Analytical Estimation Of Signal Transition Activity From Word-Level Statistics

Sumant Ramprasad, Naresh R. Shanbhag, and Ibrahim N. Hajj

Abstract

Presented in this paper is a novel methodology to determine the average number of transitions in a signal from its word-level statistical description. The proposed methodology employs: 1.) high-level signal statistics, 2.) a statistical signal generation model, and 3.) the signal encoding (or number representation) to estimate the transition activity for that signal. In particular, the signal statistics employed are mean (μ), variance (σ^2), and autocorrelation (ρ). The signal generation models considered are auto-regressive moving-average ($ARMA$) models. The signal encoding includes unsigned, one's complement, two's complement, and sign-magnitude representations. First, the following *exact* relation between the transition activity (t_i), bit-level probability (p_i) and the bit-level autocorrelation (ρ_i) for a single bit signal b_i is derived,

$$t_i \quad = \quad 2p_i(1-p_i)(1-\rho_i) \tag{1}$$

Next, two techniques are presented which employ the word-level signal statistics, the signal generation model, and the signal encoding to determine ρ_i $(i = 0, \ldots, B-1)$ in (1) for an B-bit signal. The word-level transition activity T is obtained as a summation over t_i $(i = 0, \ldots, B-1)$, where t_i is obtained from (1). Simulation results for 16 bit signals generated via $ARMA$ models indicate that an error in T of less than 2% can be achieved. Employing $AR(1)$ and $MA(10)$ models for audio and video signals, the proposed method results in errors of less than 10%. Both analysis and simulations indicate the sign-magnitude representation to have lower transition activity than unsigned, ones' complement, or two's complement. Finally, the proposed method is employed in estimation of transition activity in digital signal processing (DSP) hardware. Signal statistics are propagated through various DSP operators such as adders, multipliers, multiplexors, and delays and then the transition activity T is calculated. Simulation results with $ARMA$ inputs show that errors less than 4% are achievable in the estimation of the total transition activity in the filters. Furthermore, the transpose form structure is shown to have fewer signal transitions as compared to the direct form structure for the same input.

I. INTRODUCTION

Power dissipation has become a critical design concern in recent years driven by the emergence of mobile applications. Reliability concerns and packaging costs have made power optimization relevant even for tethered applications. As system designers strive to integrate multiple-systems on-chip, power dissipation has become an equally important parameter that needs to be optimized along with area and speed. Therefore, extensive research into various aspects of low-power system design is presently being conducted. We may classify this research into: 1.) *power reduction* techniques [6,7,9]; 2.) *low-power synthesis* techniques [5,11,31]; 3.) *power estimation* [20]; and 4.) *fundamental limits* on power dissipation [30,33]. While the work presented in this paper focuses on 3.), our eventual objective is to enable 2.).

Power reduction techniques form an integral part of low-power VLSI systems design and is presently an active area of research [6,7,9]. These 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 reduced complexity algorithms [6]), architectural level (such as pipelining [12,25] and parallel processing), logic (logic minimization [31] and precomputation [1]), circuit (reduced voltage swing [21], adiabatic logic [3]) and technological level [8]. It is now well recognized that an astute algorithmic and architectural design can have a large impact on the final power dissipation characteristics of the fabricated VLSI solution. Therefore, there is a great need for techniques which allow the evaluation of different architectures from the viewpoint of power dissipation and be able to accurately estimate their power dissipation.

Power dissipation in CMOS VLSI circuits is a direct function of the number of signal transitions occuring at the capacitive nodes present in it. The terms *switching activity*, *transition probability* [20], *transition density* [19] and *transition activity* [10] have been proposed in the past to provide a measure of the number of signal transitions. Switching activity and transition probability indicate the average number of transitions at a node per clock cycle. The term transition density equals the average number of transitions per unit time. Transition activity has been employed in [10] to indicate the average number of transitions in a clock cycle present in a bit of a signal word, in a word, and within a module. Here, we will employ the terminology transition activity as in [10] without any ambiguity.

At the logic and circuit levels, techniques such as [13–15,17,19,29,32] exist for power estimation. While these techniques provide accurate estimates of power dissipation, they require a gate or transistor level description of the circuit. Therefore, such techniques are applicable once the design has reached a substantial degree of maturity. Our interest in this paper is to enable power estimation at a higher level, which in this case is the architectural level. In the present context, an architectural description refers to the register-transfer level (RTL) model of the system. Architectural level power estimation tools will allow the system designer to choose between competing architectures and also permit major design changes when it is easiest to do so.

While a large amount of work has been done at the circuit and logic levels, not much work has been done for power estimation at the architectural level. In [22], a technique based upon the concept of *entropy* was presented for estimating the average transition density inside a combinational circuit. This technique employs the Boolean relationship between its input and output. The closest approach to our work, however, is the Dual Bit Type (DBT) model described in [10] where a word-level signal is broken up into: 1.) uncorrelated data bits, 2.) correlated data bits, and 3.) sign bits. The uncorrelated data bits are from the least significant bit (LSB) up to a certain break-point BP_0, with a fixed transition activity. The transition activity of the sign bits, which are from the most significant bit (MSB) to another break-point BP_1, are measured by an RTL simulation. A linear model is then employed for the switching activity of correlated data bits, which lie between the sign bits and uncorrelated data bits. Empirical equations defining BP_0 and BP_1 in terms of word-level statistics such as mean (μ), variance (σ^2), and autocorrelation (ρ) was also presented.

Our approach considers the same problem as [10] in that we present a methodology for estimating the average number of transitions in a signal from its word-level statistical description. However, unlike [10] where the estimation of transition activity is based on simulation, the proposed methodology is analytical requiring: 1.) high-level signal statistics, 2.) a statistical signal generation model, and 3.) the signal encoding (or number representation) to estimate the transition activity for that signal. Therefore, the two novel features of the proposed method are: 1.) it is a completely *analytical* approach and 2.) its computational complexity is independent of the *length* (i.e., number of samples) of the signal. Both of these features distinguish the proposed approach from most existing techniques to estimate signal transition activity. While [10] also estimates power dissipation by characterizing input capacitance, we focus only on the estimation of transition activity.

We first derive a new relation between the bit-level transition activity (t_i), bit-level probability (p_i) and the bit-level autocorrelation (ρ_i) for a single bit signal b_i. Then, we present 2 methods, the first exact but computationally expensive and the second fast but approximate, to estimate the word-level transition activity, T, employing word-level signal statistics (namely μ, σ, and ρ), signal generation models (such as auto-regressive (AR), moving-average (MA) and auto-regressive moving-average ($ARMA$) models), along with a certain number representation (such as unsigned, sign-magnitude, one's complement or two's complement). In the approximate method, we divide a word into three regions based on the temporal correlations, unlike [10], where a word is divided into three regions based on the transition activities. Such an approach enables us to estimate the transition activity analytically. The approximate method also uses different and more accurate formulae for estimating the break-points BP_0 and BP_1. Proceeding further, we describe the propagation of the input statistics through commonly used digital signal processing (DSP) blocks such as adders, multipliers, multiplexors, and delays. The effect of the *folding* transformation [26] on signal statistics is also studied. The word-level transition activities of all the signals in a system composed of these DSP blocks is determined. These are then summed up to determine the total transition activity for the filter. Even though we focus upon architectural level power estimation in this paper, we believe that the work presented here would lead to a formal procedure for the synthesis of low-power DSP hardware. The transition activities estimated at the inputs and outputs to blocks such as adders, multipliers, multiplexors, and delays can be used to estimate power dissipation within the block using a power macro-model [16].

The paper is organized as follows. In section II, we present some preliminaries and existing results. Determining word-level transition activity T from word-level signal properties is described in section III. In section IV, we compute transition activity for various filter structures and in section V we present simulation results for audio, video, and communication system signals and filters.

II. PRELIMINARIES

In this section, we will present definitions and review existing results that will be employed in later sections. First, we will define the *word-level* quantities such as the mean (μ), variance (σ^2), and temporal correlation (ρ). Next, we consider *bit-level* quantities such as the probability p_i of the i^{th} bit b_i being equal to a 1, the bit-level temporal correlation ρ_i, the bit-level transition activity t_i. Finally, the structures of the AR, MA, and $ARMA$ models are described.

A. Word and Bit-Level Quantities

Let $x(n)$ be a B-bit word signal given by

$$x(n) \quad = \quad \sum_{i=0}^{B-1} c_i b_i(n), \tag{2}$$

where $b_i(n) \in \{0,1\}$ represents the i^{th} bit, c_i are the weights, and n is the time index. For example, in case of unsigned number representation we have $c_i = 2^i$.

For $x(n)$ in (2), the *mean* μ or the average (or expected value) of $x(n)$ is defined as

$$\mu = E[x(n)] = \sum_{\forall k \in \mathcal{X}} k \Pr(x(n) = k), \tag{3}$$

where the elements of the set $\mathcal{X}$ are the values that $x(n)$ can assume, and $\Pr(A)$ is the probability that event A occurs. Note that the elements of the set $\mathcal{X}$ are a function of the signal encoding or the number representation.

Similarly, the *variance* σ^2 of $x(n)$ is given by

$$\sigma^2 = E[(x(n) - \mu)^2] = E[x^2(n)] - \mu^2. \tag{4}$$

The variance σ^2 is also referred to as the signal power.

The *lag-i temporal correlation* $\rho(i)$ of $x(n)$ is defined as

$$\rho(i) = \frac{E[(x(n) - \mu)(x(n-i) - \mu)]}{E[(x(n) - \mu)^2]} = \frac{E[x(n)x(n-i)] - \mu^2}{\sigma^2}. \tag{5}$$

In this paper, we will be interested mainly in $\rho(1)$ and therefore we will denote it via the simplified notation ρ.

We now consider the i^{th} bit b_i of a word-level signal $x(n)$ defined in (2). Let p_i be the probability that $b_i(n)$ is 1, i.e., $p_i = \Pr(b_i(n) = 1) = E[b_i(n)]$. If $\mathcal{X}_i$ is the set of all elements in $\mathcal{X}$ such that the i^{th} bit is 1, then,

$$\begin{aligned} p_i &= \Pr(x(n) \in \mathcal{X}_i) && (6)\\ &= \sum_{\forall j \in \mathcal{X}_i} \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(j-\mu)^2}{2\sigma^2}} (\text{assuming normal distribution}) && (7) \end{aligned}$$

Clearly, the value of p_i is dependent on the statistical distribution of the values in $\mathcal{X}$. While we have provided an example of a normal distribution here, there is no restriction on the distribution itself. Note that, the probability distribution of $x(n)$ can either be estimated or obtained from the knowledge of the parameters of the signal generation models to be discussed in subsection II.B. However, without loss of generality, we will assume that the probability distribution of $x(n)$ is known *a priori.*

The *temporal correlation*, ρ_i, of the i^{th} bit is defined as,

$$\rho_i = \frac{E[(b_i(n) - p_i)(b_i(n-1) - p_i)]}{E[(b_i(n) - p_i)^2]} = \frac{E[b_i(n)b_i(n-1)] - p_i^2}{p_i - p_i^2}. \tag{8}$$

If $p_i = 1$ or $p_i = 0$ then ρ_i is defined to be 1.

The *transition activity* (or transition probability [20]), t_i, of the i^{th} bit is defined as

$$t_i = \Pr(b_i(n) = 0 \text{ and } b_i(n-1) = 1) + \Pr(b_i(n) = 1 \text{ and } b_i(n-1) = 0). \tag{9}$$

If the bits $b_i(n)$ and $b_i(n-1)$ are independent then the transition activity is given by [20],

$$t_i = 2p_i(1 - p_i). \tag{10}$$

In section III, we will derive an equation relating the transition activity t_i and the correlation ρ_i.

Finally, we define the word-level transition activity, T, as follows,

$$T \;=\; \sum_{i=0}^{B-1} t_i. \tag{11}$$

In section III, we will show how to compute t_i and then employ (11) to compute T.

B. Signal Generation Models

As mentioned in the previous section, we will employ $ARMA$ signal generation models to calculate transition activity. These signal models are commonly employed to represent stationary signals in general and have found widespread application in speech [2] and video coding [18]. Furthermore, signals obtained from sources such as speech, audio, and video can also be modeled employing $ARMA$ models.

An (N,M) order auto-regressive moving average model ($ARMA(N,M)$) can be represented as

$$x(n) \;=\; \sum_{i=0}^{N} d_i\gamma(n-i) + \sum_{i=1}^{M} a_i x(n-i) \tag{12}$$

where the signal $\gamma(n)$ is a white (uncorrelated) noise source with zero mean, and $x(n)$ is the signal being generated. If a given signal source, such as speech, needs to be modeled via (12), then we can choose coefficients a_i and d_i to minimize a certain error measure (such as the mean-squared error) between $x(n)$ and the given source. In that case, we say that $x(n)$ represents the given signal source. As mentioned in the previous subsection, if the a_i's and d_i's in (12) are known, along with the distribution of $\gamma(n)$, then we can obtain the probability distribution of $x(n)$.

The model in (12) is an infinite-impulse response (IIR) filter with coefficients a_i and d_i, with a zero mean white noise as the input. It is also possible to transform this IIR model into one that depends only on the inputs as shown below,

$$x(n) \;=\; \sum_{i=0}^{\infty} h_i\gamma(n-i), \tag{13}$$

where h_i can be computed according to the following recursion,

$$h_k \;=\; d_k + \sum_{i=1}^{N} a_i h_{k-i}, \tag{14}$$

where $h_k = 0$ for $k < 0$, and $h_0 = d_0$. Finally, AR and MA models are special cases of $ARMA$ models. An M^{th} order auto-regressive ($AR(M)$) signal model is identical to an $ARMA(0,M)$ model. Also, an N^{th} order moving-average ($MA(N)$) signal model is the same as an $ARMA(N,0)$ model.

In proving Theorem 1 in section III, we will also employ the following result from [14],

Lemma 1: $\mathrm{E}[b_i(n)b_i(n-1)] = p_i - \frac{t_i}{2}$

III. WORD-LEVEL SIGNAL TRANSITION ACTIVITY

In this section, we will present techniques for estimating word-level transition activity, T, of a signal, $x(n)$, from its word-level statistics. We will first present a theorem relating bit-level quantities, namely, the transition activity t_i, the probability p_i, and temporal correlation, ρ_i. Next, two techniques for estimating ρ_i are presented. The first is referred to as the *exact method*, whereby ρ_i is explicitly determined for the B bits $i = 0, \ldots, B-1$ in $x(n)$. The second method is called the *approximate method* in which break-points BP_0 and BP_1 (as defined in [10]) are determined from an $ARMA$ model of the signal. Simulation results will be provided in support of the theory.

A. Transition Activity For Single-bit Signals

For single-bit signals, we have an expression given by (10) [20] for independent bits $b_i(n)$ and $b_i(n-1)$. In this subsection, we will present a more general result which is also applicable when the temporal correlation between $b_i(n)$ and $b_i(n-1)$ (i.e., ρ_i) is not zero. This result is presented as Theorem 1 as follows,

Theorem 1: *If an i^{th} bit b_i has a probability p_i of being a 1 and has a temporal correlation of ρ_i, then its transition activity t_i is given by*

$$t_i = 2p_i(1-p_i)(1-\rho_i) \tag{15}$$

Proof: From the definition of ρ_i in (8), we have

$$\rho_i = \frac{E[b_i(n)b_i(n-1)] - p_i^2}{p_i - p_i^2}. \tag{16}$$

Substituting for $E[b_i(n)b_i(n-1)]$ from Lemma 1 into (16) and solving for t_i, we get

$$t_i = 2p_i(1-p_i)(1-\rho_i), \tag{17}$$

which is the desired result. ¶

Note that, substitution of $\rho_i = 0$ (corresponding to the case of uncorrelated bits) in (15) reduces it to (10). In subsequent sections, we present two methods (the exact and approximate methods) for calculating ρ_i from word-level statistics. These will then be substituted in (15) to obtain t_i.

B. Estimation of ρ_i: The Exact Method

From (8), we see that it is necessary to compute p_i and $E[b_i(n)b_i(n-1)]$ in order to estimate ρ_i. As p_i can be obtained from the probability distribution function of $x(n)$, we will now focus upon $E[b_i(n)b_i(n-1)]$, which is given by (Recall that $\mathcal{X}_i$ is the set of all elements in $\mathcal{X}$ such that the i^{th}

bit is a '1'),

$$E[b_i(n)b_i(n-1)] = \Pr((b_i(n)=1) \text{ and } (b_i(n-1)=1)) \\ = \Pr(x(n) \in \mathcal{X}_i \text{ and } x(n-1) \in \mathcal{X}_i) \quad (18)$$

In particular, we will employ $AR(1)$ and $MA(N)$ signal models to estimate $E[b_i(n)b_i(n-1)]$. First, we present the following result for an $AR(1)$ model.

<u>Theorem</u> 2: *For an $AR(1)$ signal,*

$$E[b_i(n)b_i(n-1)] = \sum_{\forall j \in \mathcal{X}_i} \Pr(x(n-1)=j) \sum_{\forall k \in \mathcal{X}_i} \Pr(\gamma(n) = k - a_1 j) \quad (19)$$

Proof: From the definition of $E[b_i(n)b_i(n-1)]$ in (18), we have

$$E[b_i(n)b_i(n-1)] = \sum_{(\forall j \in \mathcal{X}_i)} \sum_{(\forall k \in \mathcal{X}_i)} \Pr(x(n)=k \text{ and } x(n-1)=j). \quad (20)$$

Substituting the expression for an $AR(1)$ model (obtained by substituting $N=0$, $M=1$ and $b_0=1$) in (12)) into (20), we obtain

$$\begin{aligned} E[b_i(n)b_i(n-1)] &= \sum_{(\forall j \in \mathcal{X}_i)} \sum_{(\forall k \in \mathcal{X}_i)} \Pr(\gamma(n) + a_1 x(n-1) = k \text{ and } x(n-1)=j) \\ &= \sum_{(\forall j \in \mathcal{X}_i)} \sum_{(\forall k \in \mathcal{X}_i)} \Pr(\gamma(n) + a_1 j = k \text{ and } x(n-1)=j) \\ &= \sum_{(\forall j \in \mathcal{X}_i)} \sum_{(\forall k \in \mathcal{X}_i)} \Pr(\gamma(n) + a_1 j = k) \Pr(x(n-1)=j) \quad (21) \end{aligned}$$

where the last step is justified because $\gamma(n)$ and $x(n-1)$ are independent. Note that, (19) can now be obtained by a simple rewriting of (21). Furthermore, each of the summations in (19) can be evaluated via the knowledge of the probability distribution function. ¶

In order to confirm Theorem 2, we compared the measured values of t_i and ρ_i for the data generated by an $AR(1)$ signal, SIG2, in Table VI, with the estimated values predicted by the theorem. The results shown in Figure 1 indicate the measured and theoretical values match very well. For the word-level transition activity, T, a total error of less than 1% was obtained. Similar results were obtained for the other signals in Table VI. The signals in Table VI were chosen because they represent a wide variety of signals. The signals SIG1 and SIG2 are based on an $AR(1)$ model with positive and negative correlations, respectively, whereas the signal SIG2 has an $AR(1)$ model with positive correlation. Similarly, the signal SIG3 is based on an $MA(1)$ model and the signal SIG4 is identical to SIG2 except for the mean. The signal SIG5 is derived from an $ARMA(3,5)$ model.

We now consider an $MA(1)$ process and present the following result.

Theorem 3: *Let $j, k, l \in \mathcal{X}_i$, where $j + b_1 k \in \mathcal{X}_i$ and $k + b_1 l \in \mathcal{X}_i$. Then, for an $MA(1)$ signal $x(n) = \gamma(n) + b_1\gamma(n-1)$,*

$$E[b_i(n)b_i(n-1)] = \sum_j \sum_k \sum_l \Pr(\gamma(n) = j) \Pr(\gamma(n-1) = k) \Pr(\gamma(n-2) = l) \qquad (22)$$

Proof: Employing the expression for an $MA(1)$ signal obtained by substituting $N = 1$ and $M = 0$ into (12), we get

$$\begin{aligned} E[b_i(n)b_i(n-1)] &= \Pr(\gamma(n), \gamma(n-1), \text{ and } \gamma(n-2) : x(n) \in \mathcal{X}_i \text{ and } x(n-1) \in \mathcal{X}_i) \\ &= \Pr(\gamma(n), \gamma(n-1), \text{ and } \gamma(n-2) : \gamma(n) + b_1\gamma(n-1) \in \mathcal{X}_i \text{ and} \\ &\quad \gamma(n-1) + b_1\gamma(n-2) \in \mathcal{X}_i) \qquad (23) \end{aligned}$$

If $\gamma(n) = j$, $\gamma(n-1) = k$, and $\gamma(n-2) = l$, then we can write (23) as follows,

$$\begin{aligned} E[b_i(n)b_i(n-1)] &= \Pr(\gamma(n) = j \text{ and } \gamma(n-1) = k \text{ and } \gamma(n-2) = l : j + b_1 k \in \mathcal{X}_i \text{ and } k + b_1 l \in \mathcal{X}_i) \\ &= \sum_j \sum_k \sum_l \Pr(\gamma(n) = j) \Pr(\gamma(n-1) = k) \Pr(\gamma(n-2) = l), \qquad (24) \end{aligned}$$

where $j + b_1 k \in \mathcal{X}_i$ and $k + b_1 l \in \mathcal{X}_i$, which is the desired result. ¶

In Figure 2, we show the simulation results in support of Theorem 3. Again, we compared the measured values for t_i and ρ_i in data generated by the $MA(1)$ signal, SIG3, in Table VI with the values predicted by the theorem. In this case, we found that the errors between the measured and predicted values of T were less than 2%.

Finally, we consider the computation of $E[b_i(n)b_i(n-1)]$ for an $MA(2)$ signal and show that Theorem 3 can also be extended to calculate $E[b_i(n)b_i(n-1)]$ for an $MA(N)$ signal. For an $MA(2)$ signal $x(n) = \gamma(n) + b_1\gamma(n-1) + b_2\gamma(n-2)$, the quantity $E[b_i(n)b_i(n-1)]$ is given by,

$$E[b_i(n)b_i(n-1)] = \sum_j \sum_k \sum_l \sum_m \Pr(\gamma(n) = j) \Pr(\gamma(n-1) = k) \Pr(\gamma(n-2) = l) \Pr(\gamma(n-3) = m),$$

where $j, k, l, m : j + b_1 k + b_2 l \in \mathcal{X}_i$ and $k + b_1 l + b_2 m \in \mathcal{X}_i$. It can be checked that $E[b_i(n)b_i(n-1)]$ for $AR(M)$ and $ARMA(N, M)$ signals is difficult to calculate for $M > 1$ because we need to compute the joint probability distribution function of $x(n)$ and $x(n-1)$. However, we can estimate $E[b_i(n)b_i(n-1)]$ for an $AR(M)$ or an $ARMA(N, M)$ signal by approximating the signal with an $MA(N')$ signal, where N' is sufficiently large, or approximating with an $AR(1)$ signal.

C. Estimation of ρ_i: The Approximate Method

In the previous subsection, an exact method for computing ρ_i $(i = 0, \ldots, B-1)$ was presented. For large values of B, this computation can become expensive. In order to alleviate this problem, we will present a computationally efficient method to estimate ρ_i from word-level statistics. As mentioned before, this method (referred to as the *approximate method*) uses a model similar to that described in [10].

In Figure 3, we plot the temporal correlation ρ_i versus bit position i for various audio, video, and communications channel streams described in Table VI. It can be seen that the temporal correlation ρ_i is approximately zero for the LSBs and close to the word-level temporal correlation ρ for the MSBs. Furthermore, there is a region in between the LSBs and MSBs where the bit-level temporal correlation ρ_i increases approximately linearly. As proposed in [10], we divide the bits in the signal word into three regions of contiguous bits referred to as the LSB, *linear*, and MSB regions. The break-points BP_0 and BP_1 separate the LSB from the linear region and the linear from the MSB region, respectively. Furthermore, the graph of temporal correlation ρ_i versus bit position i for the LSB, linear, and MSB regions has slopes of zero, non-zero, and zero, respectively.

In spite of this similarity with [10], the proposed approach differs from [10] in the following ways: 1.) the word is divided into 3 regions based upon the correlation and not the transition activity, 2.) the way the break-points BP_0 and BP_1 are computed, and 3.) our use of (15) to compute t_i and (11) to compute T *analytically*. In particular, we do not employ simulations to estimate transition activity of the most significant bits.

Without loss of generality, we will assume that two's complement representation is employed. By definition, $\rho_i = 0$ for $i < BP_0$. Now, let $\rho_i = \rho_{BP_1}$ for $i \geq BP_1 - 1$.

Hence, we can make the following approximation for two's complement representation,

$$\rho_i = \begin{cases} 0 & (i < BP_0) \\ \frac{(i-BP_0+1)\rho_{BP_1}}{BP_1-BP_0} & (BP_0 \leq i < BP_1 - 1) \\ \rho_{BP_1} & (i \geq BP_1 - 1) \end{cases} \tag{25}$$

We now examine the relation between the parameters in the set $\{\rho_{BP_1}, BP_0, BP_1\}$ and those in $\{\mu, \sigma, \rho\}$ in order to derive expressions for ρ_{BP_1}, BP_0, and BP_1.

C.1 Calculation of BP_0

For an uncorrelated signal, $\gamma(n)$, a good estimate of BP_0 is given by $\log_2 \sigma_\gamma$, where σ_γ is the standard deviation of $\gamma(n)$ [10]. If the signal $x(n)$ has non-zero correlation, then it can be modeled using a signal model, which can then be used to calculate BP_0. For instance, if $x(n)$ is modeled using an $ARMA$ model then it can be expressed using (13). Since the signals $h_i\gamma(n-i)$ are uncorrelated, BP_0 for each of the signals can be estimated as $\log_2 |h_i|\sigma_\gamma$. Given an adder which accepts two input signals with BP_0 break-points, BP_{01} and BP_{02} respectively, a good estimate for the BP_0 break-point at the output of the adder is $\max(BP_{01}, BP_{02})$. Hence, the break-point BP_0 for a signal $x(n) = \sum_i h_i \gamma(n-i)$ can now be estimated as the maximum of the BP_0's of the signals $h_i \gamma(n-i)$, as shown below,

$$BP_0 = [\log_2 h_{max} \sigma_\gamma], \tag{26}$$

where $h_{max} = \max(|h_i|)$ and $[k]$ is the integer nearest to k. We verified (26) by comparing the measured and estimated values of BP_0 obtained from data generated with the five signals shown in Table VI. The measured value of BP_0 was obtained by counting the number of bits with correlation close to 0. For instance, from Figure 4, we see that BP_0 for the signal SIG2 is 8 because there are 8 bits with correlation close to 0. The measured and estimated values of BP_0 are shown in Table VI, where it can be seen that the measured and estimated values match quite well.

C.2 Calculation of BP_1

Let the values of $x(n)$ lie between the values x_{min} and x_{max}. In a normal distribution, $x_{min} = \mu - 3\sigma$ and $x_{max} = \mu + 3\sigma$. We define BP_1 such that for $i \geq BP_1 - 1$, ρ_i is approximately constant. Since the dynamic range of $x(n)$ is $x_{max} - x_{min}$, the least significant $\log_2(x_{max} - x_{min})$ bits are required to cover this range. Hence, we have

$$BP_1 = [\log_2(x_{max} - x_{min})],$$

which reduces to,

$$BP_1 = [\log_2 6\sigma] \tag{27}$$

for a normal distribution where σ is the standard deviation of $x(n)$. The estimate for BP_1 in (27) is different from that in [10], which is given in (28) below for comparison purposes,

$$BP_1 = [\log_2(|\mu| + 3\sigma)]. \tag{28}$$

When $|\mu| \leq 3\sigma$, both (27) and (28) are approximately equal with the maximum difference of 1 occuring at $\mu = 0$. However, in the case where $|\mu| \gg 3\sigma$, (27) is more accurate than (28). This is due to the fact that for $|\mu| > 3\sigma$ there are 3 regions in which ρ_i is a constant. The first region consists of the bit positions i such that $i < BP_0$. The second region has bit positions i lying between BP_1 and another break-point BP_2. The third region consists of bits with positions beyond BP_2 where the bits do not have any transitions. The bits in the third region can be calculated by computing the common most significant bits in the binary representations of the numbers x_{max} and x_{min}. These are the numbers which lie at the two extremes of the probability distribution.

We verified (27) by comparing it with the measured values of BP_1 obtained from data generated by various signals in Table VI. The results are shown in Table VI where it can be seen that the measured and estimated values match closely. To verify that BP_1 is independent of the mean μ, we plot the bit-level temporal correlation ρ_i and transition activity t_i for signals SIG2 and SIG4 in Figure 5. Note that from Table VI, SIG2 and SIG4 are identical except for their mean μ. It can be seen from Figure 5, that the value of BP_1, 13, for SIG2 and SIG4 is independent of μ, which is also indicated by (27). For SIG4 BP_2 is 15 because the binary representations of x_{max}, 19384, and x_{min}, 13384, have only 1 common most significant bit.

All that now remains in the approximate method is to estimate the value for ρ_{BP_1}. If the model for $x(n)$ is known then we can use the exact method to calculate ρ_{BP_1}. If the model for $x(n)$ is not available, then we assume that $\rho_{BP_1} = \rho$ which is the word-level temporal correlation. This is because in most number representations like sign magnitude, two's complement, and one's complement, the most significant bits have higher weight than the least significant bits. Hence the correlation of the most significant bits will be close to the word-level correlation. This is especially valid for audio and video signals (see Figure 3).

D. Calculation of T

Employing (11), (15), (26), (27), we computed the value of the word-level transition activity T for the signals described in Table VI for two's complement representation. The measured and estimated word-level transition activity T, for all the signals are shown in Table VI. It can be noted that the error is less than 2% for two's complement representation.

E. Effect of Signal Encoding/Number Representation

The results presented so far in this section (Theorems 2 and 3) have implicitly included the effect of the signal encoding. This is due to the fact that the elements of the sets $\mathcal{X}$ and $\mathcal{X}_i$ will depend upon the signal encoding. In this subsection, we examine explicitly the effect of number representation on the transition activity.

In the previous subsections, we have considered two's complement number representation. The unsigned representation will have the same transition activity as two's complement because the most significant bits of the former behave identical to the sign bits of the latter. Therefore, we will not consider the unsigned representation any further. We will now analyze the one's complement and sign-magnitude representations.

E.1 One's complement

The one's complement representation is identical to the two's complement for positive numbers. For negative numbers, we can generate the two's complement representation from that of the one's complement by adding a 1 to the LSB, which will usually affect only the LSBs. In the approximate method, since we assume that LSBs are uncorrelated, the activity of the LSBs in the one's complement will be close to that of the two's complement. The remaining bits will have the same temporal correlation as in the two's complement representation. Therefore, ρ_i for one's complement representation will be the same as that for two's complement representation. The measured and estimated word level transition activity T for the signals in Table VI employing one's complement are shown in the second set of the three columns in Table VI. The measured word-level transition activity was obtained by generating data using the signal model and measuring transition activity in that data. The error in T is less than 2% for one's complement representation.

E.2 Sign magnitude

In the sign magnitude representation there is only one sign bit; namely, the most significant bit, $b_{B-1}(n)$. This bit will have the same temporal correlation as the sign bits in two's complement representation because the temporal correlation of the sign bit depends on the sign transitions. The bits $b_i(n)$ for $i < BP_0$ are uncorrelated as in the case of two's complement. We again assume a linear model for ρ_i for $BP_0 \leq i < BP_1 - 1$. The resulting expression for ρ_i is as follows,

$$\rho_i = \begin{cases} 0 & (i < BP_0) \\ \frac{(i-BP_0+1)\rho_{BP_1}}{BP_1-BP_0} & (BP_0 \leq i < BP_1 - 1) \\ 1 & (BP_1 - 1 \leq i < B - 1) \\ \rho_{BP_1} & (i = B - 1) \end{cases} \tag{29}$$

The measured and estimated word level transition activity, T, for the signals are shown in the last three columns of Table VI. As always, the measured word-level transition activity was obtained by generating data using the signal model and measuring transition activity in that data. It can be seen that the error in T is less than 2% for all the signals except for SIG4 where the error is less than 5%.

E.3 Discussion

From the expressions for ρ_i in (25) and (29) we see that the temporal correlation and hence transition activity for unsigned, one's complement, and two's complement representations are nearly equal. Also, the transition activity for sign magnitude is less than or equal to two's complement because the number of sign bits in sign magnitude representation (one) is less than or equal to the number of sign bits in two's complement representation. These conclusions are supported via the results in Table VI, which show that the transition activity for unsigned, one's complement and two's complement are similar, while the transition activity for sign magnitude is less than that of unsigned, one's complement, and two's complement.

IV. TRANSITION ACTIVITY FOR DSP ARCHITECTURES

In the previous section, we had presented techniques for estimating the word-level transition activity T for signals. In this section, we will apply these techniques to compute the transition activity for DSP architectures. First, we propagate the statistics of the input signal through a given DSP architecture so that word-level statistics for each signal in the architecture is obtained. Then, we calculate the transition activity for each signal employing the techniques presented in the previous section. These are then added up to obtain the total transition activity of the architecture.

A. Propagation of Word-Level Statistics

In this subsection, we propagate the input statistics to the output for the following DSP operators:

1. Adder
2. Multiplier
3. Multiplexor
4. Delay

These operators were chosen due to their widespread use in DSP algorithms. First, we start with the adder.

A.1 Adder

In Figure 6, the two signals $x_i(n)$ $(i = 1, 2)$ at the input to the adder have statistics μ_i, σ_i, ρ_i $(i = 1, 2)$. The mean μ_3, variance σ_3^2, and temporal correlation, ρ_3, at the output of the adder are given by the following equations.

$$
\begin{aligned}
\mu_3 &= E[x_3(n)] = E[x_1(n) + x_2(n)] = \mu_1 + \mu_2 \\
\sigma_3^2 &= E[x_3^2(n)] - \mu_3^2 \\
&= \sigma_1^2 + \sigma_2^2 + 2E[x_1(n)x_2(n)] - 2\mu_1\mu_2 \\
\rho_3 &= \frac{E[x_3(n)x_3(n-1)] - \mu_3^2}{\sigma_3^2} = \frac{E[(x_1(n)+x_2(n))(x_1(n-1)+x_2(n-1))] - (\mu_1+\mu_2)^2}{\sigma_3^2} \\
&= \frac{\rho_1\sigma_1^2 + \rho_2\sigma_2^2 + E[x_2(n)x_1(n-1)] + E[x_1(n)x_2(n-1)] - 2\mu_1\mu_2}{\sigma_3^2}
\end{aligned}
$$

If $x_1(n) = \sum_{i=0}^{k-1} c_i x(n-i)$ and $x_2(n) = c_k x(n-k)$ as in the case of an FIR filter, we have

$$\mu_3 = \mu(\sum_{i=0}^{k} c_i) \tag{30}$$

$$\sigma_3^2 = \sigma^2(\sum_{i=0}^{k} c_i^2 + 2\sum_{i=0}^{k-1}\sum_{j=i+1}^{k} \rho(j-i)c_i c_j) \tag{31}$$

$$\rho_3 = \frac{\sigma^2(\sum_{i=0}^{k-1} c_i c_{i+1} + \sum_{i=0}^{k}\sum_{j=i}^{k} c_i c_j \rho(j-i+1) + \sum_{i=0}^{k-2}\sum_{j=i+2}^{k} c_i c_j \rho(j-i-1))}{\sigma_3^2} \tag{32}$$

A.2 Multiplier

In this subsection we examine how to propagate word-level statistics through a multiplier. In Figure 6, the two signals $x_1(n)$ and $x_2(n)$ at the input to the multiplier have statistics μ_1, σ_1, ρ_1 and μ_2, σ_2, ρ_2 respectively. The statistics at the output of the multiplier are given by the following equations,

$$\begin{aligned}
\mu_3 &= \mathrm{E}[x_3(n)] = \mathrm{E}[x_1(n)x_2(n)] \\
\sigma_3^2 &= \mathrm{E}[x_3^2(n)] - \mu_3^2 = \mathrm{E}[(x_1(n)x_2(n))(x_1(n)x_2(n))] - \mathrm{E}^2[x_1(n)x_2(n)] \\
&= \mathrm{E}[x_1^2(n)x_2^2(n)] - \mathrm{E}^2[x_1(n)x_2(n)] \\
\rho_3 &= \frac{E[x_3(n)x_3(n-1)] - \mu_3^2}{\sigma_3^2} = \frac{E[x_1(n)x_2(n)x_1(n-1)x_2(n-1)] - E^2[x_1(n)x_2(n)]}{\sigma_3^2}
\end{aligned}$$

If $x_2(n)$ is a constant c_1, then $\mu_3 = c_1\mu_1$, $\sigma_3 = c_1\sigma_1$, and $\rho_3 = \rho_1$.

A.3 Multiplexor

When two signals, $x_1(n)$ and $x_2(n)$ with statistics $\{\mu_1, \rho_1, \sigma_1\}$ and $\{\mu_2, \rho_2, \sigma_2\}$, respectively, are multiplexed (Figure 6) by a control signal with probability p_c and correlation ρ_c, then the statistics $\{\mu_3, \rho_3, \sigma_3\}$ of $x_3(n)$ at the output of the multiplexor are given by (assuming 0 and 1 on the control signal selects $x_1(n)$ and $x_2(n)$ respectively),

$$\mu_3 = E[x_3(n)] = (1-p_c)\mu_1 + p_c\mu_2 \tag{33}$$

$$\begin{aligned}
\sigma_3^2 &= E[x_3^2(n)] - \mu_3^2 = E[(1-p_c)x_1^2(n) + p_c x_2^2(n)] - (1-p_c)^2\mu_1^2 - p_c^2\mu_2^2 - 2p_c(1-p_c)\mu_1\mu_2 \\
&= (1-p_c)\sigma_1^2 + p_c(1-p_c)\mu_1^2 + p_c\sigma_2^2 + p_c(1-p_c)\mu_2^2 - 2p_c(1-p_c)\mu_1\mu_2 \qquad (34)
\end{aligned}$$

$$\rho_3 = \frac{E[x_3(n)x_3(n-1)] - \mu_3^2}{\sigma_3^2}, \tag{35}$$

where $E[x_3(n)x_3(n-1)]$ is given by,

$$\begin{aligned}
E[x_3(n)x_3(n-1)] = {} & (1-p_c)(1-p_c+p_c\rho_c)E[x_1(n-1)x_1(n)] + p_c(1-p_c)(1-\rho_c)E[x_1(n-1)x_2(n)] + \\
& p_c(1-p_c)(1-\rho_c)E[x_2(n-1)x_1(n)] + p_c(p_c - p_c\rho_c + \rho_c)E[x_2(n-1)x_1(n)],
\end{aligned}$$

where the expectations in the above formula can be obtained from the auto-correlation and cross-correlation values of the input signals. Also, BP_0 for $x_3(n)$ is the maximum of BP_0 for $x_1(n)$ and $x_2(n)$.

A.4 Delay

A delay shifts the signal by one time unit, which in this case is a clock period. The statistics at the output of a delay element are identical to that at the input.

B. Example 1: FIR filter

We illustrate propagating word-level statistics using the 5-tap Finite Impulse Response (FIR) filter in Figure 7, where coefficients $c_1 = c_5 = 0.09765625$, $c_2 = c_4 = 0.1953125$, and $c_3 = 0.39453125$. The correlations ρ_{10}, ρ_{11}, ρ_{12}, and ρ_{13} require the lag-2, lag-3, lag-4, and lag-5 correlations of the input be known. If they are not available, then for most real-life signals, the lag-i correlation can be approximated by $\rho^i(1)$. Such an approximation corresponds to approximating the signal with an $AR(1)$ model. The statistics of signals within the filter can be calculated using (30), (31), and (32). As an example, the equations for the mean, variance, and temporal correlation of the output, $x_{13}(n)$ are given below,

$$
\begin{aligned}
\mu_{13} &= \mu(\sum_{i=1}^{5} c_i) \\
\sigma_{13}^2 &= \sigma^2(\sum_{i=1}^{5} c_i^2 + 2\sum_{i=1}^{4}\sum_{j=i+1}^{5} \rho(j-i)c_i c_j) \\
\rho_{13} &= \frac{\sigma^2(\sum_{i=1}^{4} c_i c_{i+1} + \sum_{i=1}^{5}\sum_{j=i}^{5} c_i c_j \rho(j-i+1) + \sum_{i=1}^{3}\sum_{j=i+2}^{5} c_i c_j \rho(j-i-1))}{\sigma_{13}^2}
\end{aligned}
$$

The measured and estimated word-level statistics for video3 data are shown in Table VI. We see that the estimated statistics match the measured statistics very closely, with errors of less than 1%. Table VI shows the measured and estimated total word-level transition activity for the FIR filter (when the signals from Table VI are passed through the filter) in Figure 7 and its transpose in Figure 8. The measured values were obtained by simulation using a C program. It can be seen that the total transition activity for the transpose form is always less than that for the direct form because of the lower transition activity at the inputs to the delays. The lower transition activity at the inputs to the delays is because multiplying by a constant of magnitude less than 1 reduces the variance and hence the transition activity.

C. Example 2: Folded FIR filter

Folding [26] is an algorithm transformation technique that allows the mapping of algorithmic operations to a given set of hardware units. For instance, the 5 tap FIR filter in Figure 7 containing 5 multiplies and 4 adds can be folded onto 3 multipliers and 2 adders using additional delays and multiplexors as shown in Figure 9.

The statistics of the signals of the unfolded filter can be calculated using (30), (31), and (32). These are used along with (33), (34), and (35) to calculate the statistics of signals of the folded filter. As an example, the statistics of the signal, $x_{11,7}(n)$, obtained by multiplexing $x_{11}(n)$ and $x_7(n)$ are given by the following equations,

$$\begin{aligned}
\mu_{11,7} &= \frac{(c_1+c_2+2c_3)\mu}{2} \\
\sigma^2_{11,7} &= 2c_3^2\sigma^2 + 2(c_1^2+c_2^2+c_3^2+2c_1c_2\rho+2c_2c_3\rho+2c_1c_3\rho(2))\sigma^2 + c_3^2\mu^2 + (c_1+c_2+c_3)^2\mu^2 - 2(c_1c_3+c_2c_3+c_3^2)\mu^2 \\
&= 2c_3^2\sigma^2 + 2(c_1^2+c_2^2+c_3^2+2c_1c_2\rho+2c_2c_3\rho+2c_1c_3\rho(2))\sigma^2 + (c_1+c_2)^2\mu^2 \\
\rho_{11,7} &= \frac{2c_1c_3E[x(n)x(n-2)+x(n)x(n-1)]+2c_2c_3E[x(n-1)x(n-2)+x^2(n-1)]+2c_3^2E[x^2(n-2)+x(n-1)x(n-2)]-(c_1+c_2+2c_3)^2\mu^2}{\sigma^2_{11,7}} \\
&= \frac{2\sigma^2c_3(c_1(\rho(2)+\rho)+(c_2+c_3)(\rho+1))-(c_1+c_2)^2\mu^2}{\sigma^2_{11,7}}
\end{aligned}$$

The measured and estimated word-level statistics are shown in Table VI. The measured and estimated word-level statistics match very closely, with errors of less than 1%. Table VI shows the measured and estimated total word-level transition activity for the folded FIR filter in Figure 9. The error between the measured and estimated transition activity for the five signals is less than 4%. A comparison between the transition activities of the original FIR filter (see Table VI) and the folded architecture (see Table VI) indicates that folding increases the number of transitions. This conclusion is consistent with that observed in [6].

D. Example 3: IIR filter

In this example we propagate word-level statistics through the simple Infinite Impulse Response (IIR) filter in Figure 10, where $c_1 = 0.1$.

The equations for the statistics of the signals in the direct form IIR filter are given by,

$$x_3(n) = \sum_{i=1}^{n} c_1^i x_0(n-i)$$

$$\begin{aligned}
\mathrm{E}[x_0(n)x_3(n)] &= \mathrm{E}[\textstyle\sum_{i=1}^{n} c_1^i x_0(n-i)x_0(n)] \\
&= \lim_{n\to\infty} \textstyle\sum_{i=1}^{n} c_1^i \mathrm{E}[x_0(n-i)x_0(n)] = \lim_{n\to\infty} \sum_{i=1}^{n}(c_1^i\rho(i)\sigma_0^2 + c_1^i\mu_0^2)
\end{aligned}$$

$= \frac{c_1\mu_0^2}{1-c_1} + \frac{\sigma_0^2 c_1 \rho(1)}{1-c_1\rho(1)}$ (assuming $\rho(i) = \rho^i(1)$)

$\mathrm{E}[x_0(n-1)x_3(n)] = \frac{c_1\mu_0^2}{1-c_1} + \sigma_0^2 \sum_{i=1}^{\infty} c_1^i \rho(i-1) = \frac{c_1\mu_0^2}{1-c_1} + \frac{\sigma_0^2 c_1}{1-c_1\rho(1)}$ (assuming $\rho(i) = \rho^i(1)$)

$\mathrm{E}[x_0(n)x_3(n-1)] = \frac{c_1\mu_0^2}{1-c_1} + \sigma_0^2 \sum_{i=1}^{\infty} c_1^i \rho(i+1) = \frac{c_1\mu_0^2}{1-c_1} + \frac{\sigma_0^2 c_1 \rho(1)^2}{1-c_1\rho(1)}$ (assuming $\rho(i) = \rho^i(1)$)

$\mu_1 = \frac{\mu_0}{1-c_1}$

$\sigma_1^2 = \sigma_0^2 + c_1^2\sigma_1^2 + 2\mathrm{E}[x_0(n)x_3(n)] - 2\mu_0\mu_3 = \frac{\sigma_0^2 + 2E[x_0(n)x_3(n)] - 2\mu_0\mu_3}{1-c_1^2}$

$\rho_1 = \frac{E[x_1(n)x_1(n-1)] - \mu_1^2}{\sigma_1^2} = \frac{\rho(1)\sigma_0^2 + \rho_1 c_1^2 \sigma_1^2 + E[x_3(n)x_0(n-1)] + E[x_0(n)x_3(n-1)] - 2\mu_0 c_1 \mu_1}{\sigma_1^2}$

$= \frac{\rho(1)\sigma_0^2 + E[x_3(n)x_0(n-1)] + E[x_0(n)x_3(n-1)] - 2\mu_0 c_1 \mu_1}{\sigma_1^2(1-c_1^2)}$

The measured and estimated statistics are shown in Table VI. The error between the measured and estimated statistics is less than 1%. Table VI shows the measured and estimated total word-level transition activity for the direct form IIR filter and its transpose in Figure 10. We see that the total transition activity is always less for the transpose form due to the lower transition activity at the input to the latch because multiplication by a constant of magnitude less than 1 reduces the variance which in turn reduces the transition activity.

V. RESULTS WITH REALISTIC BENCHMARK SIGNALS

We have so far presented results using the stationary, synthetic signals in Table VI. In this section, we will present simulation results for the non-stationary, naturally occuring, audio, video and communications channel signals described in Table VI. First, we apply the approximate method (see subsection III(C)) to compare the measured and estimated transition activity for these signals. Then, we process these signals through the direct form FIR (Figure 7) and IIR (Figure 10), transpose FIR (Figure 8) and IIR (Figure 10) and the folded direct form FIR (Figure 9) filters to compute the total transition activity in these structures.

A. Realistic benchmark signals

For the audio, video, and communications channel data described in Table VI, the approximate method was employed to estimate transition activity. The results are shown in Table VI where the measured transition activity was calculated directly from the data. We assumed $\rho_{BP_1} = \rho$, which is the word-level temporal correlation. To estimate BP_0 we assumed $AR(1)$ models for all data sets except Audio5 and Video3. We used $MA(10)$ models for Video3 and Audio5 because the $AR(1)$ models resulted in higher errors. The measured and estimated value of BP_0 is shown in Table VI. The difference in the measured and estimated value of BP_0 for signals Audio5, Audio6, and Audio7 is due to the fact that the least significant bits of these signals are correlated as can be seen from Figure 3.

From Table VI, we see that for unsigned, two's complement and one's complement representations, the estimation error in T is less than 10%. For sign magnitude representation, the error in T is less than 18%.

B. Total word-level transition activity, T, for FIR and IIR filters

In this subsection, we present the measured and estimated transition activity with audio, video, and communications channel data for the direct form filter in Figure 7 and its transpose in Figure 8 (see Table VI), the folded direct form filter in Figure 9 (see Table VI), and the IIR filter and its transpose in Figure 10 (see Table VI). The errors in T for all the filters are less than 12%. Table VI compares the run time for simulation and the run time for the approximate method on a 85 MHz SparcStation 5. We see that in most cases the run time for the approximate method is an order of magnitude less than that for simulation. The run time for simulation depends on the length of the input sequence whereas the run time for the approximate method depends on the width of the signals (8-bit for video3 and 16-bit for the rest). This is because, in our method, the computational complexity is determined by the calculation of p_i using (6) where the summation is over 2^B elements where B is the bit width. We can make the computation time of p_i essentially independent of bit width by calculating the sum over points in $\mathcal{X}_i$ spaced a certain distance ($2^{\frac{BP_0}{2}}$) apart with basically no loss of accuracy of the sum. The running times using the fast approximate method and the Dual Bit Type (DBT) method are also shown in Table VI. The run times for the approximate method can be further reduced by introducing optimizations such as setting the transition activity at the output of a delay to be equal to that at its input, etc.

VI. CONCLUSIONS AND FUTURE WORK

We have proposed a novel methodology to estimate the signal transition activity from the knowledge of the word-level statistics (viz., the mean (μ), variance (σ^2), and temporal correlation (ρ)), the signal generation model (AR, MA, and $ARMA$) and the number representation. Two techniques were presented to estimate the transition activity of the bits comprising the signal word for stationary signals only. However, a possible generalization is to adaptively compute the signal statistics and obtain a more accurate estimate of the signal transition activity. We studied common filter examples to demonstrate the propagation of the word-level statistics of the input to determine the total transition activity in the filter. The methodology presented here provides a basis for high-level power estimation and optimization, whereby the information regarding the signal characteristics along with the topology of the DSP data-flow graph can be exploited. While the present work has focussed upon the problem of high-level power estimation, our current effort is being directed towards automated high-level synthesis of low-power DSP hardware. Incorporation of circuit level parameters into the proposed methodology is also planned for the future.

References

[1] M. Alidina, J. Monterio, S. Devadas, A. Ghosh, and M. Papaefthymiou, "Precomputation-based sequential logic optimization for low-power," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 2(4), pp. 426-436, December 1994.

[2] B. Atal and M. R. Schroeder, "Predictive coding of speech and subjective error criteria", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, no. 23, pp. 247–254, June 1979.

[3] 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 Journal of Solid-State Circuits*, vol. 2, no. 4, pp. 398–407, December 1994.

[4] M. G. Bellanger, "Adaptive Digital Filters and Signal Analysis," Marcel Dekker, 1987.

[5] L. Benini and G. De Micheli, "Automatic synthesis of low-power gated-clock finite-state machines," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 15, no. 6, pp. 630–643, June 1996.

[6] A. Chandrakasan and R. W. Brodersen, "Minimizing power consumption in digital CMOS circuits," *Proceedings of the IEEE*, vol. 83, no. 4, pp. 498–523, April 1995.

[7] A. P. Chandrakasan, M. Potkonjak, R. Mehra, J. Rabaey, and R. W. Broderson, "Optimizing Power Using Transformations," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 14, no. 1, pp. 12–31, January 1995.

[8] B. Davari, R. H. Dennard, and G. G. Shahidi, "CMOS scaling for high-performance and low-power - The next ten years," *Proceedings of the IEEE*, vol. 83, no. 4, pp. 595–606, April 1995.

[9] M. Horowitz, T. Indermaur, and R. Gonzalez, "Low-power digital design," *IEEE Symposium on Low Power Electronics*, pp. 8–11, San Diego, October 1994.

[10] P. E. Landman and J. M. Rabaey, "Architectural Power Analysis: The Dual Bit Type Method," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 3, no. 2, pp. 173–187, June 1995.

[11] P. E. Landman and J. M. Rabaey, "Activity-sensitive architectural power analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 15, no. 6, June 1996.

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

[13] R. Marculescu, D. Marculescu, and M. Pedram "Switching Activity Analysis Considering Spatiotemporal Correlations," *International Conference on Computer-Aided Design*, pp. 294–299, November 1994.

[14] Tan-Li Chou, K. Roy, and S. Prasad, "Estimation of Circuit Activity Considering Signal Correlations and Simultaneous Switching," *International Conference on Computer-Aided Design*, pp. 300–303, November 1994.

[15] A. Ghosh, S. Devadas, K. Keutzer, and J. White, "Estimation of average switching activity in combinational and sequential circuits", *29th Design Automation Conference*, pp. 253–259, June 1992.

[16] S. Gupta and F. Najm, "Power Macromodeling for High Level Power Estimation", *34th Design Automation Conference*, pp. 365–370, June 1997.

[17] C. Huang, B. Zhang, A. Deng, and B. Swirski, "The design and implementation of PowerMill," in *International Symposium on Low Power Design*, Dana Point, CA, pp. 105–110, April 1995.

[18] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice-Hall, Inc. Englewood Cliffs, NJ 1984.

[19] F. Najm, "Transition density, a new measure of activity in digital circuits," *IEEE Transactions on Computer-Aided Design*, vol. 12, no. 2, pp. 310–323, February 1993.

[20] F. Najm, "A survey of power estimation techniques in VLSI circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 2, pp. 446-455, December 1994.

[21] Y. Nakagome, K. Itoh, M. Isoda, K. Takeuchi, and M. Aoki, "Sub-1-V swing internal bus architecture for future low-power ULSI's," *IEEE Journal of Solid-State Circuits*, vol. 28, no. 4, pp. 414–419, April 1993.

[22] M. Nemani and F. Najm, "Towards a High-Level Power Estimation Capability," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 15, no. 6, pp. 588–598, June 1996.

[23] A. V. Oppenheim and R. W. Schafer, "Discrete-time signal processing," Prentice-Hall, New Jersey, 1989.

[24] A. Papoulis, "Probability, Random Variables, and Stochastic Processes," McGraw-Hill, Third Edition, 1991.

[25] K. K. Parhi, "Algorithm transformation techniques for concurrent processors," *Proceedings of the IEEE*, vol. 77, pp. 1879-1895, December 1989.

[26] K. K. Parhi, Ching-Yi Wang, and A. P. Brown, "Synthesis of Control Circuits in Folded Pipelined DSP Architectures," *IEEE Journal of Solid-State Circuits*, vol. 27, no. 1, pp. 181–195, January 1992.

[27] A. Raghunathan, S. Dey, and N. K. Jha, "Glitch Analysis and Reduction in Register Transfer Level Power Optimization," *33rd Design Automation Conference*, pp. 331-336, June 1996.

[28] H. Samueli, "An Improved Search Algorithm for the Design of Multiplierless FIR Filters with Powers-of-Two Coefficients," *IEEE Transactions on Circuits and Systems*, pp. 1044–1047, July 1989.

[29] J. H. Satyanarayana and K. K. Parhi, "HEAT: Hierarchical Energy Analysis Tool," *33rd Design Automation Conference*, pp. 9-14, June 1996.

[30] N. R. Shanbhag, "A fundamental basis for power-reduction in VLSI circuits," *IEEE International Symposium on Circuits and Systems*, vol. 4, pp. 9–12, May 1996.

[31] A. Shen, A. Ghosh, S. Devadas, and K. Keutzer, "On average power dissipation and random pattern testability of CMOS combinational logic networks," *International Conference on Computer-Aided Design*, pp. 402–407, November 1992.

[32] C.-Y. Tsui, M. Pedram, and A. Despain, "Efficient estimation of dynamic power consumption under a real delay model," *International Conference on Computer-Aided Design*, pp. 224–228, November 1993.

[33] E. A. Vittoz, "Low-power design: Ways to approach the limits", *IEEE Solid-State Circuits Conference*, pp. 14–18, 1994.

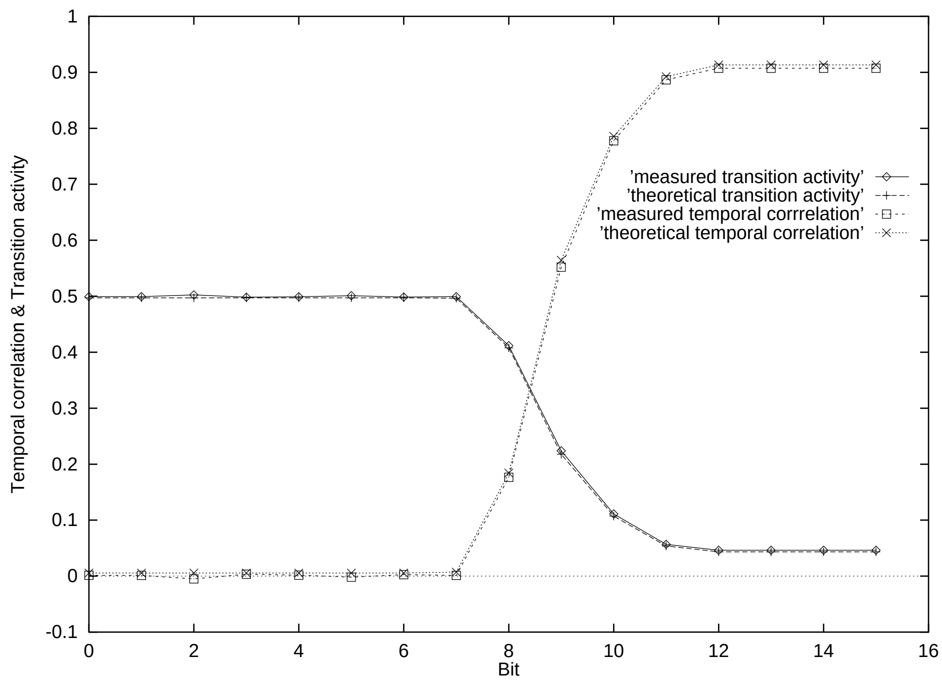
LIST OF FIGURE CAPTIONS

LIST OF TABLE CAPTIONS

ACKNOWLEDGMENT OF FINANCIAL SUPPORT AND AFFILIATION OF AUTHORS

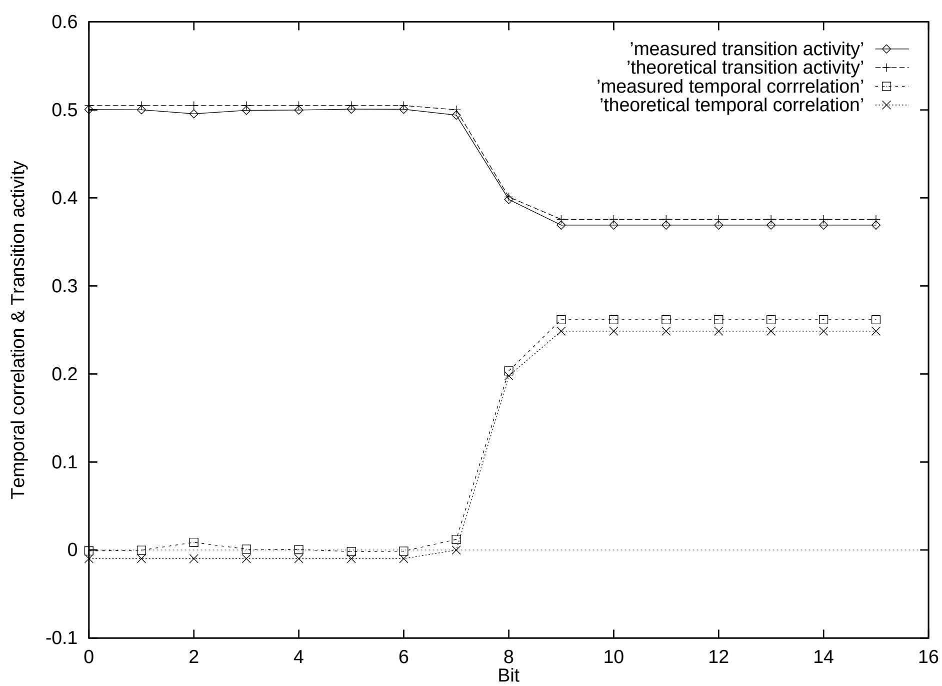
This work was supported by National Science Foundation CAREER award MIP-9623737, the Semiconductor Research Corporation, and by a grant from the University of Illinois Research Board

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



'measured transition activity'
'theoretical transition activity'
'measured temporal corrrelation'
'theoretical temporal correlation'
Temporal correlation & Transition activity
Bit

Fig. 1.



'measured transition activity'
'theoretical transition activity'
'measured temporal corrrelation'
'theoretical temporal correlation'
Temporal correlation & Transition activity
Bit

Fig. 2.

Signal	$x(n)$	σ_γ	σ	ρ	μ
SIG1	$\gamma(n) - 0.5x(n-1)$	866	1000	-0.50	0
SIG2	$\gamma(n) + 0.99x(n-1)$	141	1000	0.99	0
SIG3	$\gamma(n) + 0.5\gamma(n-1)$	100	111	0.40	0
SIG4	$\gamma(n) + 0.99x(n-1)$	141	1000	0.99	16384
SIG5	$\gamma(n) + 0.4\gamma(n-1) + 0.2\gamma(n-2) +$ $.07\gamma(n-3) + .5x(n-1) + .3x(n-2) +$ $0.1x(n-3) + 0.05x(n-4) - .2x(n-5)$	1000	2309	0.89	0

TABLE I

Signal	BP_0		BP_1	
	Measured	Estimated	Measured	Estimated
SIG1	11	10	13	13
SIG2	8	7	13	13
SIG3	8	7	10	9
SIG4	8	7	13	13
SIG5	11	10	14	14

TABLE II

Signal	Unsigned, Two's complement			One's complement			Sign magnitude		
	Measured	Estimated	% Error	Measured	Estimated	% Error	Measured	Estimated	% Error
SIG1	8.79	8.82	0.34	8.79	8.82	0.34	6.07	6.16	1.48
SIG2	4.99	5.03	0.80	4.99	5.03	0.80	4.65	4.74	1.94
SIG3	6.97	6.94	0.43	6.97	6.94	0.43	4.20	4.15	1.19
SIG4	4.99	5.03	0.80	4.99	5.03	0.80	4.65	4.86	4.52
SIG5	6.54	6.42	1.83	6.55	6.42	1.98	5.91	5.89	0.34

TABLE III

Signal	μ			ρ			σ		
	Measured	Estimated	% Error	Measured	Estimated	% Error	Measured	Estimated	% Error
x_0, x_1, x_2, x_3, x_4	99.7108	99.7108	0.00	0.9199	0.9199	0.00	55.5663	55.5663	0.00
x_5, x_9	9.7445	9.7374	0.07	0.9183	0.9199	0.17	5.4648	5.4264	0.71
x_6, x_8	19.4827	19.4748	0.04	0.9198	0.9199	0.01	10.8646	10.8528	0.11
x_7	39.3477	39.3390	0.02	0.9196	0.9199	0.03	21.9415	21.9226	0.09
x_{10}	29.2272	29.2122	0.05	0.9529	0.9534	0.05	16.0293	15.9868	0.27
x_{11}	68.5749	68.5512	0.03	0.9660	0.9661	0.01	37.1125	37.0569	0.15
x_{12}	88.0576	88.0259	0.04	0.9763	0.9764	0.01	47.1728	47.1104	0.13
x_{13}	97.8021	97.7633	0.04	0.9811	0.9812	0.01	51.9925	51.9001	0.18

TABLE IV

Signal	Direct form			Transpose		
	Measured	Estimated	% Error	Measured	Estimated	% Error
SIG1	148.31	148.92	0.13	145.45	145.97	0.36
SIG2	76.64	76.40	0.31	72.84	72.26	0.80
SIG3	113.15	113.64	0.43	109.00	109.44	0.40
SIG4	74.55	74.81	0.35	70.25	70.53	0.40
SIG5	104.63	102.10	2.42	101.41	98.62	2.75

TABLE V

Signal	μ			ρ			σ		
	Measured	Estimated	% Error	Measured	Estimated	% Error	Measured	Estimated	% Error
$x_{0,4}$	99.7108	99.7108	0.00	0.7203	0.7203	0.00	55.5663	55.5663	0.00
$x_{1,3}$	99.7108	99.7108	0.00	0.8150	0.8150	0.00	55.5663	55.5663	0.00
$x_{2,2}$	99.7108	99.7108	0.00	0.9600	0.9600	0.00	55.5663	55.5663	0.00
$x_{5,9}$	9.7444	9.7374	0.07	0.7200	0.7203	0.04	5.4648	5.4264	0.71
$x_{6,8}$	19.4827	19.4748	0.04	0.8143	0.8150	0.09	10.8646	10.8528	0.11
$x_{10,14}$	29.2272	29.2122	0.05	0.8122	0.8126	0.05	16.0297	15.9868	0.27
$x_{11,7}$	53.9613	53.9452	0.03	0.5096	0.5094	0.04	33.8074	33.8042	0.01

TABLE VI

Signal	Measured	Estimated	% Error
SIG1	202.14	208.39	3.09
SIG2	119.48	123.30	3.20
SIG3	187.04	193.08	3.23
SIG4	118.18	120.64	2.08
SIG5	166.56	169.78	1.93

TABLE VII

Signal	μ			ρ			σ		
	Measured	Estimated	% Error	Measured	Estimated	% Error	Measured	Estimated	% Error
x_0	1.43	1.43	0.00	0.9628	0.9628	0.00	7349.20	7349.20	0.00
x_1,x_2	1.59	1.58	0.63	0.9672	0.9695	0.24	8132.59	8135.16	0.03
x_3	0.16	0.16	0.00	0.9672	0.9695	0.24	812.92	813.52	0.07

TABLE VIII

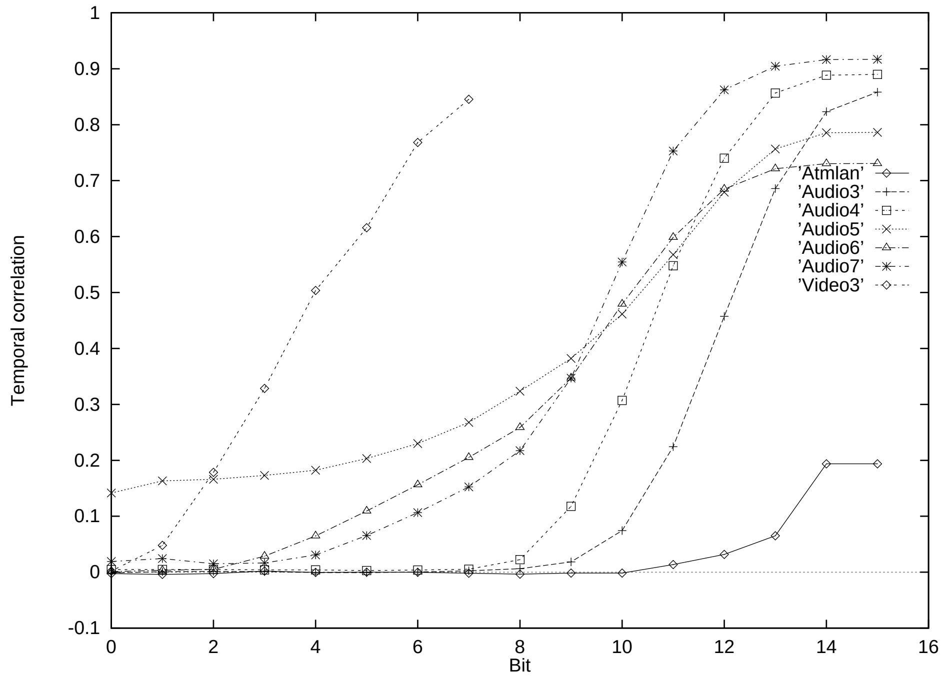


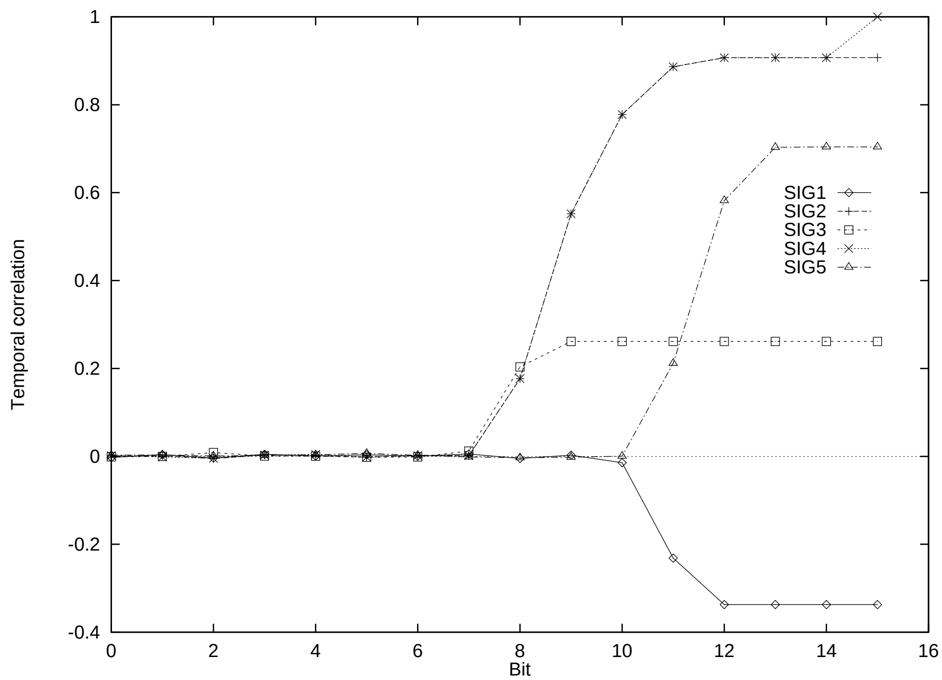
Fig. 3.

Signal	Direct form			Transpose		
	Measured	Estimated	% Error	Measured	Estimated	% Error
SIG1	35.22	35.52	0.85	35.68	35.97	0.81
SIG2	18.36	18.21	0.82	16.82	16.38	2.62
SIG3	26.86	26.92	0.22	26.33	27.25	3.49
SIG4	17.77	17.86	0.51	16.11	15.92	1.18
SIG5	24.88	24.38	2.01	23.66	23.26	1.69

TABLE IX

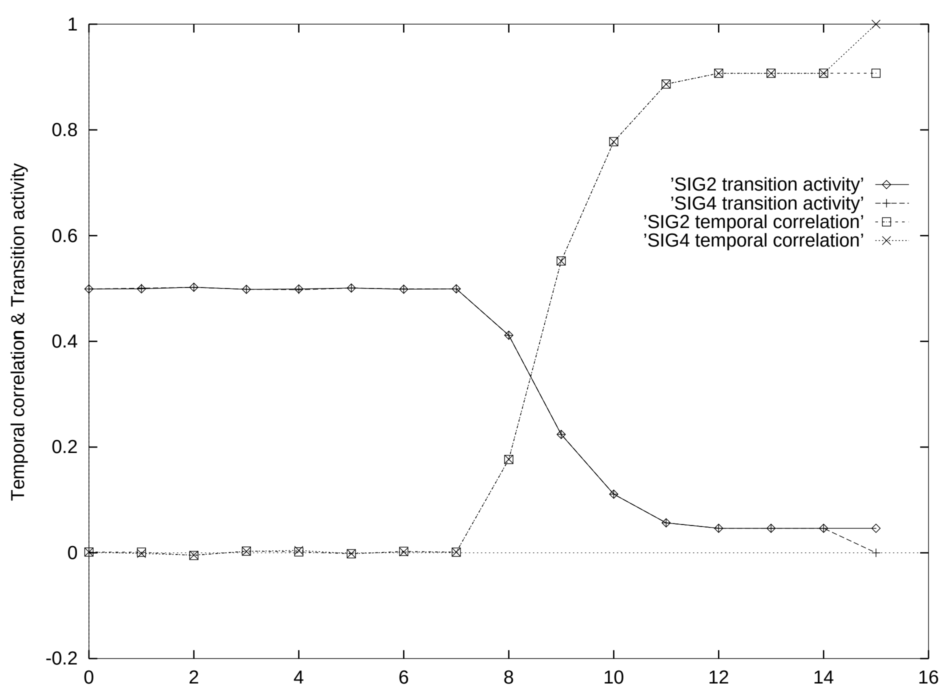
Data set	Description	μ	σ	ρ
Audio3	2.88MB of 16 bit PCM audio data (music)	1.4285	7349.20	0.9628
Audio4	2.88MB of 16 bit PCM audio data (music)	-17.6342	4040.40	0.9712
Audio5	0.37MB of 16 bit PCM audio data (speech)	59.4566	2661.75	0.9005
Audio6	0.61MB of 16 bit PCM audio data (speech)	23.6151	2328.79	0.9647
Audio7	2.88MB of 16 bit PCM audio data (music)	-39.3460	3086.30	0.9920
ATM LAN	0.80MB of 16 bit communications channel data	0.4861	5581.60	0.2952
Video3	9.70MB (380 QCIF frames) of 8 bit video data	99.7108	55.57	0.9199

TABLE X



Temporal correlation
Bit
SIG1
SIG2
SIG3
SIG4
SIG5

Fig. 4.



Temporal correlation & Transition activity
'SIG2 transition activity'
'SIG4 transition activity'
'SIG2 temporal correlation'
'SIG4 temporal correlation'

Fig. 5.

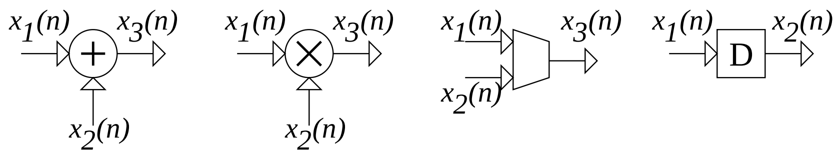


Fig. 6.

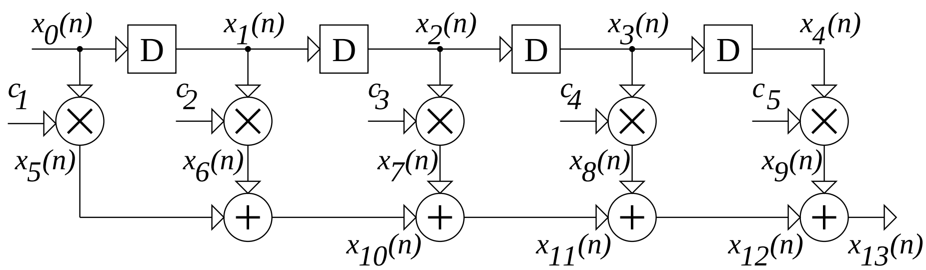


Fig. 7.

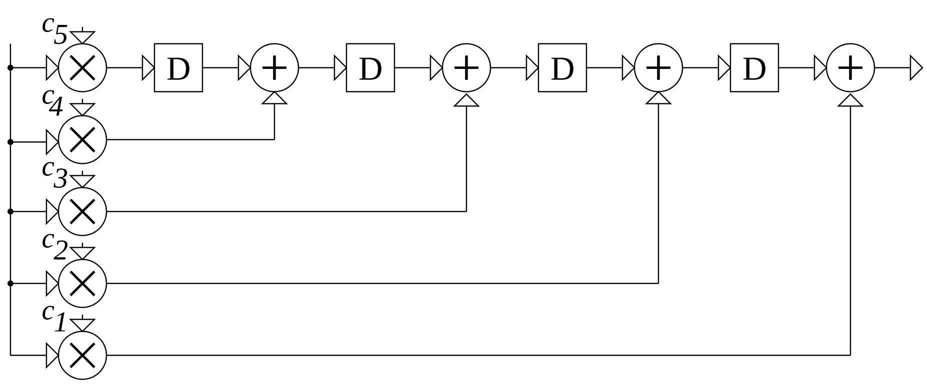


Fig. 8.

Signal	BP_0		BP_1	
	Measured	Estimated	Measured	Estimated
Audio3	10	11	16	15
Audio4	9	10	15	15
Audio5	0	3	15	14
Audio6	4	9	14	14
Audio7	5	9	14	14
ATM LAN	12	12	15	15
Video3	1	1	8	8

TABLE XI

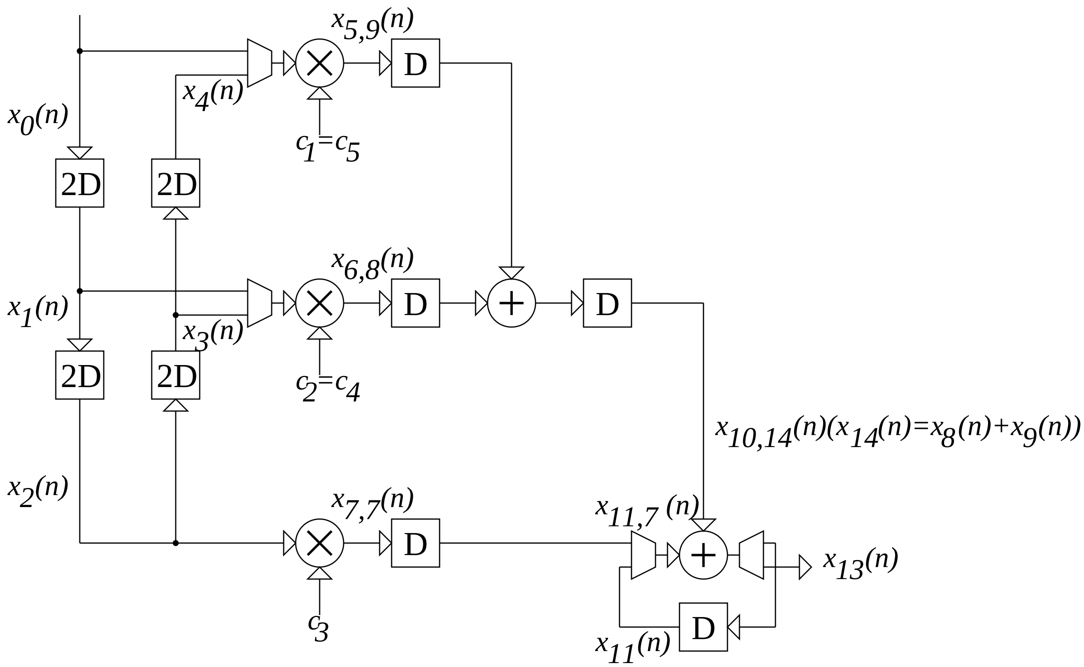


Fig. 9.

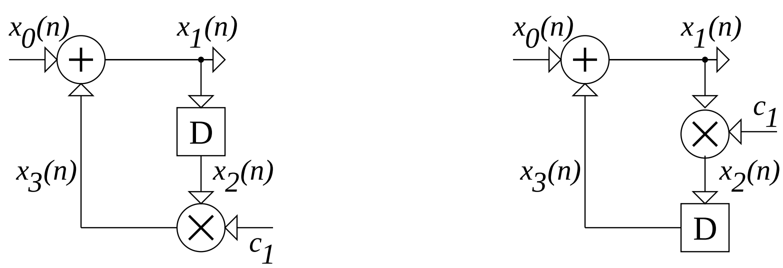


Fig. 10.

Signal	Unsigned, Two's complement			One's complement			Sign magnitude		
	Measured	Estimated	% Error	Measured	Estimated	% Error	Measured	Estimated	% Error
Audio3	6.42	6.32	1.56	6.43	6.32	1.71	6.17	6.24	1.13
Audio4	5.80	6.06	4.46	5.80	6.06	4.46	5.55	5.89	6.13
Audio5	4.78	4.40	7.95	4.79	4.40	8.14	4.22	4.23	0.24
Audio6	5.38	5.59	3.90	5.38	5.59	3.90	4.62	5.43	17.53
Audio7	5.05	5.52	9.31	5.05	5.52	9.31	4.78	5.44	13.81
ATM LAN	7.76	7.56	2.58	7.76	7.56	2.58	7.09	6.94	2.12
Video3	2.31	2.15	6.93	2.31	2.15	6.93	2.16	2.15	0.15

TABLE XII

Data set	Direct form			Transpose		
	Measured	Estimated	% Error	Measured	Estimated	% Error
Audio3	102.16	100.76	1.37	99.14	98.01	1.14
Audio4	91.40	94.37	3.25	88.42	90.62	2.49
Audio5	75.80	68.42	9.74	73.23	66.55	9.12
Audio6	84.94	86.63	1.99	82.03	83.09	1.29
Audio7	78.82	85.65	8.67	76.07	82.41	8.33
ATM LAN	129.35	124.76	3.55	127.94	122.23	4.46
Video3	31.58	33.02	4.56	28.29	31.64	11.84

TABLE XIII

Data set	Measured	Estimated	% Error
Audio3	159.14	158.29	0.53
Audio4	145.40	146.89	1.02
Audio5	123.40	132.94	7.73
Audio6	136.12	138.88	2.03
Audio7	130.60	135.04	3.40
ATM LAN	202.46	207.93	2.70
Video3	51.32	52.95	3.18

TABLE XIV

Data set	Direct form			Transpose		
	Measured	Estimated	% Error	Measured	Estimated	% Error
Audio3	24.36	24.26	0.41	22.92	22.56	1.57
Audio4	21.82	22.49	3.07	20.34	20.78	2.16
Audio5	18.06	17.15	5.04	16.99	15.55	8.48
Audio6	20.29	20.85	2.76	19.02	19.39	1.95
Audio7	18.87	20.33	7.74	17.38	18.59	6.96
ATM LAN	30.59	29.25	4.38	30.17	28.68	4.94
Video3	7.69	7.74	0.65	6.22	6.93	11.41

TABLE XV

Signal	Simulation	DBT	Approximate method	Fast Method
Audio3	42.30	6.38	2.25	0.06
Audio4	40.28	6.40	2.41	0.13
Audio5	5.00	0.85	3.10	1.23
Audio6	8.60	1.46	2.58	0.18
Audio7	39.16	6.65	2.58	0.21
ATM LAN	13.05	1.86	2.21	0.05
Video3	138.91	37.95	0.01	0.01

TABLE XVI