Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.

**ABSTRACT**

Low power linear digital signal processing circuits are fabricated based on a design synthesis process using activity metrics. The average activity value $\theta_i$ of all the input nodes of the circuit is determined. Architectural transformations of the circuit are performed in order to minimize the average activity value over all the nodes. The transformation resulting in the minimum activity value is the synthesized design used as the basis for fabricating a low power linear digital signal processing circuit.
FIG. 4

FIG. 5

OUTPUT ACTIVITY

VALUE OF CONSTANT
FIG. 6

OUTPUT ACTIVITY

VALUE OF CONSTANT

FIG. 7

FIG. 8

y=0.439506

y=0.337137

x=0

y=x

y=0.337137

x=0.5

y=0
INITIAL CIRCUITS SPECS.

FIG. 12

COMPUTE $\theta_i$ AT INPUTS TO ALL ADDER TREES BY SIMULATION

CAN AVG $\theta_i$ BE IMPROVED?

no

STOP

yes

RESTRUCTURE ALL ADDER TREES FOR MIN. AVG. $\theta_i$

FIG. 13

FIG. 14
SYNTHESIS OF LOW POWER LINEAR DIGITAL SIGNAL PROCESSING CIRCUITS USING ACTIVITY METRICS

This is a continuation of application Ser. No. 08/228,122 filed Apr. 15, 1994, now abandoned.

FIELD OF INVENTION

The present invention relates to a method of fabricating low power linear digital signal processing circuits based on a design synthesis process using activity metrics. Specifically, a data flow graph is derived which results in the minimum overall circuit activity value \( V \) for the circuit comprising word-parallel and/or bit-serial arithmetic operations. The result is a circuit design with minimal power consumption which then forms the basis of a fabricated circuit.

BACKGROUND OF THE INVENTION

Circuit power requirements have become an increasingly important design parameter as a result of the increasing use of portable systems, such as personal communication and computing systems. Linear digital signal processing (DSP) circuits have found widespread application in these systems where it is well known that power consumption in CMOS digital circuits is directly related to node activity. In such circuits, power consumption is attributable to three major components: leakage current, short circuit current and switching current. Leakage current is typically much smaller than either of the other currents. Short circuit current can be minimized by the use of proper circuit design techniques.

Interest in circuit design with power as a design parameter has been on the rise. Recently, studies have been conducted of power optimization issues during various stages of CAD, eg. technology mapping, multilevel logic optimization and FSM synthesis. In the high level synthesis area studies have been made of architectural transformations (which preserve the circuit input-output relationship) for circuit optimization based upon area, speed, and power criteria.

SUMMARY OF THE INVENTION


The present method does not make any assumption regarding gate-level implementations of adders and multipliers used in DSP synthesis. The method is based upon abstract functional models of DSP operations. In this regard, the methods are technology independent and must be analyzed at the hardware algorithm level. Moreover, the invention is not driven by accurate power estimation criteria since the invention is primarily concerned with determining if the overall power consumption increases or decreases as a result of an applied architectural transformation, without incurring a resulting high computation cost.

A principal object of the present invention is therefore, to provide a method of synthesizing low power linear digital signal processing circuit using activity metrics.

A further object of the invention is the synthesis of low power linear digital signal processing circuits using the design synthesized in accordance with the present invention.

Further and still other objects of the present invention will become more clearly apparent when the following description is read in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic diagram of a linear digital circuit model;
FIG. 2 is a data flow graph of an IIR Filter;
FIG. 3 is a schematic representation of a four-bit parallel adder;
FIG. 4 is a schematic representation of a bit-serial adder;
FIG. 5 is a graphical representation of the activity at the output of a bit-parallel multiplier as a function of the value of the multiplier constant;
FIG. 6 is a graphical representation of the activity at the output of a bit-serial multiplier as a function of the value of the multiplier constant;
FIG. 7 is a schematic representation of the arrangement used to perform activity measurements of an adder;
FIG. 8 is a graphical representation of activity experiments performed using word-parallel addition;
FIG. 9 is a graphical representation of activity experiments performed using bit-serial addition;
FIG. 10 is a schematic diagram of a balanced computation line for \( l=2\);
FIG. 11 is a schematic diagram of a pipelined bit-serial array of adders,
FIG. 12 is a preferred algorithm for low-power synthesis design;
FIG. 13 is a is a data flow graph of an IIR filter based on a design having the lowest average value of \( V \);
FIG. 14 is a data flow graph of an IIR filter based on a design having the largest average value of \( V \);

DETAILED DESCRIPTION OF THE INVENTION

Referring now to the figures and to FIG. 1 in particular, there is shown a schematic diagram of a linear digital circuit.

Circuitry in box 10 computes a linear transform of its \((n+m)\) inputs and generates \((n+w)\) outputs as shown. At time \( t \), data words corresponding to the circuit states (outputs of memory elements 12) \( s_1(t), s_2(t), \ldots, s_l(t) \) and the inputs \( u_1(t), u_2(t), \ldots, u_m(t) \) are fed to box 10, which in turn generates the outputs \( y_1(t), y_2(t), \ldots, y_w(t) \) and the circuit next states \( s_1(t+1), s_2(t+1), \ldots, s_l(t+1) \). The operation of the circuit can be represented mathematically by a system of equations given by \( Y(t+1)=MX(t)^T \), where \( X(t)=[s_1(t), s_2(t), \ldots, s_k(t), u_1(t), u_2(t), \ldots, u_m(t)] \) and \( Y(t+1)=[s_1(t+1), s_2(t+1), \ldots, s_k(t+1)] \).
Consider the expression for $s_4(t+1)$ given by Equation (1).

$$s(t + 1) = \sum_{i=1}^{m} m_i s_i(t) + \sum_{i=m+1}^{n} m_i p_i(t) \quad (1)$$

This equation can be implemented by first performing the multiplications $m_i s_i(t)$ and $m_i p_i(t)$ and then adding the results using an adder tree. Such a circuit, as shown in FIG. 1, is referred to as the computation tree for $s_4(t+1)$. In general, there will exist $(n+w)$ such computation trees in box 10. For logic minimization purposes, the values obtained at intermediate nodes of these trees can be shared by one or more computation trees.

In order to understand the invention, consider the data flow graph of an IIR filter shown in FIG. 2, which can be described by the equation.

$$s_1(t + 1) = \begin{bmatrix} c_1 & c_2 & 0 & 0 & k \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \\ s_4(t) \\ u(t) \end{bmatrix} + \begin{bmatrix} (c_1 + c_2) \\ 1 + c_2 \end{bmatrix} s_0(t)$$

The computation tree for $s_4(t+1)$ for example, is described by the equation $s_4(t+1)=c_1 s_1(t)+c_2 s_2(t)+c_3 s_3(t)+k u(t)$.

The filter of FIG. 2 can be implemented using either word-parallel or bit-serial arithmetic. In the word-parallel case, each signal (arc) of the data flow graph of FIG. 2 represents W bits of data comprising a data word. The W bits $b_{0w}, b_{1w}, \ldots, b_w$ are fed in parallel to the respective adders and multipliers shown in FIG. 2. The delays are designed to hold W bits of data in parallel. At time $t+1$, let $z$ out of W bits have different logic values than at time $t$. The ratio $\beta(t)=z/W$ is defined to be the signal activity at time $t$. The variable $\beta(t)$ is a random variable for different values of $t$ and represents a stochastic process. The average activity $\beta(t+1)$ of a signal over N consecutive time frames is defined as in Equation (2).

$$\beta(t+1) = \frac{1}{N} \sum_{i=1}^{N} \beta(t) \quad (2)$$

For example, consider the four-bit parallel adder shown in FIG. 3, where FA represents a combinational full adder. Let the inputs be $A=0100$, $B=0101$, and $C=0110$ and output $y_0, y_1, y_2, y_3=1111$ at time $t$. The input $A$ changes to $A=0001$, then at time $t+1$, the output becomes 1011. Hence the output activity is given by $\beta(t)=1/2$.

In the case of bit-serial arithmetic, the bit values of a data word are transmitted serially over a single data line over consecutive time steps. As an example, consider the bit-serial adder shown in FIG. 4. Initially, the FA value is set to logic ‘0’. To add $A=0001$ to $B=1010$, the sequence $a_3 a_2 a_1 a_0=1010$ and output $o_3 o_2 o_1 o_0=11111111$ at time $t$. The average activity $\beta(t)$ of a specified input is independent of the implementation of the circuit and depends only on the specification of M. This allows the use of the following procedure for low-power design.

First, by simulation methods, determine the average activities $\theta$ of all the inputs to the circuitry in box 10 in FIG. 1. Second, use architectural transformations on the implementation of the circuitry in box 10 to minimize the average value of $\theta$ over all its nodes. In order to determine how the data flow graph of the circuitry in box 10 should be transformed, it is necessary to...
study how node activity at the output of a constant multiplier is affected when the constant value is changed and how the activity at the output of an adder is affected by a change in the activity of either of its inputs.

In measuring the activity of constant multipliers, consider a multiplier that multiplies W-bit data words by a W-bit binary constant. First consider the case of word-parallel behavior can be explained as follows. A constant multiplier maps input X to output Y, given by Y = a·X, where -1 ≤ a ≤ 1. For a given constant value, the data input of the multiplier is simulated with random inputs with average activity 0.5. The number of patterns used for simulation was 20,000. FIG. 5 shows a graphical plot of the activity at the output of the multiplier versus the value of the multiplier constant, in the range -1.0 to +1.0. Negative numbers were represented using two’s complement arithmetic.

As is seen from the graph in FIG. 5, the average activity at the output of the multiplier monotonically increases with the absolute value of the multiplier constant. The observed behavior can be explained as follows. A constant multiplier maps input X to output Y, given by Y = a·X, where -1 ≤ a ≤ 1. Since Y and X are both W-bit wide, the 2W values of X are mapped onto values of Y in the range -a ≤ Y ≤ a. Since |a|<1, this mapping is many-to-one. The smaller the value of a, the larger the number of data values of X that are mapped to a single output value of Y. This causes the average activity of the output Y to drop off gradually with decreasing values of a.

The above experiment was repeated for the case of bit-serial multiplication. The results of the simulation of 0 ≤ a ≤ 1, are shown in FIG. 6. The graph for -1 ≤ a ≤ 1 is a mirror image about the y-axis of the graph of FIG. 6, and is not shown. The major difference between the graphs of FIGS. 5 and 6 is that in FIG. 6, the average output activity saturates to 0.5 for values of a<0.5 (due to intra-word activity).

In measuring the activity of adders consider the addition Z=X+Y. Let the average activities of the input X and Y be θX and θY, respectively. Experiments to measure the average activity of the output θo for a given value of θX and different values of θY were conducted. The experimental arrangement used to perform these measurements is shown in FIG. 7.

In FIG. 7, the selection of the constant α0 is used to create a stochastic process at the input X of the adder 20 with average activity θX. For a predetermined value of α0, and thereby θX, the average activity θo is tabulated for different values of θY. FIG. 8 graphically illustrates the results of two experiments performed using word-parallel addition for θX=0.337137 and θY=0.439506. The asymptotes y=0.337137 and y=0.439506 as well as line y=x are shown in the graph. The y-axis corresponds to the average activity θY at the output of the adder while the x-axis corresponds to the average activity θ0. An identical experiment was performed for bit-serial addition and the results are shown in FIG. 9.

### TABLE 3-continued

<table>
<thead>
<tr>
<th>Adder no.</th>
<th>in1 activity</th>
<th>in2 activity</th>
<th>max activity</th>
<th>output activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>0.275021</td>
<td>0.313297</td>
<td>0.313207</td>
<td>0.317222</td>
</tr>
<tr>
<td>7</td>
<td>0.349178</td>
<td>0.381570</td>
<td>0.381570</td>
<td>0.381559</td>
</tr>
<tr>
<td>8</td>
<td>0.568457</td>
<td>0.389430</td>
<td>0.384930</td>
<td>0.393405</td>
</tr>
</tbody>
</table>

In both of the above experiments the stochastic processes at the two adder inputs are mutually independent and uncorrelated. Under this assumption the following conclusion (hereby referred to as the max hypothesis) is reached from analysis of FIGS. 8 and 9.

For word-parallel and bit-serial arithmetic, the average activity θY at the output of the adder of FIG. 7 can be closely approximated by θY=α·θX, where 0 ≤ α ≤ 1. This result, although derived for uncorrelated inputs, was found to be applicable with reasonable accuracy to cases of correlated inputs due to reconvergent fanout and sequential feedback. The result can be applied to linear digital circuits since these circuits contain only constant multipliers and adders. Table 3 shows average activity values at the inputs and outputs of all the adders in the computational tree in FIG. 2 under sinusoidal input stimulation. The adders are numbered row-wise from left to right, starting with upper left corner. Hence, the max hypothesis is quite accurate under practical conditions which include signal correlation effects.

In order to perform synthesis for low power circuit design, consider a word-parallel computation tree with I inputs i1, i2, ..., iI and output y=Σi=1I ai·yi being rational constants. Assume that all operations are pipelined. For simplicity, let I = 2-1, where 1 is the number of levels in a perfectly balanced adder tree. FIG. 10 shows such a tree for I=2. If the input values to the tree are assumed to be mutually independent and uncorrelated, then the following result is obtained for word-parallel as well as bit-serial circuits the minimum average value of θY over all the nodes of a balanced adder tree with I inputs is obtained when (a) α0 ≥ α1 ≥ ... ≥ αI or (b) α0 ≤ α1 ≤ ... ≤ αI. That is, the value of αY monotonically increases or decreases in order of addition for minimum power requirement.

The result for Case (a) is proved by showing that any other ordering of the constants αi will cause the average activity to increase as follows. From the graphs of FIGS. 5 and 6, it is clear that θY>θY>...>θY, where θY is the average activity at the output of the multiplier corresponding to αY. Consider two adders at the topmost level of the adder tree, the first with inputs α0·i0 and α1·i1 and the second with inputs α0·i0 and α1·i1 such that α0>α1>α2>...>αI. If α0 and α1 are interchanged, then the average activity at the output of the first adder will still be proportional to α0 due to the max hypothesis. However, the average activity at the output of the second adder will increase from α0 to α1.

The proof for Case (b) is obtained by symmetry. The worst case assignment of constants to multipliers, that maximizes average θY, occurs when α0 ≥ α1 ≥ α2 ≥ ... ≥ αI. The above results can be trivially extended to adder trees that are not perfectly balanced. The following result is based on the max hypothesis that the minimum average value of θY over all the nodes of a computation tree is achieved by a pipelined bit-serial array of adders as in FIG. 11 for which α0 ≥ α1 ≥ α2 ≥ ... ≥ αI.

The synthesis algorithm is simulation-based and proceeds as shown in FIG. 12. The initial circuit specifications are defined 30 including deriving a data flow graph. The average...
activity \( \theta_i \), at each node \( i \) at the input to all the adder trees is computed 32 by simulation. The given circuit is simulated at the functional level with random, mutually independent circuit input values. The preferred simulation method is based on a Monte-Carlo power estimation technique proposed in Burch et al., “McPower: A Monte Carlo Approach to Power Estimation” in Proc. Euro. Des.-Autom Conf., pp 90 to 97, 1992.

The present invention uses simplifying heuristics to speed up the simulation process. The underlying measurements of \( \theta_i \) are very accurate and have been validated by simulations over large sampling periods. From the above measurements, the activities at the inputs to all adder trees are noted. By applying the max hypothesis, the adder trees are restructured as discussed earlier and average activities recomputed. The procedure forces additions with high activity to be moved closer to the root of a computation tree and vice versa.

A decision is then made 34 whether the average activity value \( \theta_i \) can be minimized, i.e. improved. The decision is made by determining whether \( \theta_i \) monotonically increases or monotonically decreases. If the activity value cannot be improved, the process is terminated 36 and the design synthesis is completed. If the activity value can be improved, all of the adder trees are restructured 38 to improve the minimum average activity value \( \theta_i \). The restructured tree configuration is provided to recompute the average activity value 32 and the process repeats until it is decided 34 that the average activity value cannot be further minimized, at which time the process stops 36 and the design synthesis is complete.

After the design synthesis is stopped 36, a circuit is fabricated 40 based on the synthesized design. The circuit comprises either word-parallel or bit-serial arithmetic devices, the fabrication of which forms a combinational logic circuit and is well known in the art.

The optimization algorithm was applied to several practical circuits and the decrease in the average value of \( \theta_i \) over all the circuit nodes was measured. This is a close approximation to the power consumption savings as other factors such as node capacitance and power supply voltage were left unchanged by the data flow graph transformations. It was assumed that word-parallel implementations were pipelined and that the outputs of parallel adders and multipliers were clocked into data registers. FIG. 13 shows the transformed data flow graph of the IIR filter in FIG. 2 corresponding to the lowest average value of \( \theta_i \) over all the circuit nodes. For comparison, the worst-case data flow graph for the same IIR filter in FIG. 2 is shown in FIG. 14 (with the largest average value of \( \theta_i \)).

The synthesis procedure of FIG. 12 uses random input values for circuit simulation. Experiments were conducted to validate the conclusions for other input waveforms. Table 4 shows results obtained on several test circuits. For each circuit, measurements of average \( \theta_i \) were made for the best-case and worst-case data flow graph transformations for both word-parallel (WP) as well as bit-serial (BS) implementations. For each circuit, a measurement was made of how well it performed with other types of stimuli: random, sinusoidal (Sin), and speech input (Audio 1 and Audio 2) waveforms. Among the four circuits considered, CKT1 and CKT2 are two elliptic filters, CKT3 is a control circuit and CKT4 is an IIR filter. In Table 4, WC and BC stand for the average values of \( \theta_i \) for the worst-case and best-case data flow graphs. The last column is a measure of the power savings possible. From Table 4, it is clear that the data flow graph transformations made by the synthesis procedure of the present invention perform equally well for different input signals.

The present invention provides a new synthesis procedure for low power linear DSP circuits. The activity effects in adders and multipliers is used to derive optimal designs of computation trees of linear circuits from which a method for designing low-power linear sequential circuits is developed. The method uses the knowledge that the stochastic activity process is strict-sense stationary. While there has been described and illustrated a preferred method of synthesizing low power linear DSP circuits, it will be apparent to those skilled in the art that variations and modifications are possible without deviating from the spirit and broad principles of the present invention which shall be limited solely by the scope of the claims appended hereto.

What is claimed is:

1. A method of synthesizing a low power linear digital signal processing circuit comprising the steps of:
   (a) defining initial circuit specifications;
   (b) computing the average activity value \( \theta_i \) at the node \( i \) for each input of the circuit;
   wherein said average activity value is derived by averaging the result of a stochastic process representative of potential changes in the state of said node over a finite number of consecutive time frames;
   (c) determining if the computed average activity value is the minimum value,
      (i) if the computed average activity value is not the minimum value, transforming the circuit to minimize the average activity value and return to step (b);
      (ii) if the computed average activity value is the minimum value, selecting the circuit used for computing the minimum value \( \theta_i \) as a synthesized design;
   (d) synthesizing a low power linear digital signal processing circuit based on said synthesized design.

2. A method of synthesizing a low power linear digital signal processing circuit as set forth in claim 1 where said computing the average activity value \( \theta_i \) at node \( i \) is performed by minimizing the value \( \sum_{j=0}^{N-1} \langle i, j \rangle \sum_{i=0}^{C-1} \), where \( y \) is the total number of nodes, \( N \) is the total number of said consecutive time frames and \( C \) is the capacitive load on the \( n \)th node.
3. A method of synthesizing a low power linear digital signal processing circuit as set forth in claim 2, where said determining step is performed by further determining whether multiplier constants \( a_i \) monotonically increase or monotonically decrease in order of addition.

4. A method of fabricating a low power linear digital signal processing circuit as set forth in claim 3, where said synthesized design comprises word-parallel arithmetic devices.

5. A method of fabricating a low power linear digital signal processing circuit as set forth in claim 3, where said synthesized design comprises bit-serial arithmetic devices.

6. A method of synthesizing a low power linear digital signal processing circuit as set forth in claim 1, where said determining step is performed by further determining whether multiplier constants \( a_i \) monotonically increase or monotonically decrease in order of addition.

7. A method of fabricating a low power linear digital signal processing circuit as set forth in claim 1, where said defining initial circuit specifications is in terms of multipliers and adder trees.

8. A method of fabricating a low power linear digital signal processing circuit as set forth in claim 7, where said computing is performed by simulation.

9. A method of synthesizing a low power linear digital signal processing circuit as set forth in claim 7, where said transforming the circuit comprises restructuring the tree configuration.

10. A method of fabricating a low power linear digital signal processing circuit as set forth in claim 1, where said defining initial circuit specification is in terms of a data flow graph.

11. A method of synthesizing a low power linear digital signal processing circuit as set forth in claim 10, where said transforming the circuit comprises architectural transformation.

12. A method of fabricating a low power linear digital signal processing circuit as set forth in claim 1, where said synthesized design comprises word-parallel arithmetic devices.

13. A method of fabricating a low power linear digital signal processing circuit as set forth in claim 1, where said synthesized design comprises bit-serial arithmetic devices.

14. A low power linear digital signal processing circuit fabricated in accordance with the method as set forth in claim 1.

15. A low power linear digital signal processing circuit fabricated in accordance with the method as set forth in claim 2.

16. A low power linear digital signal processing circuit fabricated in accordance with the method as set forth in claim 3.

17. A low power linear digital signal processing circuit fabricated in accordance with the method as set forth in claim 4.

18. A low power linear digital signal processing circuit fabricated in accordance with the method as set forth in claim 5.

19. A low power linear digital signal processing circuit fabricated in accordance with the method as set forth in claim 12.

20. A low power linear digital signal processing circuit fabricated in accordance with the method as set forth in claim 13.

* * * * *
UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,808,917
DATED : September 15, 1998
INVENTOR(S) : Abhijit CHATTERJEE and Rabindra K. ROY

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

On the title page item [54],
In the Title, please delete “SIGNAL”.

Column 2, line 5, delete “of”.

Column 4, line 47, please delete “$\sum_{i=1}^{i=n} \theta$” and insert -- $\sum_{i=1}^{i=n} \theta$ --. (first occurrence)

Column 6, line 32, delete “$\sum_{i=0}^{i=n} \alpha$” and insert -- $\sum_{j=0}^{j=n} \alpha$ --.

Column 8, line 64, please delete “$\sum_{i=1}^{i=n} \theta$” and insert -- $\sum_{i=1}^{i=n} \theta$ --.

Column 9, line 5, delete “fabricating” and insert -- synthesizing --;
Column 9, line 9, delete “fabricating” and insert -- synthesizing --;
Column 9, line 17, delete “fabricating” and insert -- synthesizing --;
Column 9, line 19, delete “defining” and insert -- defining --;
Column 9, line 20, delete “fabricating” and insert -- synthesizing --; and
Column 9, line 27, delete “fabricating” and insert -- synthesizing --.
UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 5,808,917
DATED : September 15, 1998
INVENTOR(S) : Abhijit CHATTERJEE and Rabindra K. ROY

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 10, line 1, delete “fabricating” and insert --synthesizing--; and

Column 10, line 5, delete “fabricating” and insert --synthesizing--.

Signed and Sealed this
Fourteenth Day of September, 1999

Attest:

Q. TODD DICKINSON
Acting Commissioner of Patents and Trademarks