Dynamic Algorithm Transforms for Low-Power Reconfigurable Adaptive Equalizers

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

***Abstract*—In this paper, we present low-power reconfigurable adaptive equalizers derived via *dynamic algorithm transforms* (DAT's). The principle behind DAT is that conventional signal processing systems are designed for the worst case and are not energy-optimum on average. Therefore, significant energy savings can be achieved by optimally reconfiguring the hardware in these situations. Practical reconfiguration strategies for adaptive filters are presented. These strategies are derived as a solution to an optimization problem. The optimization problem has energy as the objective function and a constraint on the algorithm performance (specifically the SNR). The DAT-based adaptive filter is employed as an equalizer for a 51.84 Mb/s very-high-speed digital subscriber loop (VDSL) over 24-pair BKMA cable. The channel nonstationarities are due to variations in cable length and number of far-end crosstalk (FEXT) interferers. For this application, the traditional design is based on 1 kft cable length and 11 FEXT interferers. It was found that up to 81% energy savings can be achieved when cable length varies from 1–0.1 kft and the number of FEXT interferers varies from 11 to 4. On the average, 53% energy savings are achieved as compared with the conventional worst-case design.**

***Index Terms*— Algorithm transforms, equalizers, low power, reconfigurable, subscriber loops.**

I. Introduction

THE POWER dissipation of CMOS circuits [1], [2] is a grave concern in the VLSI industry. This concern is mainly driven by the limited battery life in mobile applications, reliability, as well as packaging costs in both mobile and tethered applications. Several low-power techniques [1] have been proposed for general VLSI as well DSP-specific systems. General low-power techniques include logic minimization [3]–[4] and precomputation [5] (at the logic level), reduced voltage swing [6] and adiabatic logic [7] (at the circuit level), and CMOS scaling [8] (at the technological level). The low-power techniques specific to DSP systems include strength reduction [9]–[11] and DECOR [12] (at the algorithmic level) and pipelining [13]–[14] and parallel processing [14] (at the architecture level). Moreover, *algorithm transformation techniques* [15] such as *look-ahead* [14], *relaxed look-ahead* [16], *algebraic transformations* [17], and *retiming* [18] have been employed in high-speed and more recently low-power DSP system design. All of the above-mentioned techniques are applied during the VLSI design phase and their implementation is time invariant. Therefore, we refer to this class of low-power methods as *static techniques*.

Recently, *dynamic techniques* both at the circuit level and algorithmic level have been proposed. Dynamic techniques can be applied after static techniques to obtain even greater energy savings. These techniques are based on 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. Such systems are referred to as *reconfigurable signal processing* systems [19]–[29]. In [19], reconfigurability is employed to map a wide class of signal processing algorithms to an appropriate architectural template. Field programmable gate array (FPGA)-based devices and their reconfiguration schemes are discussed in [20]–[22]. Hybrid architectures based on FPGA's and general-purpose DSP's is the topic of research in [23] and [24]. Other dynamic techniques include approximate signal processing [25], [26], where just the right amount of computational resources needed at a specific instant/period to meet the algorithm performance requirements is allocated. In addition to dynamic techniques at circuit and architecture level, techniques at the algorithmic level [27]–[29] are also being developed. The key goal of these techniques is to improve the algorithm performance (such as convergence rate [27], data-rate [28], and image distortion [29]) by exploiting variabilities in the data and channel. Thus, there are several emerging dynamic techniques at the circuit, architecture, and algorithmic levels.

Our approach to the design of reconfigurable DSP systems is to add just the right degree of flexibility (as demanded by the application) to ASIC's, resulting in application-specific reconfigurable IC's (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. Recently, we have proposed dynamic algorithm transform (DAT) [30], [31] (see Fig. 1) for the design of low-power ASRIC's. From an implementation perspective, a DAT-based reconfigurable DSP system has the signal processing algorithm (**SPA**) implemented in a reconfigurable hardware (see Fig. 1) (such as FPGA, certain DSP's, or ASIC's), whereas the input state and state transitions are monitored by a signal monitoring algorithm (**SMA**) block (or a controller). In [30], DAT techniques are studied in the context of a system-identification scenario with application to a near-end crosstalk (NEXT) canceller for 155.52 Mb/s ATM-LAN. A general framework for DAT is presented in [31], where the variabilities in the input are modeled as transitions in an *input state-space*, and the reconfigurations

Manuscript received May 21, 1998; revised January 5, 1999. This work was supported by DARPA under Contract DABT63-97-C-0025 and National Science Foundation CAREER Award MIP-9623737. The associate editor coordinating the review of this paper and approving it for publication was Dr. Konstantinos Konstantinides.

The authors are with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA.

Publisher Item Identifier S 1053-587X(99)07661-8.

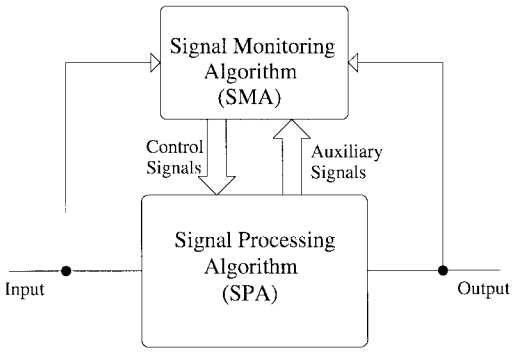


Fig. 1. DAT-based reconfigurable signal processing system.

are modeled as transitions in the *configuration space* of the reconfigurable hardware fabric. In [30] and [31], DAT was employed to derive energy-optimal reconfiguration strategies for adaptive equalizers. These strategies were employed in the context of a 155.52 ATM-LAN transceiver, and significant energy savings were demonstrated. In this paper, we provide proofs of energy optimality of the reconfiguration strategies in [31] and study their application in a 51.84 Mb/s very high-speed digital subscriber loop (VDSL) [32], [34].

Preliminaries results of this work have appeared in [35], where we extended the strategy in [30] to include the equalization scenario along with precise energy models. Related works include a nonuniformly spaced equalizer [36], where a technique for choosing the best K taps (in terms of MSE) out of total N taps is presented. This technique, however, is very complex and is not suitable for the real-time implementation. In contrast, our reconfiguration strategy is much simpler and is employed for real-time reconfigurations. In [37], a 128-tap adaptive equalizer was proposed in which the tap length and precision are varied to maintain a certain SNR. Our approach is systematic as we employ the *Lagrange multiplier method* [38] to find optimum set of powered-up taps, which are not necessarily the end taps. Further, we consider a fractionally spaced linear equalizer (FSLE) and a complex-valued strength-reduced [38] feedback equalizer.

The rest of the paper is organized as follows. In Section II, we present some preliminaries related to the adaptive filter and multiplier energy models. The dynamic algorithm transforms are discussed in Section III, and energy-optimum reconfiguration strategy for adaptive filters is presented in Section IV. Finally, in Section V, we employ the DAT-based equalizer in 51.84 Mb/s VDSL transceiver and present simulation results.

II. Preliminaries

In this section, we present some preliminaries regarding the reconfigurable architecture of the adaptive filter and energy models for the multipliers. Later sections will employ these energy models to determine the energy-optimum configuration for the adaptive filter architecture.

A. Reconfigurable Adaptive Filter

Let $x(n)$ be the input signal to an N-tap adaptive filter, and let $w_k(n), k = 1,2,\cdots,N$ be the real-valued filter coefficients. The least mean square (LMS) algorithm [39] is then given by

$$e(n) = d(n) - \sum_{k=1}^{N} w_k(n-1)x(n-k+1) \quad (2.1)$$

$$w_k(n) = w_k(n-1) + \mu e(n)x(n-k+1) \quad k = 1,2,\cdots,N \quad (2.2)$$

where $e(n)$ and $d(n)$ are the output error and the desired output, respectively, and μ is the step size. If the correlation sequence $r_x(k) = E[x(n)x(n-k)]$ of the input signal is known, then the mean squared error (MSE) $\mathcal{J} = E[e^2(n)]$ is computed as [40]

$$\mathcal{J} = \sigma_d^2 - \sum_{k=1}^{N}\sum_{j=1}^{N} w_k w_j r_x(k-j) \quad (2.3)$$

where $\sigma_d^2 = E[d^2(n)]$ is the desired signal power, and w_k are the optimum filter coefficients.

A direct implementation of the LMS algorithm is shown in Fig. 2(a), where each tap consists of two multipliers and two adders: the filter (**F**) block implements (2.1) and the weight-update (**WUD**) block implements (2.2). The architecture in Fig. 2(a) can be modified to obtain a reconfigurable architecture in Fig. 2(b), where we have introduced additional control signals α_k and β_k, which are employed to 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, thereby powering down the kth tap in the **F**-block. Similarly, $\beta_k = 0$ powers down the kth tap in the **WUD**-block.

Additional energy savings can be achieved by adapting the precisions of the input signal and the coefficients. The input precision B_x is chosen to achieve a specified signal-to-quantization noise ratio (SQNR). It can be shown that SQNR at the input is given by

$$\text{SQNR(db)} = 6B_x + 4.8 - \text{PAR(db)} \quad (2.4)$$

where PAR is the peak-to-average ratio at the input and is computed by dividing the maximum value of the input signal with its root-mean squared (RMS) value. It can be seen from (2.4) that a 6-dB reduction in PAR results in a 1-bit reduction in B_x.

It can be shown [41] that to achieve a specified SQNR at the output, the coefficient precision B_w is given by

$$B_w = B_{w,\max} + \frac{1}{2}\log_2\left(\frac{1}{N}\sum_{k=1}^{N}\alpha_k\right) \quad (2.5)$$

where $B_{w,\max}$ is the maximum value of B_w (for $\alpha_k = 1$, $k = 1,2,\cdots N$). Thus, (2.5) indicates that a 1-bit reduction in the precision is achieved for each four-fold reduction in filter length. In Section IV, we will present reconfiguration strategies to choose the parameters α_k and β_k in an energy-optimum manner. In the next subsection, we present a multiplier energy model that will be employed to derive the reconfiguration strategies presented in Section IV.

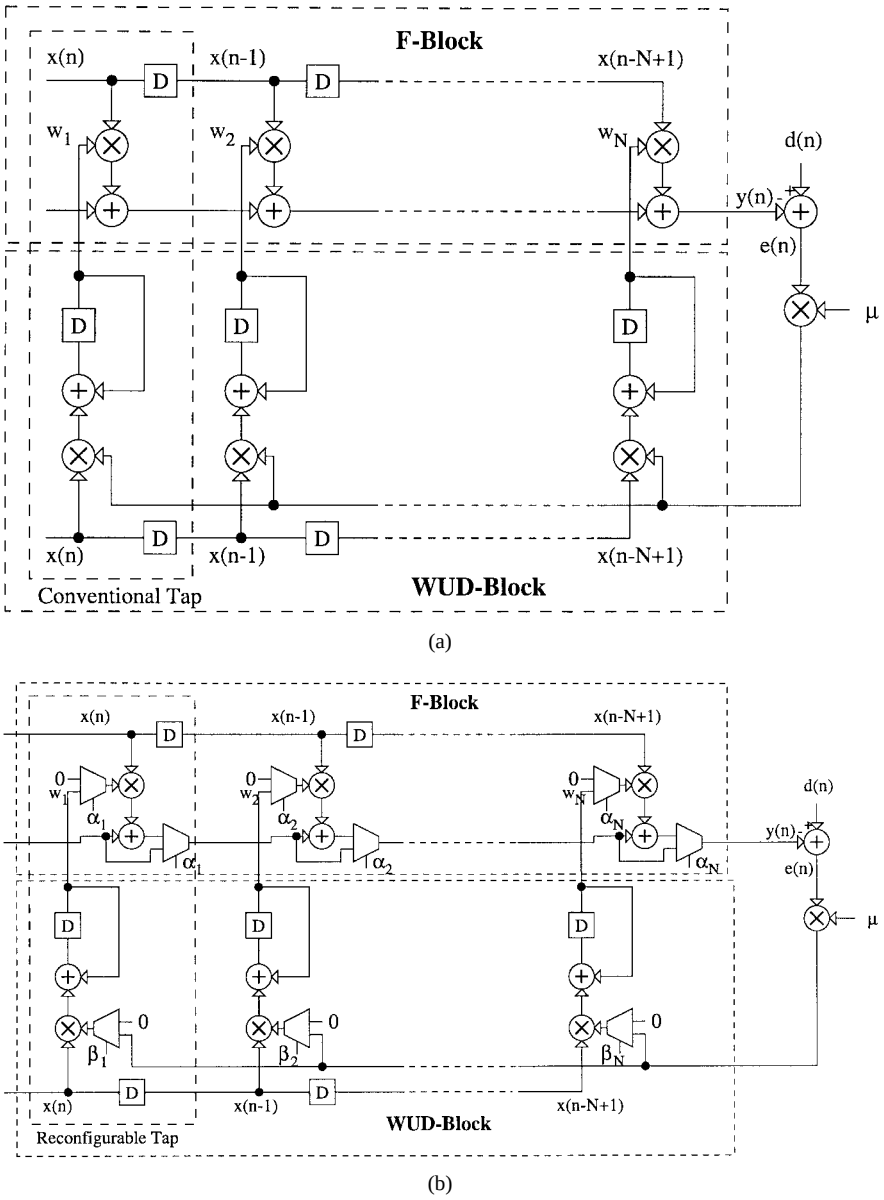


Fig. 2. LMS adaptive filter. (a) Conventional architecture. (b) Reconfigurable architecture.

B. Multiplier Energy Model

We focus on energy models for the multipliers as these consume a large percentage of the total energy. There is a significant on-going effort [42], [43] in the computer-aided design (CAD) community to find accurate estimates of the energy dissipation. Our interest here is to determine accurate and simple relative power dissipation models that can be employed by the **SMA** block to determine an energy-optimum configuration in real time. 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. We will assume that a B_x-bit signal $x(n)$ is being multiplied by a B_w-bit constant coefficient w_k. The constant coefficient assumption is valid for adaptive filters if we assume that the **WUD** block is powered down after convergence. Assuming a two's-complement representation, we have

$$w_k = -w_k^{(0)} 2^{B_w-1} + \sum_{j=1}^{B_w-1} w_k^{(j)} 2^{B_w-1-j} \tag{2.6}$$

where $w_k^{(j)}$ is the value of the jth bit of coefficient w_k. In [35], we define $\mathcal{N}_1(w_k)$ as the number of nonzero bits in the coefficient w_k given in (2.6). Similarly, $\mathcal{N}_2(w_k)$ is defined as the difference of B_w and the number of zeros at least significant bit (LSB) positions and is given as

$$\mathcal{N}_2(w_k) = B_w - \sum_{j=0}^{B_w-1} \prod_{i=j}^{B_w-1} (1 - w_k^{(i)}). \tag{2.7}$$

It was found via a real-delay gate-level simulations [44] that the linear energy model based on $\mathcal{N}_1(w_k)$ and $\mathcal{N}_2(w_k)$ underestimates and overestimates the energy consumption of the multiplier, respectively. Therefore, a better energy model can be obtained by taking a weighted sum of $\mathcal{N}_1(w_k)$ and $\mathcal{N}_2(w_k)$ as

$$\mathcal{E}_m(w_k) = \mathcal{E}_{\max}\frac{\eta\mathcal{N}_1(w_k) + (1-\eta)\mathcal{N}_2(w_k)}{B_w}. \tag{2.8}$$

The constant η was chosen to equal 0.9 in order to minimize the error between $\mathcal{E}_m(w_k)$ and the real-delay energy values obtained via a gate-level simulation tool MED [44]. Standard cells based on 0.18 μm, 2.5 V technology are assumed for the real-delay simulations. It was found that the model in (2.8) is accurate with less than 9% estimation error, as compared with a real-delay gate-level simulation. Note that models based on closed-form expressions such as (2.8) are useful in computing the energy-optimum configurations.

III. Dynamic Algorithm Transforms (DAT)

In this section, we present the general framework for DAT's. The motivation for DAT is that the conventional signal processing system designed for the worst case is usually not optimum (from energy perspective) for the best and the nominal cases. Hence, significant energy efficiencies can be gained by having a signal monitoring algorithm or the **SMA** block (see Fig. 1) that monitors the input state and then reconfigures the **SPA** block to match the input. This naturally leads to the definition of input state and configuration, which are presented in Sections III-A and III-B, respectively. In Section III-C, we show how energy savings can be calculated.

A. Input State Space

We employ the *input state space* to distinguish between different scenarios at the input. In general, the input state space depends on the input nonstationarity and the hardware granularity. For example, the input state- space needs to have more states for a hardware platform with fine granularity of reconfiguration as compared with the one with coarse granularity. The *input state* is formally defined as follows.

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

For example, assume that a 51.84 Mb/s VDSL network has 100 connections, out of which 80% of the are approximately 0.6 kft from the transmitter, and the remaining are distributed equally between 0.1–1 kft. Assume further that the equalizer complexities of these three cable lengths are substantially different in order to warrant reconfiguration. In that case, we can define the input state space $\mathcal{S} = \{\boldsymbol{s}_1, \boldsymbol{s}_2, \boldsymbol{s}_3\}$, $p(\boldsymbol{s}_1) = 0.1$, $p(\boldsymbol{s}_2) = 0.8$, and $p(\boldsymbol{s}_3) = 0.1$, where $\boldsymbol{s}_1$, $\boldsymbol{s}_2$, and $\boldsymbol{s}_3$ are the input states corresponding to the cable length of 0.1, 0.6, and 1 kft, respectively. For this example, $\boldsymbol{s}_3$ corresponds to the worst case, whereas $\boldsymbol{s}_1$ and $\boldsymbol{s}_2$ represent the best and the nominal cases. Proceeding further, we can define each state as $\boldsymbol{s}_i = [\sigma_{x_i}^2, \text{PAR}_i, \text{SNR}_{\text{in}_i}]$, where $\sigma_{x_i}^2$, PAR_i, and SNR_{in_i} are input signal energy, input peak-to-average ratio (PAR), and input signal-to-noise ratio, respectively. Thus, we an input state-space with three states $(\boldsymbol{s}_1, \boldsymbol{s}_2, \boldsymbol{s}_3)$ will suffice where each state is a three-element vector.

B. Configuration Space

A reconfigurable hardware fabric is characterized by its *configuration vector* as defined below.

Definition 2: The **configuration vector** $\boldsymbol{c}(n) \in \mathcal{C} = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_{N_c}\}$ (where $\mathcal{C}$ is the **configuration-space**) at time instant n is defined as a vector of reconfiguration control signals. Each configuration vector $\boldsymbol{c}_i$ corresponds to a particular value of the control signals.

For example, for the N-tap reconfigurable adaptive filter in Fig. 2(b), the configuration vector is defined as

$$\boldsymbol{c}(n) = [\alpha_1(n), \cdots, \alpha_N(n), \beta_1(n), \cdots, \beta_N(n)]$$

where $\alpha_i(n)$ and $\beta_i(n)$ are 1-bit control signals indicating whether the filtering and weight-update portions of the ith tap are powered up or not. Thus, for $N = 48$, $\boldsymbol{c}(n)$ is a 96-bit control signal. Therefore, the configuration space $\mathcal{C}$ corresponds to $N_c = 2^{96}$ binary tuples of dimension 96. The above example indicates how easily the number of configurations explodes with an increase in reconfigurability options. We are interested in determining the energy-optimum configuration $\boldsymbol{c}^*(\boldsymbol{s}_i)$ for every input state $\boldsymbol{s}_i$ from a total of N_c possible configurations while satisfying a mean squared error (MSE) constraint. Due to the large number of possible configurations N_c, it becomes important to develop a systematic approach to determining the energy-optimum configuration $\boldsymbol{c}^*(\boldsymbol{s}_i)$. For every state $s_i \in \mathcal{S}$, there exists an energy-optimum configuration $\boldsymbol{c}^*(\boldsymbol{s}_i)$ defined as follows.

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

$$\begin{aligned} \boldsymbol{c}^*(s_i) &= \arg\min_{\boldsymbol{c}\in\mathcal{C}} \mathcal{E}_{\text{SPA}}(\boldsymbol{c}) \\ &\text{s.t. } \mathcal{J}_{\text{SPA}}(\boldsymbol{s}_i, \boldsymbol{c}) \le \mathcal{J}_o \end{aligned} \tag{3.1}$$

where

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

The optimum **SPA** configuration is illustrated in Fig. 3. For each state $\boldsymbol{s}_i \in \mathcal{S}$, we need to compute the optimum configuration $\boldsymbol{c}_j \in \mathcal{C}$ such that the energy dissipation is minimized while satisfying the constraint on the algorithm performance.

C. Energy Savings

The average energy savings $(\mathcal{E}_{\text{sav,ave}})$ of a DAT-based system as compared with the worst-case design is given as

$$\mathcal{E}_{\text{sav,ave}} = \frac{\mathcal{E}_{\text{WC,ave}} - \mathcal{E}_{\text{DAT,ave}}}{\mathcal{E}_{\text{WC,ave}}} \times 100\% \tag{3.2}$$

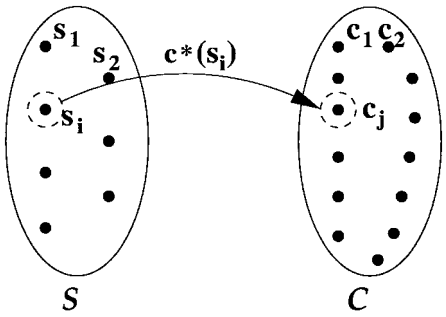


Fig. 3. Reconfiguration as a mapping from the input state-space to the configuration-space.

where $\mathcal{E}_{\mathrm{DAT,ave}}$ and $\mathcal{E}_{\mathrm{WC,ave}}$ are the average energy dissipation of the DAT-based system and worst-case design, respectively. Large energy savings can be expected for situations where $\mathcal{E}_{\mathrm{WC,ave}} \gg \mathcal{E}_{\mathrm{DAT,ave}}$. Such situations arise if the energy dissipation requirements for the worst and the nominal cases are considerably different and the probability of occurrence of worst-case input is sufficiently small. Note that $\mathcal{E}_{\mathrm{DAT,ave}}$ includes the energy of the **SPA** datapath $\mathcal{E}_{\mathrm{SPA,ave}}$ and that of the **SMA** controller $\mathcal{E}_{\mathrm{SMA,ave}}$. The **SPA** energy consumption $\mathcal{E}_{\mathrm{SPA,ave}}$ can be computed by averaging $\mathcal{E}_{\mathrm{SPA}}(\boldsymbol{c}^*(\boldsymbol{s}_i))$ over all the states $\boldsymbol{s}_i \in \mathcal{S}$. Most of the **SMA** block is activated after L samples only if there is a transition in the state. Therefore, the energy dissipation of the **SMA** block will be negligible for L sufficiently large. However, we do include a fixed constant value for the **SMA** energy consumption to reflect the fact that the state monitor in the **SMA** is always active. This was found to be 2% of the worst-case energy consumption for the VDSL application. In the next section, we derive energy-optimum reconfiguration strategies for adaptive filters as a solution to (3.1).

IV. Reconfiguration Strategy for Adaptive Filters

In this section, we employ the *Lagrange multiplier method* to derive an energy-optimum reconfiguration strategy for the adaptive filter architecture in Fig. 2(b). The reconfigurable parameters in this architecture are the control signals α_is and β_is, which indicate the powered-up taps, and precisions B_w and B_x. The choice of precisions B_w and B_x were presented in Section II-A. Once the optimum value of α_is are obtained, then the coefficient precision B_w can be computed via (2.5), and the input precision B_x can be computed from (2.4). In this section, we present strategies for determining energy-optimal values of α_is and β_is. In Section IV-A, we formulate the energy optimization problem and derive the reconfiguration strategy in Section IV-B.

A. Lagrange Formulation

The optimization problem in (3.1) can be rewritten as

$$\min_{\boldsymbol{c}\in\mathcal{C}} \mathcal{E}_{\mathrm{SPA}}(\boldsymbol{c}) \quad \text{s.t. } \mathcal{J}_{\mathrm{SPA}}(\boldsymbol{c}) - \mathcal{J}_o \le 0 \tag{4.1}$$

where we have dropped the state $\boldsymbol{s}_i$ to simplify notation and with the understanding that we will now compute the optimum configuration for a given state. We refer to the optimization problem in (4.1) as a *primal problem*. The Lagrange multiplier method [38] requires the definition of the *Lagrangian function* $\mathcal{L}(\boldsymbol{c},\lambda)$, which is defined as

$$\mathcal{L}(\boldsymbol{c},\lambda) = \mathcal{E}_{\mathrm{SPA}}(\boldsymbol{c}) + \lambda(\mathcal{J}_{\mathrm{SPA}}(\boldsymbol{c}) - \mathcal{J}_o) \tag{4.2}$$

where $\lambda \ge 0$ is a real-valued Lagrange multiplier. Now, the problem is to find the pair $(\boldsymbol{c}^*,\lambda^*)$ that optimizes $\mathcal{L}(\boldsymbol{c},\lambda)$. This can be done by defining the *saddle point* [38] as follows.

Definition 4: A **saddle-point** $(\boldsymbol{c}^*,\lambda^*)$ of Lagrangian function $\mathcal{L}(\boldsymbol{c},\lambda)$ is defined as one that satisfies

$$\mathcal{L}(\boldsymbol{c}^*,\lambda) \le \mathcal{L}(\boldsymbol{c}^*,\lambda^*) \le \mathcal{L}(\boldsymbol{c},\lambda^*) \tag{4.3}$$

for all $(\boldsymbol{c}^*,\lambda)$ and $(\boldsymbol{c},\lambda^*)$ sufficiently close to $(\boldsymbol{c}^*,\lambda^*)$.

The above definition implies that a saddle point is a local minimum of $\mathcal{L}(\boldsymbol{c},\lambda)$ in the $\boldsymbol{c}$ space and a local maximum in λ space. Thus, a natural way for finding the saddle point is to descend in the $\boldsymbol{c}$ space and ascend in the λ space. The Lagrange multiplier λ can also be viewed as the penalties associated with constraints. Therefore, ascents of $\mathcal{L}$ in λ-space corresponds to increasing the penalties for the unsatisfied constraints. Similarly, a descent in the $\boldsymbol{c}$-space corresponds to the minimization of the objective function while satisfying the constraints. Based on this understanding, we can redefine the optimization problem in (4.1) as

$$\max_{\lambda\ge 0} \min_{\substack{\boldsymbol{c}\in\mathcal{C} \\ \mathcal{J}_{\mathrm{SPA}}(\boldsymbol{c})\le\mathcal{J}_o}} \mathcal{L}(\boldsymbol{c},\lambda). \tag{4.4}$$

It can be shown [38] that $\boldsymbol{c}^*$ in the saddle point $(\boldsymbol{c}^*,\lambda^*)$ of (4.2) obtained as the solution to (4.4) is also the optimum solution for the primal problem in (4.1). This is also called the *saddle point theorem*; refer to [38] for a detailed proof.

B. Energy-Optimum Reconfiguration Strategy for Adaptive Filters

We solve (4.4) for adaptive filters under the following assumptions:

1) The **WUD** block in Fig. 2(b) is switched off (i.e., $\beta_i = 0, i = 1,2,\cdots,N$) after the filter has converged.
2) The input $x(n)$ is uncorrelated. In other words, we will assume that the correlation sequence $r_x(k-j) = E[x(n-k)x(n-j)]$ of the input signal $x(n)$ is nonzero (and equal to σ_x^2) only if $k = j$.

Under these two assumptions, the energy dissipation of the adaptive filter in Fig. 2(b) is given by

$$\mathcal{E}_{\mathrm{SPA}}(\underline{\alpha}) = \sum_{k=1}^{N} \alpha_k \mathcal{E}_m(w_k) \tag{4.5}$$

where $\mathcal{E}_m(w_k)$ is the energy dissipated by a multiplier with coefficient w_k, and $\underline{\alpha} = [\alpha_1,\cdots,\alpha_N]$ is a vector representation of α_is. Note that we have ignored the energy dissipation of the adder in each tap. This is a reasonable assumption since multipliers are the power-hungry blocks in digital filters. We have also employed assumption 1 so that the energy consumption of the **WUD** block can be ignored. Substituting

$\alpha_k w_k$ for w_k in (2.3) and employing assumption 2 above, we obtain

$$\mathcal{J}_{\mathrm{SPA}}(\underline{\alpha}) = \sigma_d^2 - \sum_{k=1}^{N} \alpha_k w_k^2 \sigma_x^2 \tag{4.6}$$

where σ_d^2, $r_x(k-j), w_k$, and α_k are defined in Section II-A. The Lagrangian function $\mathcal{L}(\underline{\alpha}, \lambda)$ can now be derived by employing (4.2), (4.5), and (4.6) as

$$\mathcal{L}(\underline{\alpha}, \lambda) = \sum_{k=1}^{N} \alpha_k [\mathcal{E}_m(w_k) - \lambda |w_k|^2 \sigma_x^2] + \lambda(\sigma_d^2 - \mathcal{J}_o). \tag{4.7}$$

Employing the *saddle-point theorem* [see (4.3)], we define the Lagrangian optimization problem as

$$\max_{\lambda \geq 0} \min_{\substack{\underline{\alpha} \in \{0,1\}^N \\ \mathcal{J}(\underline{\alpha}) \leq \mathcal{J}_o}} \left[\sum_{k=1}^{N} \alpha_k [\mathcal{E}_m(w_k) - \lambda |w_k|^2 \sigma_x^2] + \lambda(\sigma_d^2 - \mathcal{J}_o) \right]. \tag{4.8}$$

We show in Appendix A that the solution to (4.8) is given by

$$\alpha_{k,\mathrm{opt}} = \begin{cases} 1, & \text{if } \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} > \tilde{\lambda} \\ 0, & \text{if } \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} \leq \tilde{\lambda} \end{cases} \tag{4.9}$$

where $\tilde{\lambda} = 1/(\sigma_x^2 \lambda^*)$ is a constant, and λ^* is the optimum value of λ. In practice, we do not need to compute the constant λ^* if we employ the reconfiguration strategy of powering down the taps, starting with the smallest value of $|w_k|^2/\mathcal{E}_m(w_k)$ until the MSE constraint [see (4.1)] is violated.

The reconfiguration strategy derived from (4.9) indicates that it is better to power down taps with small values of $|w_k|^2/\mathcal{E}_m(w_k)$. Intuitively, this makes sense as small values of $|w_k|^2/\mathcal{E}_m(w_k)$ imply that the kth tap contributes less to the performance measure (as w_k is small) but consumes more energy [$\mathcal{E}_m(w_k)$ is large]. Note also that different multiplier models can easily be accommodated by redefining $\mathcal{E}_m(w_k)$. Furthermore, 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. This is the strategy employed in [25] and [27].

The optimum value for β_k $(k = 1, 2, \cdots, N)$ is chosen as 0 if either $\alpha_{k,\mathrm{opt}} = 0$ or the filter has converged. The justification for this is that we do not need to update the kth tap if it is not being employed in **F**-block computation. In addition, if the filter has converged, then the **WUD** portion of all the taps can be powered down. Thus, we have presented a practical reconfiguration strategy that determines the configuration parameters $\alpha_{k,\mathrm{opt}}$ and $\beta_{k,\mathrm{opt}}$ for an adaptive filter. In the next section, we employ this reconfiguration strategy for 51.84 Mb/s very high-speed digital subscriber loop (VDSL).

V. Application to 51.84 Mb/s VDSL

In this section, we employ DAT-based adaptive equalizer for 51.84 Mb/s VDSL. First, we present an overview of the VDSL environment and the VDSL transceiver.

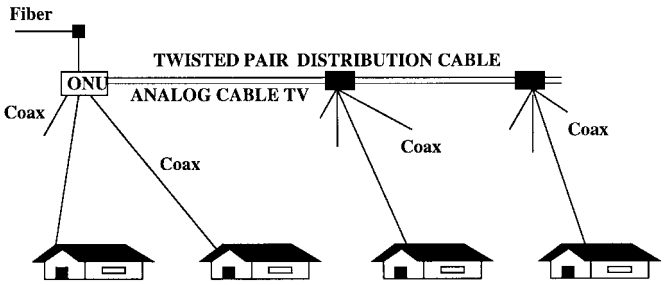


Fig. 4. VDSL environment.

A. The VDSL Environment

The VDSL application assumes a fiber-to-the-curb (FTTC) [34] network architecture. In this architecture shown in Fig. 4, the optical fiber goes to a curbside pedestal that serves a small number of homes. At the pedestal, the optical signal is converted into an electrical signal and then demultiplexed for delivery to individual homes on copper wiring. These functions are performed in an optical network unit (ONU). The ONU also performs the multiplexing and signal conversion functions required in the opposite direction, i.e., from the homes to the network. In the VDSL system considered here, the downstream channel (from the optical network unit (ONU) to the home) operates at a data rate of 51.84 Mb/s. A receiver for this data rate is conventionally designed for 1 kft cable length and 11 far-end crosstalk (FEXT) interferers. However, in practice, the cable length and the number of interferers may vary. A DAT-based receiver can exploit these variations to achieve energy savings.

Next, we briefly discuss channel and FEXT characteristics of a BKMA cable, which is employed for twisted pair distribution cable in Fig. 4. The propagation loss of a BKMA cable is similar to that of a category-5 cable specified in the TIA/EIA-568A Standard [45] and is given by

$$L_P(f) = (6.4\sqrt{f} + 0.091f)d \tag{5.1}$$

where the propagation loss $L_P(f)$ is expressed in decibels, the frequency f is expressed in megahertz, and d is the length of the cable in kilofeet. As far as FEXT is concerned, a quantity of interest is the ratio $V_r^2/V_{f\mathrm{ext}}^2$, where V_r and $V_{f\mathrm{ext}}$ are the received signal and FEXT signal, respectively. This ratio [which is also called the equal-level FEXT (EL-FEXT) loss or the input signal-to-noise ratio SNR_i in a FEXT dominated environment] can be written as:

$$\text{EL-FEXT} = \frac{V_r^2}{V_{f\mathrm{ext}}^2} = 40.75 - 20 \log f - 10 \log d + 6 \log m/n \tag{5.2}$$

where the EL-FEXT is expressed in decibels, the frequency f is expressed in megahertz, d is the length of the cable in kilofeet, m is the maximum number of crosstalk interferers in the cable, and n is the number of active crosstalk interferers. We assume $m = 24$ for this work. The FEXT impairment can be modeled as a Gaussian source because the FEXT sources are independent of each other.

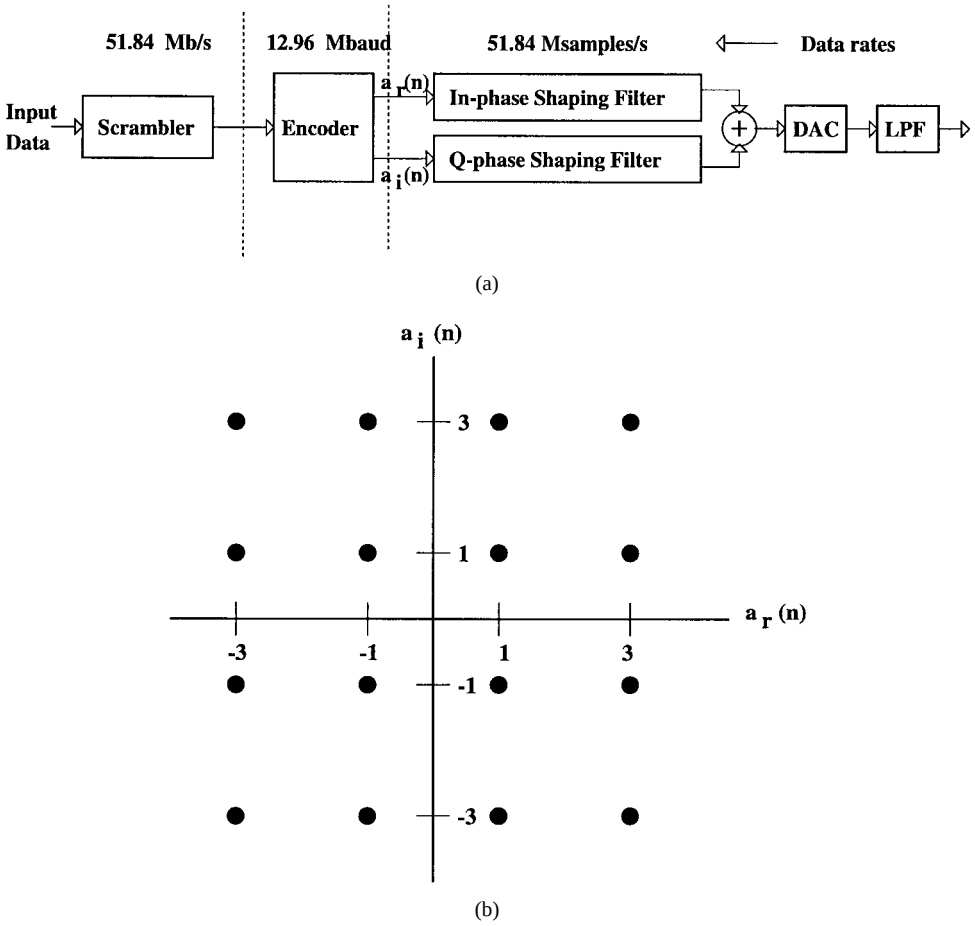


Fig. 5. Transmitter for 51.84 Mb/s VDSL. (a) Block diagram. (b) Signal constellation.

B. 51.84 Mb/s DAT-based VDSL Transceiver

In this subsection, we describe the transmitter and the receiver for 51.84 Mb/s VDSL. We will assume that the carrierless amplitude phase (CAP) [46] modulation scheme is being employed. The block diagram of a digital CAP transmitter is shown in Fig. 5(a). 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 4 bits onto one of 16 different complex symbols [see Fig. 5(b)] $a(n) = a_r(n) + ja_i(n)$ corresponding to 16-CAP (4 bits/symbol) line code. The symbols $a_r(n)$ and $a_i(n)$ are processed by digital shaping filters. The shaping filter impulse response is specified by a square-root raised cosine pulse with center frequency $f_c = 12.96$ MHz and excess bandwidth $\alpha = 38\%$. 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. We choose $f_s = 51.84$ MHz here. The outputs of the filters are subtracted, and the result is passed through a digital-to-analog (D/A) converter operating at 51.84 MHz.

On the receiver (see Fig. 6), an analog signal is first amplified by a programmable gain amplifier (PGA). The gain of PGA is controlled by a digital PGA control block. The output of PGA is passed to an A/D operating at 51.84 MHz, which converts the analog signal to the digital signal. The sampling instant of the A/D is controlled by a timing recovery block. The digital output of the A/D is processed by a decision feedback equalizer (DFE). The output of the DFE is passed through a 16-CAP slicer to obtain the output symbols. The DFE consists of the two filters: a feedforward filter and a feedback filter. The feedforward filter is a fractionally spaced linear equalizer (FSLE), which is a pair of 48-tap adaptive filters. The feedback filter is a complex 10-tap adaptive filter operating at symbol rate. A low-power strength-reduced architecture proposed in [11] is employed for implementing the complex adaptive filter. As the precision requirements and the number of taps in the feedback filter are much smaller than those in the feedforward filter, we apply DAT only to the feedforward filter.

The complexity of the DFE is reduced by simplifying the adaptation algorithm. The equalizer is blindly adapted by employing reduced constellation algorithm (RCA) [47]. In this algorithm, the adaptive filter first converges to a coarse solution based on a 4-CAP constellation (in place of a 16-point constellation). After the convergence with a 4-CAP constellation, the filter is adapted with a 16-CAP constellation. We also employ *powers-of-two approximations* [48] of the output error for the update of the coefficients. This simplification, along with powers-of-two step-sizes, allows the replacement of the multipliers in the weight-update block with shifters.

The equalizer output is passed through the slicer, decoder, and descrambler to retrieve the 51.84 Mb/s data. The algorith-

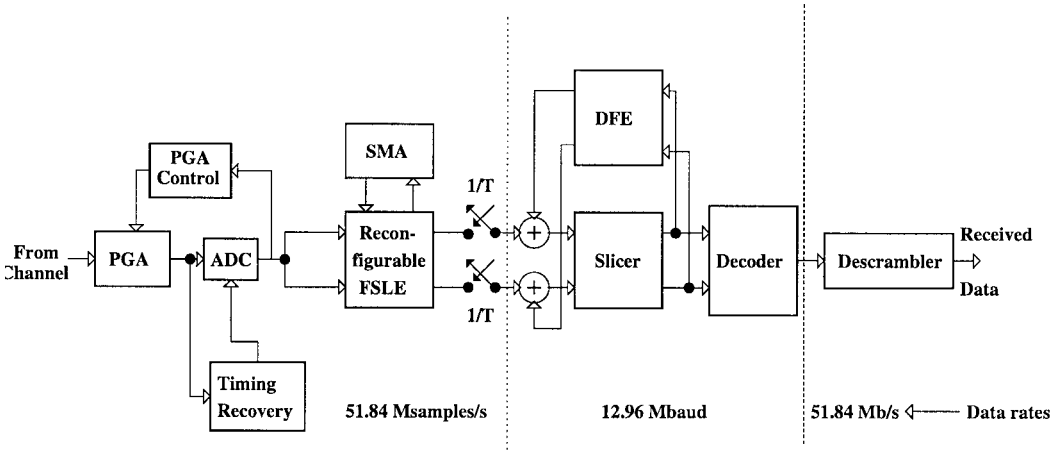


Fig. 6. DAT-based receiver for 51.84 Mb/s VDSL.

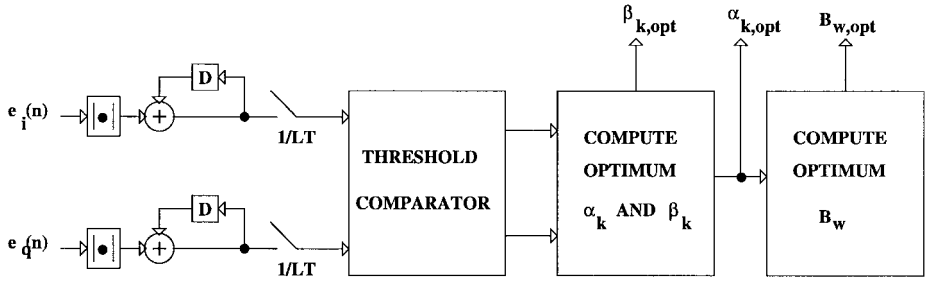


Fig. 7. **SMA** block for the DAT-based 51.84 Mb/s VDSL receiver.

mic performance measure in this case is the SNR at the slicer (SNR_o), which is equal to the ratio of signal constellation power (which equals 10 for 16-CAP) to the MSE across the slicer. For 16-CAP, an $\text{SNR}_o = 21.5$ dB is sufficient to obtain a probability of error less than 10^{-7}. To reduce undesirable glitching, we employ a window of $\delta = 2$ dB around $\text{SNR}_o = 21.5$ dB. This implies that if SNR_o lies between 21.5 and 23.5 dB, then no reconfiguration takes place. From [33], we obtain the parameters of the worst-case design as a 1-kft cable length and 11 FEXT interferers. In order to achieve $\text{SNR}_o = 21.5$ dB, the requirements for the feedforward filter are number of filter taps $N = 48$, coefficient precision $B_w = 10$ bits, and data precision $B_x = 8$ bits. Similarly, the requirements for the feedback filter are number of filter taps $N = 10$, coefficient precision $B_w = 8$ bits, and data precision $B_x = 3$ bits.

We assume that cable length can vary from 1–0.1 kft in steps of 0.1 kft. Similarly, the number of FEXT interferers can be 11, 7, or 4. These variations define the state space $\mathcal{S}$ with 30 states

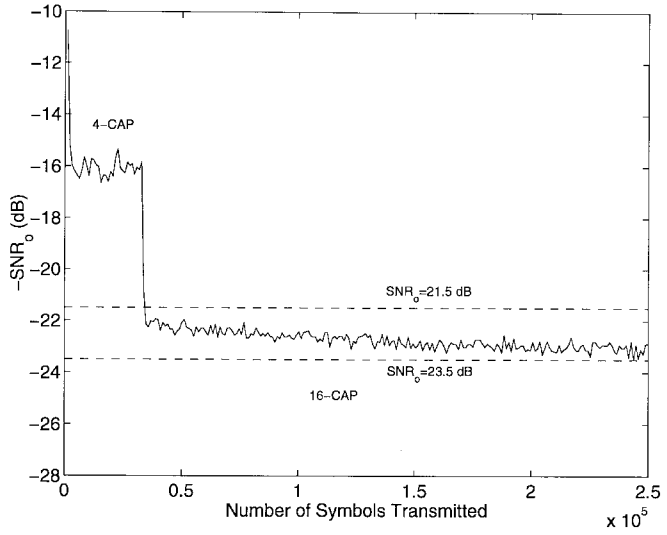
$$\mathcal{S} = \{(1\text{ kft}, 11\text{-FEXT}), (1\text{ kft}, 7\text{-FEXT}), (1\text{ kft}, 4\text{-FEXT}), \cdots, (0.1\text{ kft}, 4\text{-FEXT})\}. \quad (5.3)$$

The exact probability distribution of the states requires a survey of the installation of the VDSL network. Since this information is not known at the present moment, we assume that states in (5.3) have a Gaussian distribution with 0.6 kft and 4-FEXT as the mean values (nominal case) and standard deviation of 0.2 kft and 3 FEXT.

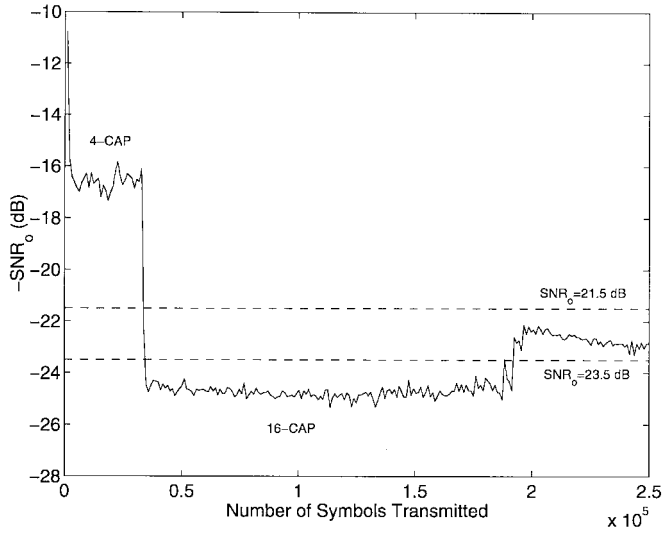
The **SMA** block detects transitions in the input states by monitoring SNR_o as a function of the cable length or the number of FEXT interferers. The optimum **SPA** configuration is then computed via the reconfiguration strategy presented in Section IV-B. The variation in SNR_o can be detected by observing $E[|e_i(n)|]$ and $E[|e_q(n)|]$, where $e_i(n)$ and $e_q(n)$ are errors across in-phase slicer and quadrature-phase slicer, respectively. As shown in Fig. 7, $E[|e_i(n)|]$ and $E[|e_q(n)|]$ are computed by summing $|e_i(n)|$ and $|e_q(n)|$ over L symbols with L being chosen to be 4096 for this experiment. All the blocks before and including threshold comparator are always powered up. The subblocks after the threshold comparator in Fig. 7 compute the energy-optimum value of α_ks, β_ks, and B_w only when a state transition occurs.

C. Simulation Results

In this subsection, we present simulation results for 51.84 Mb/s VDSL in terms of converged configurations SNR_o and energy consumption. The SNR_o is computed as a ratio of the signal constellation power (which is 10 for 16-CAP) and MSE across the 16-CAP slicer. Multiplier energy consumption $\mathcal{E}_m(w_k)$ is obtained via a real-delay gate-level simulator MED [44], assuming a 0.18 μm, 2.5 V CMOS technology. The energy consumption of the equalizer is then obtained by summing the energy values of the powered-up multipliers in the **F** block. We assume that all multipliers are powered-up in the worst-case design. The energy consumption of the **SMA** block is due to the blocks up to the threshold comparator in



(a)



(b)

Fig. 8. SNR$_o$ convergence plots for the DAT-based 51.84 Mb/s VDSL receiver. (a) The 1 kft cable length and 11-FEXT. (b) The 1 kft cable length and 4-FEXT.

Fig. 7 and is assumed fixed at 0.061 mW/MHz. The energy savings for a DAT-based design are computed via (3.2).

In Fig. 8, we show SNR$_o$ convergence curves for a DAT-based VDSL equalizer for two input states. Fig. 8(a) shows convergence curve for the worst-case input state corresponding to a 1-kft cable length and 11 FEXT interferers. Recall that the equalizer is adapted blindly via the reduced constellation algorithm (RCA). In the 4-CAP constellation mode, the equalizer converges to an SNR$_o$ of 16 dB after 32 768 symbols, after which it is switched to a 16-CAP constellation mode. Finally, the equalizer converges to an SNR$_o$ of 22.5 dB, which is within the SNR$_o$ constraint window of [21.5 dB, 23.5 dB]. Now, consider the situation where the cable length is 1 kft and the number of FEXT interferers is 4. The SNR$_o$ convergence curve is plotted in Fig. 8(b). The adaptive filter converges to an SNR$_o$ of 24.5 dB in this case, thereby falling outside the desired SNR window [21.5 dB, 23.5 dB]. The **SMA** block detects the surplus SNR$_o$ and activates the reconfiguration strategy after 131 072 symbols in Fig. 8(b). This is done by powering down taps one at a time starting with the one with the smallest value of $[w_k^2/\mathcal{E}_m(w_k)]$ until the final SNR$_o$ lies between 21.5–23.5 dB. The final configuration vector of the in-phase adaptive filter is $\underline{\alpha} = [000\,000\,000\,000\,000\,000\,000\,011\,111\,100\,011\,100\,010\,101\,011\,101]$, where a "1" indicates a tap that is powered up, and "0" indicates that it is powered down. It can be seen that only 16 (out of 48) taps are powered up. Unlike [25] and [37], taps other than those at the end are also powered down. Similarly, the final configuration vector for the quadrature-phase filter $\underline{\alpha} = [000\,000\,000\,000\,000\,000\,001\,111\,111\,100\,101\,100\,101\,010\,101\,010]$, which corresponds to 17 powered-up taps. In this configuration, an SNR$_o$ of 22.2 dB is achieved. Overall, the filter dissipates 63% less energy as compared with the worst-case design.

TABLE I
CONVERGED CONFIGURATIONS

Cable Length (kft)	Number of FEXT interferers	In-phase filter $\sum \alpha_{k,opt}$	In-phase filter $B_{w,opt}$	Quadrature-phase filter $\sum \alpha_{k,opt}$	Quadrature-phase filter $B_{w,opt}$
1.0	11	48	10	48	10
	7	48	10	48	10
	4	16	10	17	10
0.9	11	48	10	48	10
	7	48	10	48	10
	4	10	9	10	9
0.8	11	48	10	48	10
	7	41	10	42	10
	4	12	9	11	9
0.7	11	48	10	48	10
	7	15	10	14	10
	4	14	10	7	9
0.6	11	45	10	44	10
	7	15	10	13	10
	4	8	9	8	9
0.5	11	27	10	23	10
	7	12	9	11	9
	4	10	9	14	10
0.4	11	13	10	8	9
	7	16	10	11	9
	4	9	9	9	9
0.3	11	8	9	9	9
	7	10	9	9	9
	4	9	9	10	9
0.2	11	9	9	9	9
	7	9	9	6	9
	4	10	9	8	9
0.1	11	11	9	8	9
	7	14	10	8	9
	4	15	10	9	9

Table I shows the converged configurations for in-phase and quadrature-phase filters. It can be seen that the number of powered-up taps (i.e., $\sum \alpha_{k,\mathrm{opt}}$) decreases as the cable length decreases. Similarly, the number of powered-up taps reduces with a reduction in the number of FEXT interferers for a fixed cable length. From Table I and consistent with (2.5), we observe that B_w reduces by 1 bit for shorter cable lengths. For example, the number of powered-up taps in the in-phase filter ranges from 9–48, B_w ranges from 9–10 for cable lengths ranging from 100 ft to 1000 ft, and the number of FEXT interferers is fixed at 11. The number of powered-up taps in the in-phase filter range from 16–48 when the number of FEXT interferers varies from 4–11, and the cable length is fixed at 1 kft. Similar observations can be made for the number of powered-up taps and precision of the quadrature-phase filter.

Table II shows the converged SNR$_o$ for equalizers based on both the worst-case design and DAT-based design. It can be

TABLE II
CONVERGED SNR_o

Cable Length (kft)	Number of FEXT interferers	Worst-case design SNR_o (dB)	DAT-based design SNR_o (dB)
1.0	11	21.3 dB	21.3 dB
	7	21.8 dB	21.8 dB
	4	23.1 dB	22.2 dB
0.9	11	21.6 dB	21.6 dB
	7	22.6 dB	22.6 dB
	4	23.8 dB	22.5 dB
0.8	11	21.9 dB	21.9 dB
	7	22.9 dB	22.5 dB
	4	24.2 dB	22.2 dB
0.7	11	22.4 dB	22.4 dB
	7	23.3 dB	22.3 dB
	4	24.2 dB	22.4 dB
0.6	11	22.8 dB	22.0 dB
	7	23.9 dB	22.5 dB
	4	25.0 dB	22.8 dB
0.5	11	23.5 dB	22.2 dB
	7	24.3 dB	22.6 dB
	4	25.5 dB	22.9 dB
0.4	11	24.1 dB	22.3 dB
	7	25.2 dB	22.7 dB
	4	26.1 dB	23.0 dB
0.3	11	25.0 dB	22.4 dB
	7	25.9 dB	23.0 dB
	4	27.1 dB	23.3 dB
0.2	11	26.3 dB	23.8 dB
	7	27.2 dB	24.0 dB
	4	28.5 dB	23.2 dB
0.1	11	28.8 dB	23.6 dB
	7	29.7 dB	23.2 dB
	4	30.5 dB	23.1 dB

TABLE III
ENERGY CONSUMPTION

Cable length (kft)	Number of FEXT interferers	$p(\mathbf{s}_i)$	Worst-case design $\mathcal{E}_{WC}$ (mW/MHz)	DAT-based design $\mathcal{E}_{SPA}$ (mW/MHz)	$\mathcal{E}_{SMA}$ (mW/MHz)	$\mathcal{E}_{DAT}$ (mW/MHz)	% $\mathcal{E}_{sav}$
1.0	11	0.007	3.498	3.498	0.061	3.559	-1.8
	7	0.014	3.439	3.439	0.061	3.500	-1.8
	4	0.007	3.408	1.199	0.061	1.260	63.0
0.9	11	0.016	3.412	3.412	0.061	3.474	-1.8
	7	0.033	3.417	3.417	0.061	3.479	-1.8
	4	0.016	3.426	0.684	0.061	0.745	78.3
0.8	11	0.031	3.395	3.395	0.061	3.456	-1.8
	7	0.061	3.262	2.981	0.061	3.042	6.7
	4	0.031	3.334	0.852	0.061	0.914	72.6
0.7	11	0.045	3.452	3.452	0.061	3.514	-1.8
	7	0.089	3.169	1.012	0.061	1.073	66.1
	4	0.045	3.405	0.855	0.061	0.916	73.1
0.6	11	0.051	3.223	2.971	0.061	3.033	5.9
	7	0.101	3.418	1.144	0.061	1.205	64.7
	4	0.051	3.196	0.553	0.061	0.614	80.8
0.5	11	0.045	3.124	1.445	0.061	1.506	51.8
	7	0.089	3.230	0.744	0.061	0.805	75.1
	4	0.045	2.847	0.809	0.061	0.870	69.4
0.4	11	0.031	2.844	0.623	0.061	0.684	75.9
	7	0.061	3.169	0.915	0.061	0.977	69.2
	4	0.031	2.978	0.611	0.061	0.672	77.4
0.3	11	0.016	3.090	0.571	0.061	0.632	79.5
	7	0.033	3.257	0.581	0.061	0.642	80.3
	4	0.016	3.135	0.665	0.061	0.727	76.8
0.2	11	0.007	3.189	0.550	0.061	0.611	80.8
	7	0.014	3.099	0.537	0.061	0.599	80.7
	4	0.007	3.347	0.601	0.061	0.663	80.2
0.1	11	0.002	3.326	0.659	0.061	0.721	78.3
	7	0.004	3.484	0.723	0.061	0.785	77.5
	4	0.002	3.294	0.815	0.061	0.876	73.4
Average			**3.240**	**1.471**	**0.061**	**1.532**	**52.7**

seen that the SNR_o for the worst-case design increases from 21.3–28.8 dB as the cable length decreases from 1–0.1 kft and the number of FEXT interferers is fixed at 11. The SNR_o ranges from 21.3–23.1 dB as the number of FEXT interferers reduces from 4–11 with the cable length fixed at 1 kft. Overall, the DAT-based receiver maintains the SNR_o within the range of 21.5–23.5 dB for all the states.

Table III compares the energy consumption of the worst-case and DAT-based designs. The energy consumption of the worst-case design varies from 2.8–3.5 mW/MHz even though the configuration is fixed. This variation reflects the variation of the energy consumption on input data. The average energy consumption of the worst-case design was found to be 3.2 mW/MHz. The energy consumption of the DAT-based design varies from 0.6–3.6 mW/MHz. Of this, the energy of the **SPA** block varies from 0.5–3.5 mW/MHz, whereas that of the **SMA** block is 0.06 mW/MHz. Employing (3.2), we find that the energy savings range from -2 to 81% with an average of 53% assuming a Gaussian distribution for the input states. Thus, it can be seen that the DAT-based approach is quite attractive from the viewpoint of energy savings for the VDSL application.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented dynamic algorithm transforms (DAT's) as a systematic method for designing low-power application-specific reconfigurable DSP systems. In particular, we employed DAT in the equalization scenario for a 51.84 Mb/s VDSL transceiver. Substantial energy savings are observed due to variations in the cable length and the number of FEXT interferers. DAT techniques jointly optimize system performance and energy dissipation, thus representing a growing trend to synergize across the design hierarchy. In addition, DAT provides a convenient framework within which ongoing research in the areas of data-adaptive DSP algorithms and reconfigurable circuits and architectures can be synergistically combined to enable the design of energy-efficient reconfigurable DSP systems.

APPENDIX A
DERIVATIVE OF (4.9)

Consider the optimization problem in (4.8).

$$\max_{\lambda\geq 0} \min_{\substack{\underline{\alpha}\in\{0,1\}^N \\ \mathcal{J}(\underline{\alpha})\leq\mathcal{J}_o}} \mathcal{L}(\underline{\alpha},\lambda) = \left[\sum_{k=1}^{N} \alpha_k[\mathcal{E}_m(w_k) - \lambda|w_k|^2\sigma_x^2] + \lambda(\sigma_d^2 - \mathcal{J}_o)\right]. \quad \text{(A1)}$$

We will solve (A1) by first determining the solution α_k^* of the inner optimization problem

$$\min_{\substack{\underline{\alpha}\in\{0,1\}^N \\ \mathcal{J}(\underline{\alpha})\leq\mathcal{J}_o}} \sum_{k=1}^{N} \alpha_k[\mathcal{E}_m(w_k) - \lambda|w_k|^2\sigma_x^2] \quad \text{(A2)}$$

as a function of λ and then proving that the outer optimization problem

$$\max_{\lambda\geq 0} \left[q(\lambda) \triangleq \min_{\substack{\underline{\alpha}\in\{0,1\}^N \\ \mathcal{J}(\underline{\alpha})\leq\mathcal{J}_o}} \mathcal{L}(\underline{\alpha},\lambda)\right] \quad \text{(A3)}$$

has a solution λ^*. The optimum value $\alpha_{k,\mathrm{opt}}$ can then be obtained by the substitution $\lambda = \lambda^*$ in the solution to (A2). The solution to (A2) is derived via the following theorem.

Theorem 1: For the following optimization problem:

$$\min_{\underline{\alpha}\in\{0,1\}^N} \sum_{k=1}^{N} \alpha_k[\mathcal{E}_m(w_k) - \lambda|w_k|^2\sigma_x^2] \tag{A4}$$

the optimum value of α_k is given by

$$\alpha_k^*(\lambda) = \begin{cases} 1, & \text{if } \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} > \tilde{\lambda} \\ 0, & \text{if } \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} \le \tilde{\lambda} \end{cases} \tag{A5}$$

where $\tilde{\lambda} = 1/(\sigma_x^2\lambda)$.

Proof: Define function $f_k(\lambda)$ as

$$f_k(\lambda) = \mathcal{E}_m(w_k) - \lambda|w_k|^2\sigma_x^2, \qquad k = 1, 2\cdots, N. \tag{A6}$$

Then, the objective function in (A4) can be written as

$$\begin{aligned}\mathcal{F}(\underline{\alpha}, \lambda) &= \sum_{k=1}^{N} \alpha_k f_k(\lambda) \\ &= \alpha_1 f_1(\lambda) + \alpha_2 f_2(\lambda) + \cdots + \alpha_N f_N(\lambda)\end{aligned} \tag{A7}$$

where $f_k(\lambda)$ is as defined in (A6). From (A7), it is clear that the terms for different indices k are decoupled from each other. Therefore, for a particular index k, the optimum α_k can be obtained by solving the following optimization problem:

$$\min_{\alpha_k\in\{0,1\}} \alpha_k f_k(\lambda) \tag{A8}$$

where we want to find the optimum value of α_k, and $f_k(\lambda)$ is a term independent of α_k. The optimum value $\alpha_k^*(\lambda)$ is obtained as follows.

Case 1: $(f_k(\lambda) \le 0)$

$$\begin{aligned} f_k(\lambda) &\le 0 \\ \Rightarrow \alpha_k f_k(\lambda)|_{\alpha_k=1} &\le \alpha_k f_k(\lambda)|_{\alpha_k=0} \\ \Rightarrow \alpha_k^*(\lambda) &= 1. \end{aligned}$$

Case 2: $(f_k(\lambda) \ge 0)$

$$\begin{aligned} f_k(\lambda) &\ge 0 \\ \Rightarrow \alpha_k f_k(\lambda)|_{\alpha_k=1} &\ge \alpha_k f_k(\lambda)|_{\alpha_k=0} \\ \Rightarrow \alpha_k^*(\lambda) &= 0. \end{aligned}$$

Combining Cases 1 and 2 above, we get the following solution:

$$\alpha_k^*(\lambda) = \begin{cases} 1, & f_k(\lambda) < 0, \\ 0 & f_k(\lambda) \ge 0. \end{cases} \tag{A9}$$

Substituting $f_k(\lambda)$ from (A6) into (A9) and rearranging, we obtain

$$\alpha_k^*(\lambda) = \begin{cases} 1, & \text{if } \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} > \dfrac{1}{\sigma_x^2\lambda} \\ 0, & \text{if } \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} \le \dfrac{1}{\sigma_x^2\lambda} \end{cases} \tag{A10}$$

which is identical to (A5). ♣

Next, we prove that the optimization problem (A3) has a well-defined maximum point λ^*.

Theorem 2: $q(\lambda)$ defined in (A3) is a concave function.

Proof: For any $\underline{\alpha}, \lambda_1, \lambda_2$ and $\epsilon \in [0, 1]$, we have from (A1)

$$\mathcal{L}(\underline{\alpha}, \epsilon\lambda_1 + (1-\epsilon)\lambda_2) = \epsilon\mathcal{L}(\underline{\alpha}, \lambda_1) + (1-\epsilon)\mathcal{L}(\underline{\alpha}, \lambda_2). \tag{A11}$$

Taking the minimum over $\alpha_k \in \{0, 1\}$ of both sides in (A11), we obtain

$$\begin{aligned}&\min_{\alpha_k\in\{0,1\}} \mathcal{L}(\underline{\alpha}, \epsilon\lambda_1 + (1-\epsilon)\lambda_2) \\ &\quad\ge \epsilon \min_{\alpha_k\in\{0,1\}} \mathcal{L}(\underline{\alpha}, \lambda_1) + (1-\epsilon) \min_{\alpha_k\in\{0,1\}} \mathcal{L}(\underline{\alpha}, \lambda_2)\end{aligned} \tag{A12}$$

which leads to

$$q(\epsilon\lambda_1 + (1-\epsilon)\lambda_2) \ge \epsilon q(\lambda_1) + (1-\epsilon)q(\lambda_2). \tag{A13}$$

Hence, $q(\lambda)$ is a concave function of λ. ♣

Thus, we have shown that $q(\lambda)$ is concave, and hence, it has a well-defined maxima λ^*. Therefore, $\alpha_{k,\text{opt}}$ can be obtained by substituting $\lambda = \lambda^*$ in (A5) as

$$\alpha_{k,\text{opt}} = \begin{cases} 1, & \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} > \tilde{\lambda} \\ 0, & \dfrac{|w_k|^2}{\mathcal{E}_m(w_k)} \le \tilde{\lambda}, \end{cases} \tag{A14}$$

which is identical to (4.9).

REFERENCES

[1] A. Chandrakasan and R. W. Brodersen, "Minimizing power consumption in digital CMOS circuits," *Proc. IEEE*, vol. 83, pp. 498–523, Apr. 1995.

[2] M. Horowitz, T. Indermaur, and R. Gonzalez, "Low-power digital design," in *Proc. IEEE Symp. Low Power Electron.*, Oct. 1994, pp. 8–11.

[3] A. Shen, A. Ghosh, S. Devdas, and K. Keutzer, "On average power dissipation and random pattern testability of CMOS combinational logic networks," in *Proc. IEEE Int. Conf. Comput.-Aided Des.*, 1992, pp. 402–407.

[4] S. Iman and M. Pedram, "An approach for multi-level logic optimization targeting low-power," *IEEE Trans. Comput.-Aided Design*, vol. 15, pp. 889–901, Aug. 1996.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

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

[10] 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.

[11] 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, pp. 1276–1290, May 1997.

[12] S. Ramprasad, N. R. Shanbhag, and I. N. Hajj, "Decorrelating (DECOR) transformations for low-power digital filters," *IEEE Trans. Circuits Syst. II*, vol. 45, pp. 776–788, June 1999.

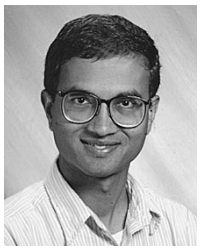
[13] H. H. Loomis and B. Sinha, "High speed recursive digital filter realization," *Circuits, Syst., Signal Process.*, vol. 3, no. 3, pp. 267–294, 1984.

[14] K. K. Parhi and D. G. Messerschmitt, "Pipeline interleaving and parallelism in recursive digital filters—Parts I & II," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 1099–1134, July 1989.
[15] K. K. Parhi, "Algorithm transformation techniques for concurrent processors," *Proc. IEEE*, vol. 77, pp. 1879–1895, Dec. 1989.
[16] N. R. Shanbhag and K. K. Parhi, *Pipelined Adaptive Digital Filters*. Boston, MA: Kluwer, 1994.
[17] M. Potkonjak and J. Rabaey, "Fast implementation of recursive programs using transformations," in *Proc. IEEE ICASSP*, San Francisco, CA, Mar. 1992, pp. 569–572.
[18] C. Leiserson and J. Saxe, "Optimizing synchronous systems," *J. VLSI Comput. Syst.*, vol. 1, pp. 41–67, 1983.
[19] A. Abnous and J. M. Rabaey, "Ultra-low-power domain-specific multimedia processors," in *Proc. IEEE VLSI Signal Process. Workshop*, Oct. 1996.
[20] J. Villasenor and B. Hutchings, "The flexibility of configurable computing," *IEEE Signal Processing Mag.*, pp. 67–84, Sept. 1998.
[21] S. Trimberger, D. Carberry, A. Johnson, and J. Wong, "A time-multiplexed FPGA," in *Proc. IEEE Workshop FPGA's for Custom Comput. Machines*, J. Arnold and K. L. Pocek, Eds., Napa, CA, Apr. 1997, pp. 22–28.
[22] 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 *Proc. Field-Programmable Logic Appl. 5th Int. Workshop Field-Programmable Logic Appl.*, W. Moore and W. Luk, Eds. Oxford, U.K.: Springer-Verlag, Sept. 1995, pp. 282–292.
[23] 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.
[24] C. Rupp, M. Landguth, T. Garverick, E. Gomersall, H. Holt, J. Arnold, and M. Gokhale, "The NAPA adaptive processing architecture," in *Proc. IEEE Symp. Field-Programmable Custom Comput. Machines*, Napa, CA, Apr. 1998.
[25] 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.
[26] V. Gutnik and A. P. Chandrakasan, "Embedded power supply for low-power DSP," *IEEE Trans. VLSI Systems*, vol. 12, pp. 425–435, Dec. 1997.
[27] K. Wesolowski, "Adaptive blind equalizers with automatically controlled parameters," *IEEE Trans. Commun.*, vol. 43, pp. 170–172, Feb. 1995.
[28] B. Vucetic, "An adaptive coding scheme for time-varying channels," *IEEE Trans. Commun.*, vol. 39, pp. 653–663, May 1991.
[29] S. Appadwedula, D. L. Jones, K. Ramchandran, and L. Qian, "Joint source channel matching for wireless image transmission," in *Proc. Int. Conf. Image Process.*, Chicago, IL, Oct. 1998.
[30] M. Goel and N. R. Shanbhag, "Dynamic algorithm transformations (DAT) for low-power adaptive signal processing," in *Proc. Int. Symp. Low Power Electron. Des. (ISLPED)*, Monterey, CA, Aug. 1997, pp. 161–166.
[31] ———, "Dynamic algorithm transformations (DAT): A systematic approach to low-power reconfigurable signal processing," *IEEE Trans. VLSI Syst.*, to be published.
[32] P. S. Chow, J. C. Tu, and J. M. Cioffi, "Performance evaluation of a multichannel transceiver system for ADSL and VHDSL services," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 909–919, Aug. 1991.
[33] N. R. Shanbhag and G.-H. Im, "VLSI system design of 51.84 Mb/s transceiver for ATM-LAN and broadband access," *IEEE Trans. Signal Processing*, vol. 46, pp. 1403–1416, May 1998.
[34] Davic 1.0 Specification Part 8, *Lower Layer Protocols and Physical Interfaces*, Jan. 1996.
[35] 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 Process. Syst. (SiPS): Des. Implement.*, Boston, MA, Oct. 1998, pp. 317–326.
[36] S. A. Raghavan, J. K. Wolf, L. B. Milstein, and L. C. Barbosa, "Nonuniformly spaced tapped-delay-line equalizers," *IEEE Trans. Commun.*, vol. 41, pp. 1290–1295, Sept. 1993.
[37] 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.
[38] D. P. Bertsekas, *Nonlinear Programming*. Boston, MA: Athena Scientific, 1995.
[39] B. Widrow *et al.*, "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proc. IEEE*, vol. 64, pp. 1151–1162, Aug. 1976.
[40] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1991.
[41] M. Goel and N. R. Shanbhag, "Finite precision analysis of the pipelined strength-reduced adaptive filter," *IEEE Trans. Signal Processing*, vol. 46, pp. 1763–1769, June 1998.
[42] F. Najm, "A survey of power estimation techniques in VLSI circuits," *IEEE Trans. VLSI Syst.*, vol. 2, pp. 446–455, Dec. 1994.
[43] M. Pedram, "Power minimization in IC design: principles and applications," *ACM Trans. Design Automat. Electron. Syst.*, vol. 1, no. 1, pp. 3–56, Jan. 1996.
[44] M. G. Xakellis and F. N. Najm, "Statistical estimation of the switching activity in digital circuits," in *Proc. Design Automat. Conf.*, June 1994, pp. 728–733.
[45] TIA/EIA-568-A Standard, *Commercial Building Telecommunications Cabling Standard*, 1994.
[46] G. H. Im and J. J. Werner, "51.84 Mb/s 16-CAP ATM-LAN standard," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 620–632, May 1995.
[47] R. D. Gitlin, J. F. Hayes, and S. B. Weinstein, *Data Communications Principles*. New York: Plenum, 1992.
[48] P. Xue and B. Liu, "Adaptive equalizer using finite-bit power-of-two quantizer," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 1603–1611, Dec. 1986.

Manish Goel received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, New Delhi, in 1994 and the M.S. degree in electrical and computer engineering from University of Illinois at Urbana-Champaign, Urbana, in 1997. He is currently a Ph.D. candidate in the Electrical and Computer Engineering Department, University of Illinois at Urbana-Champaign.

His research interests are in the 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 from Indian Institute of Technology, New Delhi.



Naresh R. Shanbhag (SM'99) received the B.Tech. degree from the Indian Institute of Technology, New Delhi, in 1988, the M.S. degree from Wright State University, Dayton, OH, in 1990, and the Ph.D. degree from University of Minnesota, Minneapolis, in 1993, all in electrical engineering.

From July 1993 to August 1995, he worked at AT&T Bell Laboratories, Murray Hill, NJ, with the Wide-Area Networks Group, where he was responsible of 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 asynchronous transfer mode (ATM)-LAN and broadband access chip sets. In August 1995, he joined the Coordinated Science Laboratory and the Electrical and Computer Engineering Department, University of Illinois at Urbana-Champaign as an Assistant Professor. His research interests are in exploring the limits of computation in an integrated circuit media in terms of power dissipation, reliability, and throughput; VLSI architectures; and algorithms for signal processing and communications and computer aided-design tools for system exploration. This includes the design of high-speed and/or low-power algorithms for speech and video processing, adaptive filtering, and high-bit rate digital communications systems.

Dr. Shanbhag received the 1999 Xerox Faculty Award, 1999 IEEE Leon K. Kirchmayer Best Paper Award, the National Science Foundation CAREER Award in 1996, and the 1994 Darlington Best Paper Award from the IEEE Circuits and Systems Society. Since July 1997, he has been a Distinguished Lecturer for IEEE Circuits and Systems Society and an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS: PART II. He is also a co-author of the research monograph *Pipelined Adaptive Digital Filters* (Boston, MA: Kluwer, 1994).