming*. Boston, MA: Athena Scientific, 1995.
[38] M. Goel and N. R. Shanbhag, "Dynamic algorithm transforms for low-power reconfigurable adaptive equalizers," *IEEE Trans. Signal Processing*, to be published.
[39] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1994.
[40] G. H. Im and J. J. Werner, "Bandwidth-efficient digital transmission up to 155-Mb/s over unshielded twisted-pair wiring," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 1643–1655, Dec. 1995.
[41] *155.52 Mb/s Physical Layer Specifications for Category 3 Unshielded Twisted Pair*, ATM Forum Tech. Committee af-phy-0047.00, Nov. 1995.
[42] *Commercial Building Telecommunications Cabling Standard*, Standard TIA/EIA-568-A, 1994.

Manish Goel received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, New Delhi, India, in 1994, the M.S. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, in 1997, and is currently working toward the Ph.D. degree in electrical and computer engineering at University of Illinois at Urbana-Champaign.

His research interests include low-power DSP system design for high bit-rate wireless and wireline communications.

Mr. Goel received the 1993 Motorola Undergraduate Project Award and the 1994 Best Undergraduate Project Award presented by the Indian Institute of Technology.



Naresh R. Shanbhag (S'87–M'88) received the B.Tech. degree from the Indian Institute of Technology, New Delhi, India, in 1988, the M.S. degree from Wright State University in 1990, and the Ph.D. degree from the University of Minnesota at Minneapolis–St. Paul, in 1993, all in electrical engineering.

From July 1993 to August 1995, he was with AT&T Bell Laboratories, Murray Hill, NJ, in the Wide-Area Networks Group, where he was responsible for development of VLSI algorithms, architectures and implementation for high-speed data communications applications. In particular, he was the Lead Chip Architect for AT&T's 51.84-Mb/s transceiver chips over twisted-pair wiring for ATM–LAN and broad-band access applications. Since August 1995, he has been a Research Assistant Professor with the Coordinated Science Laboratory, VLSI Circuits Group, and an Assistant Professor in the Department of Electrical and Computer Engineering at the University of Illinois at Urbana-Champaign. His research interests include exploring the limits of computation in an integrated-circuit media in terms of power dissipation, reliability, and throughput, and developing VLSI algorithms, architectures, and integrated circuits for signal processing and communication systems. He has published over 20 journal articles/book chapters and over 30 conference publications on these subjects.

Dr. Shanbhag received the 1999 Xerox Faculty Award, the 1999 IEEE Leon K. Kirchmayer Best Paper Award, the 1996 NSF Career Award, and the 1994 Darlington Best Paper Award presented by the IEEE Circuits and Systems Society. He has served as an associate editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART II: ANALOG AND DIGITAL SIGNAL PROCESSING since July 1997. He was a Distinguished Lecturer of the IEEE Circuits and Systems Society from 1997 to 1999.