A Spurious-Power Suppression Technique for Multimedia/DSP Applications

Kuan-Hung Chen, *Member, IEEE*, and Yuan-Sun Chu, *Member, IEEE*

***Abstract*—This paper presents the design exploration and applications of a spurious-power suppression technique (SPST) which can dramatically reduce the power dissipation of combinational VLSI designs for multimedia/DSP purposes. The proposed SPST separates the target designs into two parts, i.e., the most significant part and least significant part (MSP and LSP), and turns off the MSP when it does not affect the computation results to save power. Furthermore, this paper proposes an original glitch-diminishing technique to filter out useless switching power by asserting the data signals after the data transient period. This paper adopts two multimedia/DSP design examples, i.e., a multitransform design for H.264 and a versatile multimedia functional unit (VMFU), to evaluate the proposed SPST. These two design examples have quite different hardware configurations, thus, the realization issues of the SPST on every design also remarkably differ from each other. The multitransform design can compute three transforms which are required in H.264 encoding while the VMFU possesses six commonly used multimedia/DSP functions, namely, addition, subtraction, multiplication, MAC, interpolation, and sum-of-absolute-difference. After optimizing the design elaborately, we find that the proposed SPST can, respectively, save 27% and 24% power dissipation on average of the H.264 multitransform design and the VMFU at the expense of less than 20% area augmentation.**

***Index Terms*—Digital-signal processing chips, image coding, low-power design, video coding.**

I. Introduction

ONE OF THE accompanying challenges in designing ICs for portable electrical devices is lowering down the power consumption to prolong the operating time on the basis of given limited energy supply from batteries. Owing to the vigorous development of the wireless infrastructure and the personal electronic devices like video mobile phones, mobile TV sets, PDAs, etc., multimedia and DSP applications have been adopted in wireless environments. However, advanced multimedia/DSP applications such as H.264 CODECs induce much more algorithmic complexity [1], [2], which increases the power consumption in real-time operation besides the cost in implementation. Therefore, dedicated low-power techniques are undoubtedly important for multimedia/DSP VLSI implementation.

Various techniques have been developed for reducing the power consumption of VLSI designs, including voltage scaling, switched-capacitance reduction, clock gating, power-down techniques, threshold-voltage controlling, multiple supply voltages, and dynamic voltage frequency scaling [3]–[6]. These low-power techniques have been proven to be efficient at certain expense and are applicable to multimedia/DSP designs. Among these low-power techniques, a promising direction for significantly reducing power consumption is reducing the dynamic power which dominates total power dissipation. Consequently, this paper develops a new low-power technique which can reduce dynamic power. The proposed low-power technique can be used with some of the aforementioned techniques without conflicts to further reduce the power consumption of the multimedia/DSP designs. The existing works that reduce the dynamic power consumption by minimizing the switched capacitance include the designs in [7]–[12]. The design in [7] proposes a concept called partially guarded computation (PGC), which divides the arithmetic units, e.g., adders and multipliers, into two parts and turns off the unused part to minimize the power consumption. The reported results show that the PGC can reduce power consumption by 10%–44% in an array multiplier with 30%–36% area overheads in speech-related applications. However, the PGC technique cannot gain any power reduction when applied on adders because of the overhead-augmented circuitry. The design in [8] proposes a 32-bit 2's complement adder equipping a two-stage (master and slave stages) flip-flop at each of the two inputs, a dynamic-range determination (DRD) unit and a sign-extension (SE) unit, which tends to reduce the power dissipation of conventional adders for multimedia applications. Additionally, the design in [9] presents a multiplier using the DRD unit to select the input operand with a smaller effective dynamic range to yield the Booth codes. However, the DRD unit induces additional delay and area overheads. Besides, the input data flows are also frequently switched if the input operands with a smaller effective dynamic range often change between operands A and B, and vice versa. In such cases, the power dissipation of the designs in [8] and [9] is increased rather than decreased. The design in [10] incorporates a technique for glitching power minimization by replacing some existing gates with functionally equivalent ones that can be frozen by asserting a control signal. This technique can be applied to replace layout-level descriptions and guarantees predictable results. However, it can only achieve savings of 6.3% in total power dissipation, since it operates in the layout-level environment which is tightly restricted. The design in [11] proposes a double-switch circuit-block switch scheme capable of reducing power dissipation during downtime by shortening the settling time after reactivation. The drawbacks of the scheme are the necessity for

Manuscript received December 27, 2006; revised September 2, 2007. First published June 6, 2008; current version published February 4, 2009. This paper was recommended by Associated Editor J. R. Chen.

K.-H. Chen is with the Department of Electronic Engineering, Feng Chia University, Tai-Chung 407, Taiwan (e-mail: kuanhung@fcu.edu.tw).

Y.-S. Chu is with the Department of Electrical Engineering, National Chung Cheng University, Chiayi 621, Taiwan (e-mail: chu@ee.ccu.edu.tw).

Digital Object Identifier 10.1109/TCSI.2008.926582

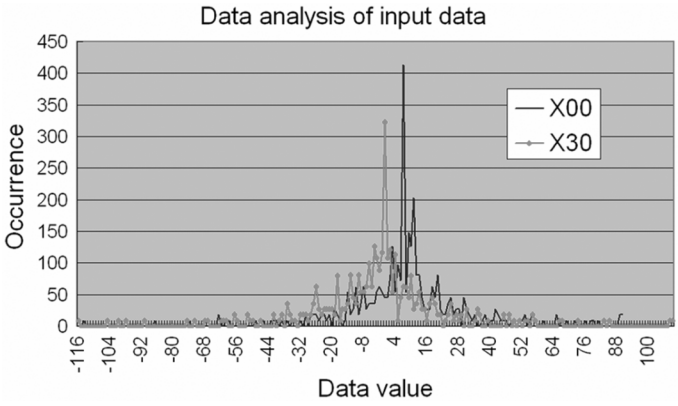


Fig. 1. Data analysis of the input data in the transform design, as shown in Fig. 8, when executing the H.264 forward transform.

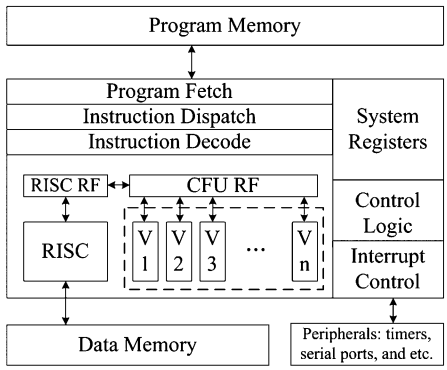


Fig. 2. Multimedia/DSP processor encapsulating VMFUs as denoted by a dotted rectangle.

two independent virtual power rails and the requirement for two additional transistors for switching each cell. At last, the design in [12] presents a DCT core exploiting an adaptive bandwidth approach and a method which trades off power consumption and arithmetic precision.

This paper explores the circuitry of implementing the so-called *Spurious-Power Suppression Technique* (SPST), as well as two design examples for the multimedia/DSP applications. The SPST can dramatically reduce the power dissipation of combinational VLSI designs for multimedia/DSP applications. The data of the multimedia/DSP computations, such as transform coding and texture coding in MPEG-1/2/4/H.264 systems, tend to fluctuate within a small range of bit width due to the temporal and spatial redundancies existing in video signals. However, the corresponding hardware design still needs to provide the maximum data bit width to avoid data accuracy loss. Fig. 1 shows the data analysis of the input pattern (i.e., X00 and X30) of the adder/subtractor in MPE-I-0, as shown in Fig. 8 which is introduced in Section III, in an example dealing with the forward transformation of a real video sequence. Fig. 1 shows that most of the input data values fall in the range between -2^6 and $+2^6$, implying that high-byte data only rarely affect the computational results. This scenario creates a good opportunity to minimize the extra power dissipation by elaborately optimizing the circuitry. From the viewpoint of logic design, the adders/subtractors in the transform coding design are separated into two parts, i.e., the *Most Significant Part* (MSP) and the *Least Significant Part* (LSP), and the input data of the MSP circuits are latched whenever they do not influence the computation results. Besides, detection-logic and SE units are introduced in the proposed SPST to determine the effective ranges of the operands and compensate for the sign signals of the MSP, respectively. Although this concept is similar to the PGC [7], efficiently implementing the concept and manifesting its effects on power saving in real circuits remains challenging. To conquer this design challenge, this paper explores the basic reasons of the occurrence of spurious power and presents a thorough analysis for efficiently implementing SPST adders/subtractors. From those analyses and the implementing experience, we discover the damage of the glitches to the combinational circuits and further propose a glitch-diminishing technique, a novel idea of this paper, to visibly filter out useless switching power by asserting the data signals after the data transient period. This action is realized easily by controlling the three-bit output of the detection-logic unit with extremely tiny cost. A similar design concept has been found in [10]. Nevertheless, this concept is used here in the logic level, while in [10], it is in the layout level.

In the two design examples, the first one is an *Efficient multi-Transform Design* (ETD) for MPEG-4 AVC/H.264 [13], [14], and the second one is a *Versatile Multimedia Functional Unit* (VMFU) design [15]. The ETD can compute three transforms, i.e., the forward, inverse, and Hadamard transform, adopted in H.264 video encoding. The ETD can achieve the real-time performance requirements of HD720, HD1080, and digital cinema of the H.264 encoding system when operated at 22, 50, and 100 MHz, respectively. On the other hand, the VMFU can compute six commonly used arithmetic operations in multimedia/DSP processing, i.e., addition, subtraction, multiplication, MAC, interpolation, and Sum-of-Absolute-Difference (SAD). Encapsulating the VMFUs, as shown in Fig. 2, designers can increase the flexibility and scalability of multimedia/DSP processors. When applying the SPST to these two designs, the realization issues in every design highly differ from each other due to the large hardware-configuration differences. However, with an elaborate design optimization, the proposed SPST can reduce power dissipation by an average of 27% and 24% for the ETD and the VMFU with 1.8-V supply voltage, respectively.

The remainder of this paper is arranged as follows. Section II explores the proposed SPST technique in detail. Section III, then, presents two design examples adopting the SPST which are for H.264 transform coding, as well as for the arithmetic operations used in multimedia/DSP purposes. Section IV discusses the implementation and verification of the proposed SPST designs. Section V shows the analysis and comparisons in implementing approaches and power reductions. Finally, Section VI presents conclusions.

II. Proposed SPST

Besides the explanations presented in our former studies [14], [15], this paper provides further illustrations of the proposed SPST as described in the following sections.

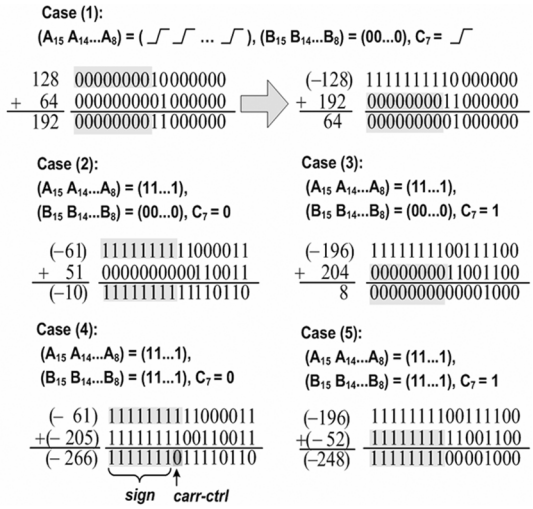
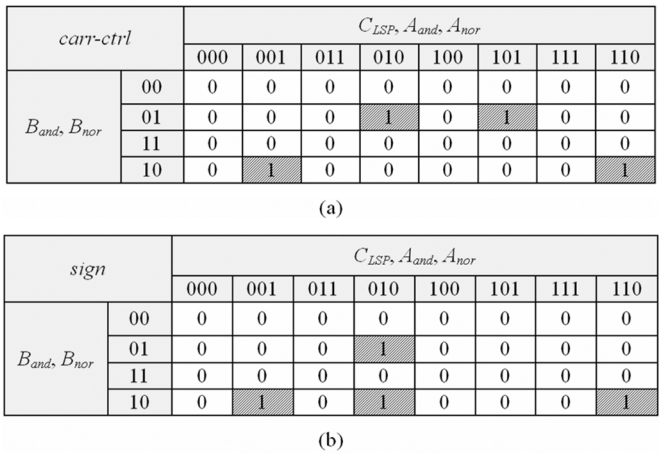


Fig. 3. Spurious transitions in the multimedia/DSP computations.



carr-ctrl		$C_{LSP}, A_{and}, A_{nor}$							
		000	001	011	010	100	101	111	110
B_{and}, B_{nor}	00	0	0	0	0	0	0	0	0
	01	0	0	0	1	0	1	0	0
	11	0	0	0	0	0	0	0	0
	10	0	1	0	0	0	0	0	1

(a)

sign		$C_{LSP}, A_{and}, A_{nor}$							
		000	001	011	010	100	101	111	110
B_{and}, B_{nor}	00	0	0	0	0	0	0	0	0
	01	0	0	0	1	0	0	0	0
	11	0	0	0	0	0	0	0	0
	10	0	1	0	1	0	0	0	1

(b)

Fig. 4. Representations of (a) *carr-ctrl* signal and (b) *sign* signal in terms of *KARNAUGH* maps.

A. Theoretical Analysis and Logic Derivation

To illustrate the reason of those spurious signal transitions shown in Fig. 1, we explore five cases of 16-bit additions as shown in Fig. 3. The cases of exchanging the operands A and B in additions lead to the same spurious transitions with those shown in Fig. 3. Hence, there is probably no other case beyond these five based on this design. The first case illustrates a transient state in which spurious transitions of carry signals occur in the MSP, although the final result of the MSP is unchanged. Meanwhile, the second and third cases describe situations involving one negative operand adding another positive operand without and with carry-in from the LSP, respectively. Moreover, the fourth and fifth cases demonstrate the addition of two negative operands without and with carry-in from the LSP, respectively. In those cases, the results of MSP are predictable; therefore, the computations in MSP are useless and can be neglected. Eliminating those spurious computations not only can save the power consumption inside the adder/subtractor in the current stage but also can decrease the glitching noises which cause power wastage inside the arithmetic circuits in the next stage. From the analysis of Fig. 3, we are motivated to propose the SPST that separates the adder/subtractor into two parts and then latches the input data of the MSP whenever they do not affect the computation results. The SPST can be expanded to be a fine-grain scheme in which the adder/subtractor is divided into more than two parts. However, the hardware complexity of the augmented circuits such as the detection-logic unit, the data latches, and the SE unit increases dramatically. Based on an adder/subtractor example, we actually find that the power expense caused by the augmented circuits is larger than the power reduction in a tripartitioned scheme. This is the reason we propose a bipartitioned SPST scheme in this paper.

To know whether the MSP affects the computation results in the bipartitioned SPST scheme, a detection-logic unit must be used to detect the effective input ranges. The Boolean logical equations shown as follows express the behavioral principles of the detection-logic unit:

$$A_{\mathrm{MSP}} = A[15:8] \quad B_{\mathrm{MSP}} = B[15:8] \tag{1}$$

$$A_{\mathrm{and}} = A[15] \times A[14] \times \cdots \times A[8] \tag{2}$$

$$B_{\mathrm{and}} = B[15] \times B[14] \times \cdots \times B[8] \tag{3}$$

$$A_{\mathrm{nor}} = \overline{A[15] + A[14] + \cdots + A[8]} \tag{4}$$

$$B_{\mathrm{nor}} = \overline{B[15] + B[14] + \cdots + B[8]} \tag{5}$$

$$close = \overline{(A_{\mathrm{and}} + A_{\mathrm{nor}}) \times (B_{\mathrm{and}} + B_{\mathrm{nor}})} \tag{6}$$

where $A[\mathrm{m}]$ and $B[\mathrm{n}]$, respectively, denote the mth bit of the operand A and the nth bit of the operand B, and A_{MSP} and B_{MSP}, respectively, denote the MSP parts, i.e., the 9th bit to the 16th bit, of the operands A and B in the examples shown in Fig. 3. When the bits in A_{MSP} and/or in B_{MSP} are all ones, the value of A_{and} and/or that of B_{and}, respectively, become one, while when the bits in A_{MSP} and/or in B_{MSP} are all zeros, the value of A_{nor} and/or that of B_{nor}, respectively, turn into one. Being one of the three outputs of the detection-logic unit, *close* denotes whether the MSP circuits can be neglected or not. When the two input operands can be classified into one of the five cases shown in Fig. 3, the value of *close* becomes zero, which indicates that the MSP circuits can be closed to save power dissipation. This design intends to close the MSP circuits by feeding zero inputs into them, which may freeze the switching activities in the MSP circuits to avoid dynamic power consumption. Compared with the use of transmission gates to latch the inputs, this scheme can prevent the voltage-drop problems caused by the floating-connected points after the MSP circuits are closed for a relatively long span of time. The ways to compensate for the sign bits of the computing results are also shown in case 4 in Fig. 3. Accordingly, we derive the *KARNAUGH* maps shown in Fig. 4 which lead to the Boolean logical equations

$$\begin{aligned} carr-ctrl =& \overline{C_{\mathrm{LSP}}} \times \overline{A_{\mathrm{and}}} \times A_{\mathrm{nor}} \times B_{\mathrm{and}} \times \overline{B_{\mathrm{nor}}} \\ &+ \overline{C_{\mathrm{LSP}}} \times A_{\mathrm{and}} \times \overline{A_{\mathrm{nor}}} \times \overline{B_{\mathrm{and}}} \times B_{\mathrm{nor}} \\ &+ C_{\mathrm{LSP}} \times \overline{A_{\mathrm{and}}} \times A_{\mathrm{nor}} \times \overline{B_{\mathrm{and}}} \times B_{\mathrm{nor}} \\ &+ C_{\mathrm{LSP}} \times A_{\mathrm{and}} \times \overline{A_{\mathrm{nor}}} \times B_{\mathrm{and}} \times \overline{B_{\mathrm{nor}}} \\ =& \overline{C_{\mathrm{LSP}}} \times (\overline{A_{\mathrm{and}}} \times B_{\mathrm{and}} + A_{\mathrm{and}} \times \overline{B_{\mathrm{and}}}) \\ &\times (A_{\mathrm{and}} \times B_{\mathrm{and}} + A_{\mathrm{and}} \times B_{\mathrm{nor}} + A_{\mathrm{nor}} \times B_{\mathrm{and}} \\ &\quad + A_{\mathrm{nor}} \times B_{\mathrm{nor}}) + C_{\mathrm{LSP}} \\ &\times (A_{\mathrm{and}} \times B_{\mathrm{and}} + \overline{A_{\mathrm{and}}} \times \overline{B_{\mathrm{and}}}) \\ &\times (A_{\mathrm{and}} \times B_{\mathrm{and}} + A_{\mathrm{and}} \times B_{\mathrm{nor}} + A_{\mathrm{nor}} \times B_{\mathrm{and}} \\ &\quad + A_{\mathrm{nor}} \times B_{\mathrm{nor}})(C_{\mathrm{LSP}} \oplus A_{\mathrm{and}} \oplus B_{\mathrm{and}}) \\ &\times (A_{\mathrm{and}} + A_{\mathrm{nor}}) \times (B_{\mathrm{and}} + B_{\mathrm{nor}}) \end{aligned} \tag{7}$$

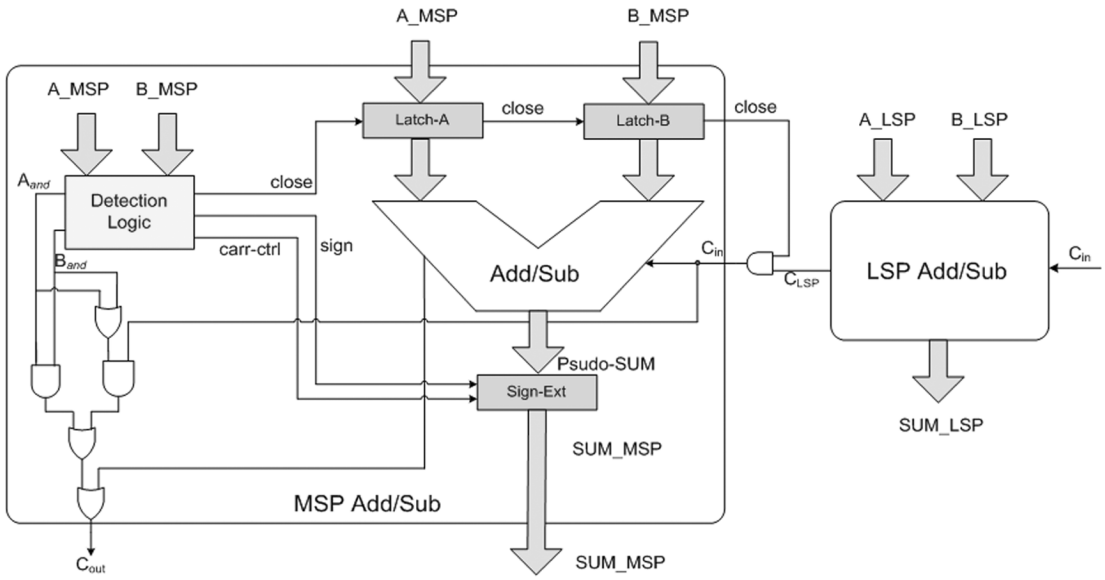


Fig. 5. Low-power adder/subtractor design example adopting the proposed SPST.

$$\begin{aligned} sign = {} & \overline{C_{\mathrm{LSP}}} \times (\overline{A_{\mathrm{and}}} \times A_{\mathrm{nor}} \times B_{\mathrm{and}} \times \overline{B_{\mathrm{nor}}} + A_{\mathrm{and}} \\ & \times \overline{A_{\mathrm{nor}}} \times \overline{B_{\mathrm{and}}} \times B_{\mathrm{nor}} + A_{\mathrm{and}} \times \overline{A_{\mathrm{nor}}} \\ & \times B_{\mathrm{and}} \times \overline{B_{\mathrm{nor}}}) \\ & + C_{\mathrm{LSP}} \times A_{\mathrm{and}} \times \overline{A_{\mathrm{nor}}} \times B_{\mathrm{and}} \times \overline{B_{\mathrm{nor}}} \\ = {} & \overline{C_{\mathrm{LSP}}} \times (\overline{A_{\mathrm{and}}} \times B_{\mathrm{and}} + A_{\mathrm{and}}) \\ & + C_{\mathrm{LSP}} \times A_{\mathrm{and}} \times B_{\mathrm{and}} \\ = {} & \overline{C_{\mathrm{LSP}}} \times (A_{\mathrm{and}} + B_{\mathrm{and}}) + C_{\mathrm{LSP}} \times A_{\mathrm{and}} \times B_{\mathrm{and}} \end{aligned} \tag{8}$$

for the *carr-ctrl* and the *sign* signals, respectively.

B. Realization Issues of the Proposed SPST

Fig. 5 shows a 16-bit adder/subtractor design example adopting the proposed SPST. In this example, the 16-bit adder/subtractor is divided into MSP and LSP between the eighth and the ninth bits. Latches implemented by simple AND gates are used to control the input data of the MSP. When the MSP is necessary, the input data of MSP remain unchanged. However, when the MSP is negligible, the input data of the MSP become zeros to avoid glitching power consumption. The two operands of the MSP enter the detection-logic unit, except the adder/subtractor, so that the detection-logic unit can decide whether to turn off the MSP or not. Based on the derived Boolean equations (1) to (8), the detection-logic unit of SPST is shown in Fig. 6(a), which can determine whether the input data of MSP should be latched or not. Moreover, we propose the novel glitch-diminishing technique by adding three 1-bit registers to control the assertion of the *close*, *sign*, and *carr-ctrl* signals to further decrease the transient signals occurred in the cascaded circuits which are usually adopted in VLSI architectures designed for multimedia/DSP applications. The timing diagram is shown in Fig. 6(b). A certain amount of delay Φ is used to assert the *close*, *sign*, and *carr-ctrl* signals after the period of data transition Ψ which is achieved by controlling the three 1-bit registers at the outputs of the detection-logic unit. Hence, the transients of the detection-logic unit can be filtered out; thus, the data latches shown in Fig. 5 can prevent the glitch signals from flowing into the MSP with tiny cost. The data transient time Ψ and the earliest required time of all the inputs Δ are also illustrated in Fig. 6(b). The delay Φ should be set in the range of $\Psi < \Phi < \Delta$, which is shown as the shadow area in Fig. 6(b), to filter out the glitch signals as well as to keep the computation results correct. Based on Figs. 5 and 6, the timing issue of the SPST is analyzed as follows.

1) *When the detection-logic unit turns off the MSP:* At this moment, the outputs of the MSP are directly compensated by the SE unit; therefore, the time saved from skipping the computations in the MSP circuits shall cancel out the delay caused by the detection-logic unit.
2) *When the detection-logic unit turns on the MSP:* The MSP circuits must wait for the notification of the detection-logic unit to turn on the data latches to let the data in. Hence, the delay caused by the detection-logic unit will contribute to the delay of the whole combinational circuitry, i.e., the 16-bit adder/subtractor in this design example.
3) *When the detection-logic unit remains its decision:* No matter whether the last decision is turning on or turning off the MSP, the delay of the detection logic is negligible because the path of the combinational circuitry (i.e., the 16-bit adder/subtractor in this design example) remains the same.

From the analysis earlier, we can know that the total delay is affected only when the detection-logic unit turns on the MSP. However, the detection-logic unit should be a speed-oriented design. When the SPST is applied on combinational circuitries, we should first determine the longest transitions of the interested cross sections of each combinational circuitry, which is a timing characteristic and is also related to the adopted technology. The longest transitions can be obtained from analyzing the timing differences between the earliest arrival and the latest arrival signals of the cross sections of a combinational circuitry. Then, a delay generator similar to the delay line used in the DLL

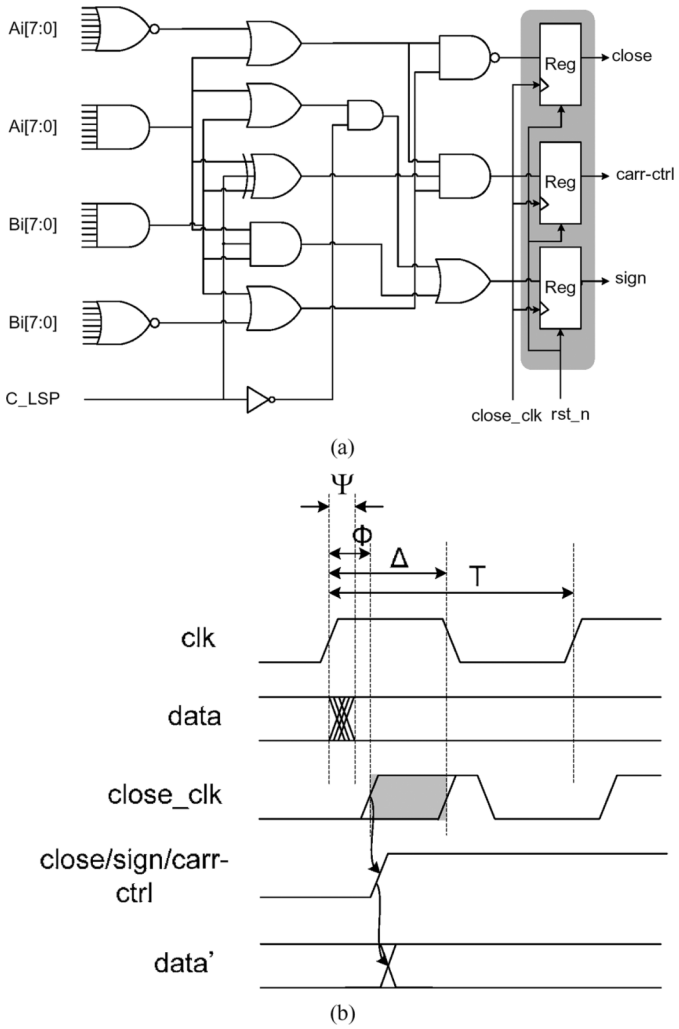


Fig. 6. (a) Detection-logic unit and (b) its timing diagram.

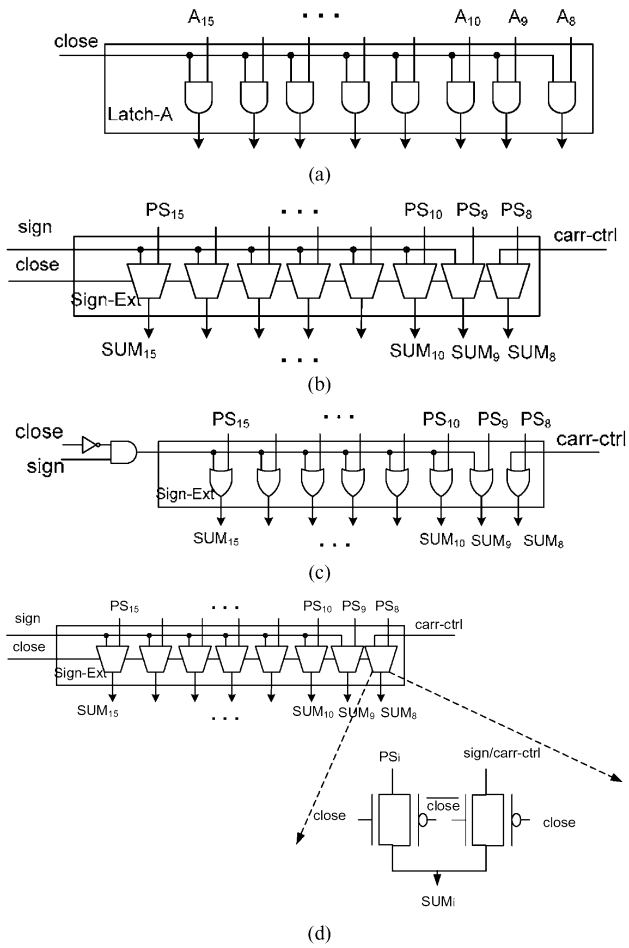


Fig. 7. Data-controlling components of the SPST where PS denotes pseudo-summations as shown in Fig. 5. (a) Data-latch design. (b) SE circuits. (c) SE circuits designed using OR gates. (d) SE circuits designed using CPL.

designs [16], [17], comprising several invertors and some capacitors, can be used to generate a proper delay to control the "*close*," "*sign*," and "*carr-ctrl*" signals.

Fig. 7 shows the data-controlling components of the SPST, where Fig. 7(a) shows the design of the data latch. The SE circuits can be intuitively implemented by multiplexers to compensate for the sign signals of the MSP, as shown in Fig. 7(b). The input data of the SE circuits are pseudosummations (PS) from the MSP adder/subtractors. In this paper, we further explore two more approaches besides using multiplexers to optimize the SE circuits. One approach uses simple OR gates, as shown in Fig. 7(c). The other adopts Complementary Pass-transistor Logics (CPLs) [18], as shown in Fig. 7(d). Both of these approaches can help realize the needed SE circuits. From the performance and overhead comparisons, fully discussed in Section IV, we decide to adopt the CPL circuits in our design.

III. Design Examples of SPST

Two multimedia/DSP design examples using the proposed SPST are introduced in detail as follows.

A. Design Example I—An Efficient H.264 Multitransform Coding Design (ETD)

The first design example using the proposed SPST is the ETD, as shown in Fig. 8. The ETD possesses two types of PEs. From the algorithmic view point, the four PEs in the left-hand side of Fig. 8, denoted by MPE-Is, are used for computing the 1-D transform. Meanwhile, the two PEs on the right-hand side of Fig. 8, denoted by MPE-IIs, are used for computing the 2-D transform. The ETD possesses a throughput of eight pixels per cycle so that it can perform 720p HD, 1080i HD, and digital cinema video formats at 22, 50, and 100 MHz, respectively (more details of the algorithm derivation and design exploration of ETD can be found in [13]). Based on the previous data analysis, we decide to divide the 15-bit SPST arithmetic units of the MPE-Is into 8-bit MSP and 7-bit LSP and divide the 17-bit SPST arithmetic units of the MPE-IIs into 8-bit MSP and 9-bit LSP, respectively, as shown in the fraction values near the SPST arithmetic units as shown in Fig. 8 where the SPST arithmetic units are marked with dotted shadows. The adder/subtractors located on the second and the fourth stages of the ETD are kept unchanged because most of the spurious signals existed in the data flowing out of the first and the third stages have been filtered out by the SPST. Therefore, it is useless to replace these

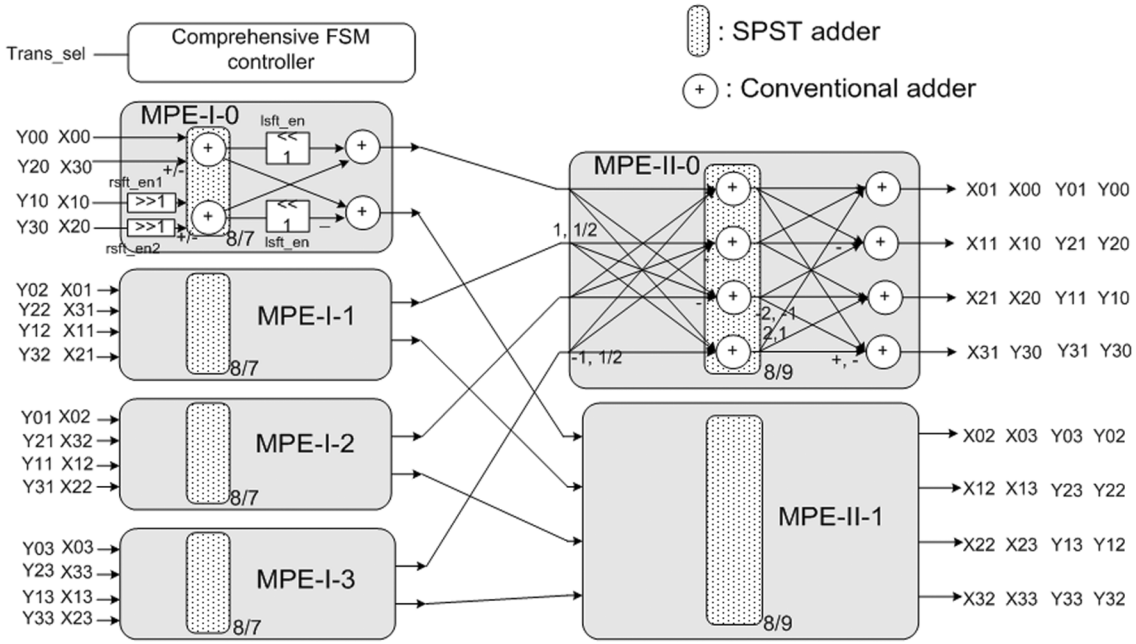


Fig. 8. Applications of the proposed SPST on the ETD for H.264, where the dotted shadows mark the locations of the SPST adders/subtractors and the nearing fraction values denote the bit widths of the MSP and LSP of each SPST adder/subtractor, respectively, in numerator and denominator.

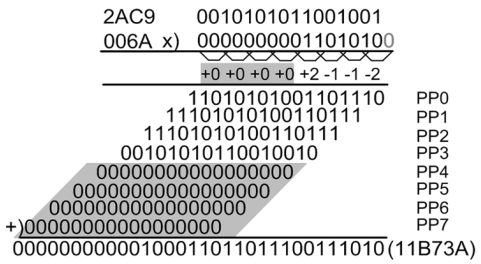


Fig. 9. Illustration of multiplication using modified Booth encoding.

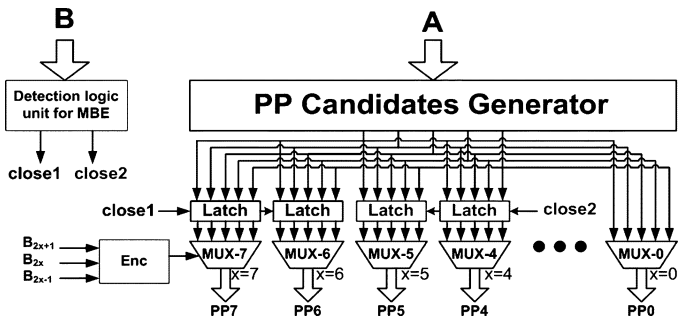


Fig. 10. SPST modified Booth encoder.

adder/subtractors located on the second and the fourth stages with the SPST ones.

B. Design Example II—A VMFU

The second design example using the proposed SPST is the VMFU, which is constructed on the basis of a modified Booth encoding multiplier. The proposed VMFU can compute six kinds of arithmetic operations, i.e., addition, subtraction, multiplication, MAC, interpolation, and SAD, which are frequently used in multimedia/DSP computations. There are three distinguishing design considerations in designing the VMFU, as listed as follows.

1) Applying the SPST to the Modified Booth Encoder: Fig. 9 shows a computation example of Booth multiplying two numbers "$2\mathrm{AC9}_{\mathrm{hex}}$" and "$006\mathrm{A}_{\mathrm{hex}}$," where the shadow denotes that the numbers in this part of Booth multiplication are all zeros so that this part of the computations can be neglected. Saving those computations can significantly reduce the power consumption caused by the transient signals. According to the analysis of the multiplication shown in Fig. 9, we propose the SPST modified Booth encoder which includes a detection unit, as shown in Fig. 10. From one of the two operands, e.g., the operand A, the partial product (PP) candidate generator generates five candidates of the partial products, i.e., $\{-2\mathrm{A}, -\mathrm{A}, 0, \mathrm{A}, 2\mathrm{A}\}$, which are then selected according to the Booth encoding results of the other operand, i.e., the operand B. Meanwhile, the detection unit has the second one of the two operands, i.e., the operand B in this case, as its input to decide whether the Booth encoder includes redundant computations. As shown in Fig. 10, the latches can, respectively, freeze the inputs of MUX-4 to MUX-7 or only those of MUX-6 to MUX-7 when the PP4 to PP7 or only the PP6 to PP7 are zeros to reduce the transition power dissipation. Such cases occur frequently in wireless multimedia-data coding like texture coding, orthogonal frequency-division multiplexing, and filter designs.

2) Applying the SPST to the Compression Tree: Fig. 11 shows the architecture diagram for the proposed VMFU in which the SPST modified Booth encoder has been shown in Fig. 10. The VMFU can be roughly decomposed into three sections, i.e., the *Partial Product Generation*, the *Partial Product Reduction* (PPR), and the *Accumulation* (ACC) sections. When the operand besides the Booth encoded one has a small absolute

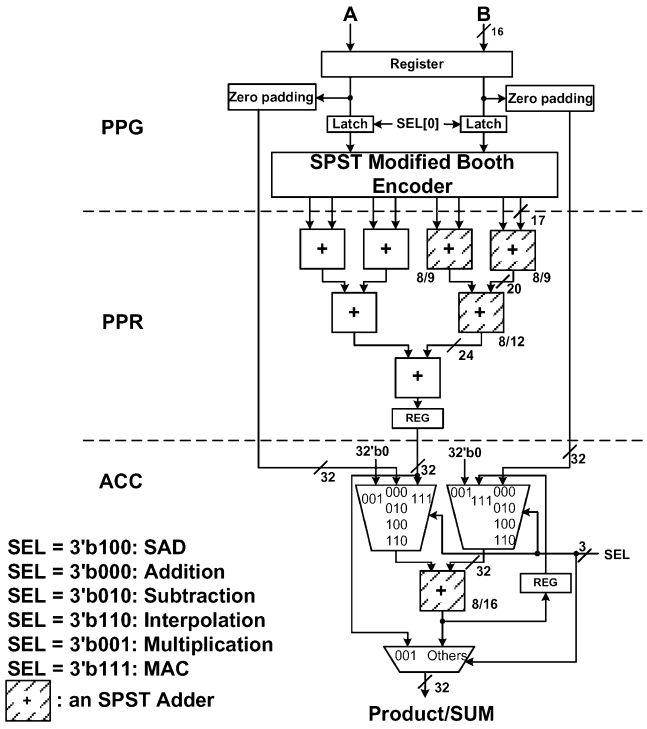


Fig. 11. Proposed low-power SPST-equipped VMFU.

value, there are opportunities to reduce the spurious power dissipated in the compression tree in the PPR section. According to the analysis of the addition shown in Figs. 3 and 9, we replace some of the adders in the compression tree of the VMFU, which add up the PP0 to PP3, with the SPST-equipped adders. Besides, the adder in the ACC section is also replaced with the SPST-equipped adder. This adder is used to accumulate the multiplication results in the MAC operation and compute the interpolation, SAD, addition, and subtraction. These adders are marked with oblique lines as shown in Fig. 11 with their bit widths of the MSP and LSP indicated, respectively, in the numerator and the denominator of the nearing fraction values.

3) Freezing the Switching Activities of the Unused Circuits: The data flows of the VMFU are controlled by the properly arranged multiplexers. By freezing the unused circuits in response to the selection of a certain kind of functionality, wasteful switching power dissipation can be avoided, as shown in Fig. 11. Besides, the circuits induced by the SPST can be frozen to turn off the SPST function of the VMFU. This option may be useful when the input data do not possess advantageous features like the multimedia data because the SPST may not contribute positive power saving when both the input data are random.

IV. Implementation and Verification

The two SPST design examples have been realized by following the standard IC/IP design flow with an in-house 0.18-μm CMOS cell library which is constructed following the TSMC 1P6M 0.18-μm CMOS technology. The two design examples are verified via C/MATLAB behavioral simulation, *nLint* HDL coding rule check, *VERILOG* simulation, *SYNOPSYS* logic synthesis, *Silicon Encounter* placement and routing, design rule check, layout versus schematic, and *NanoSim* postlayout simulation before chip fabrication.

TABLE I
Simulation Results of the SPST Using Three Implementing Approaches for SE Circuits, Where the Power Reduction Denotes the Percentage of the Power Saved by the SPST for the ETD

	Power Reduction (PR) (%)	Area (A, #tr.)	Magnitude order of PR/A
MUX	28.71	38572	3
OR	28.62	38376	2
CPL	29.69	37964	1

The three approaches for implementing the SE circuits of the SPST, described in Section II, are discussed in Table I based on the simulation results using *NanoSim*. The results in Table I are obtained by operating the proposed SPST-equipped ETD at 22 MHz with 1.8-V supply for real-time processing of the 720p HD video format. Other simulation results of the SPST-equipped ETD when targeting at 1080i HD (50 MHz) and digital cinema (100 MHz) video formats illustrate the same trend. Table I reveals that adopting the CPL circuits can minimize both the power dissipation and the area costs of the SPST-equipped ETD; thus, achieving the highest power reduction per area cost. The reason is that the CPL circuits are more compact which lead to smaller capacitance. To adopt the CPL circuits in the cell-based design flow, we design the CPL cell in a full-custom manner and place it in the in-house cell library for reuse. Therefore, the proposed design can be automatically placed and routed using EDA tools.

Moreover, searching for the minimum of the timing and area product, we can find that the most optimized synthesis results of the proposed SPST-equipped ETD can be obtained by using the 10-ns timing constraint. The *SYNOPSYS* synthesis results show that the gate count of the ETD without equipping the SPST is 6482. When the SPST adder/subtractors are equipped, the gate count increases to 7839. The increased hardware cost, which is about 20% of the original costs, results from the detection-logic units, the latches, and the SE circuits of MSP. Moreover, Table II lists the physical implementation results of the proposed SPST-equipped ETD chip. Using voltage scaling, the proposed SPST-equipped ETD consumes only $2.64 * 10^{-2}$, $5.82 * 10^{-2}$, and 0.24 mW/MHz for the 720p HD, 1080i HD, and digital cinema video format, respectively. The chip micrograph of the SPST-ETD is shown in Fig. 12 where the input scheduling module (ISM) and the output scheduling module (OSM) are used to reduce the I/O pin number and enable the at-speed test by rearranging the in/out data.

Furthermore, we consider three different configurations of compression tree, namely, the column-, row-, and row-based with carry-save-adders, for optimizing the VMFU. Using a 0.18-μm CMOS technology, the implementing results of the three configurations are listed in Table III. Because the row-based configuration consumes the lowest power dissipation and also the least power-delay product among the three configurations, it is adopted as the basis of the VMFU. Table IV shows the specifications of the SPST-equipped VMFU chip. The VMFU occupies a $418 * 426 - \mu m^2$ silicon area and has a

TABLE II
SPECIFICATIONS OF THE DESIGN EXAMPLE I—THE SPST-EQUIPPED ETD CHIP

SPST-equipped ETD chip		
Process	TSMC 1P6M 0.18-μm	
Supply voltage	0.7V ~ 1.8V	
Operating freq.	~ 100 MHz	
Throughput	8 pixels/cycle	
Power (mW/MHz)	$2.64*10^{-2}$@0.7V	for 720p HD
	0.25@1.8V	
	$5.82*10^{-2}$@1.0V	for 1080i HD
	0.24@1.8V	
	0.24@1.8V	for digital cinema
Processing capability	4096*2048 @ 30Hz (Digital cinema)	
Core area	468*468 (μm^2)	
Gate count	7839	
Tr. Count	31356	

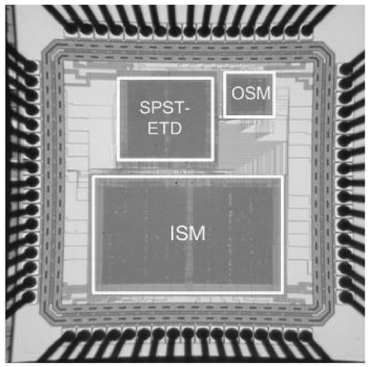


Fig. 12. Chip micrograph of the SPST-ETD where ISM and OSM, respectively, denote ISM and OSM.

TABLE III
PERFORMANCE EVALUATION AND COMPARISON OF THREE POPULAR BOOTH MULTIPLIERS IN TERMS OF DIFFERENT COMPRESSION TREE CONFIGURATIONS

	Column-based	Row-based	Row-based w/t CSA
Area (Tr.)	11722	10544	13524
Delay (ns)	5.5	5	5
Power/Freq. (mW/MHz)	0.0217	0.0201	0.0263
Norm. power-delay product	1	0.84	1.10

maximum operating speed of 125 MHz. The VMFU computes one datum per cycle for addition, subtraction, multiplication, and MAC and two data per cycle for interpolation and SAD. The VMFU consumes only 5.0×10^{-3} mW/MHz with 0.7 V for the multiplications in texture coding in terms of the "Foreman" sequence, or 3.2×10^{-4} mW/MHz with 0.4 V for the interpolation in terms of the "Foreman" sequence in H.264 encoders. Fig. 13 shows the micrograph of the SPST-VMFU chip where the ISM and the OSM have a similar function to those in the SPST-ETD chip.

TABLE IV
SPECIFICATIONS OF THE DESIGN EXAMPLE II—PROPOSED SPST-EQUIPPED VMFU WHERE TC DENOTES TEXTURE CODING AND ITP DENOTES INTERPOLATION

SPST-equipped VMFU chip		
Process	TSMC 1P6M 0.18-μm	
Supply voltage	0.4V~1.8V	
Operating freq.	~ 125 MHz	
Throughput	(1) 1 datum/cycle for addition, subtraction, multiplication, MAC (2) 2 data/cycle for interpolation, SAD	
Power (mW/MH)	Multiplications in H.264 TC of "Foreman" sequence	4.2×10^{-2} @1.8V
		5.0×10^{-3} @ 0.7V
	H.264 ITP of "Foreman" sequence	8.7×10^{-3} @1.8V
		3.2×10^{-4} @ 0.4V
Core area	418*426 (μm^2)	
Gate count	5232	
Tr. Count	20926	

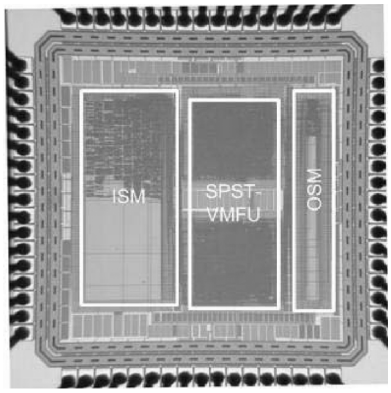


Fig. 13. Chip micrograph of the SPST-VMFU where ISM and OSM, respectively, denote ISM and OSM.

V. PERFORMANCE EVALUATION AND COMPARISON

Comparing the proposed SPST adder/subtractor designs with the existing precomputation techniques, we can start from two points of view. From the quantitative point of view, the performance comparison of the SPST ETD shall represent the trend of the performance comparison of the SPST adder/subtractor, since all the computations are simplified into additions/subtractions and shift operations in the integer transform of H.264. From the qualitative point of view, the concept of proposed SPST adder/subtractor is similar to the PGC design [7]. Both of these two techniques use a detection-logic unit to determine the effective ranges of the input data, a SE unit to compensate for the correct sign signals, and data latches to block the input data. However, the proposed SPST has a critical improvement in filtering out the glitches inside the adders/subtractors using only three 1-bit registers to control the assertion of the *close*, *sign*, and *carr-ctrl* signals, respectively. The realizing approaches of the data latches and the SE unit are also different from [7]. The designs in [8] and [9] present another realizing approach based on the similar concept with the PGC designs. Nevertheless, their applications on multipliers may not gain power saving because the input data flows are switched frequently when the

TABLE V
MEASURING RESULTS OF THE PROPOSED SPST-EQUIPPED ETD WITH 1.8-V SUPPLY VOLTAGE, WHERE "SPST OFF" DENOTES THAT THE SPST FUNCTION IS TURNED OFF AND "SPST ON" DENOTES THAT THE SPST FUNCTION IS TURNED ON

Freq. (MHz)	Applications	Power (P, mW)		Reduction of P. (%)
		SPST ON	SPST OFF	
22	HD 720p	5.49	7.57	27.48
50	HD 1080i	12.05	16.59	27.37
100	Digital cinema	24.18	33.25	27.28

input operands with a smaller effective dynamic range often exchange between operands A and B. This problem will not occur in this paper. The design in [12] presents two low-power techniques, i.e., the most significant bit (MSB) rejection (MSBR) and the row–column classification (RCC). The MSBR method, a kind of precomputation technique, uses a circuit to determine whether the most significant bits of an operand are the same or not. This circuit is conceptually analogous to the detection-logic unit in the PGC and SPST. The MSBR circuit outputs a mask signal to control the activation of the ROM and accumulator (RAC) unit in the DCT core using gated clocks. Hence, there is no SE unit in the MSBR approach. The RCC method, on the other hand, trades off image quality with power dissipation.

A. Evaluation and Comparison of the SPST-Equipped ETD

The measuring results of the proposed SPST-equipped ETD chip are listed in Table V, where "SPST off" and "SPST on," respectively, denote turning off and on the SPST function. By turning on the SPST function, 27.48%/27.37%/27.28% power reductions are, respectively, gained when the SPST-equipped ETD is operated at 22/50/100 MHz with 1.8-V supply voltage. This indicates that the proposed SPST technique can certainly reduce the power dissipation of transform coding for H.264. Table V compares the performance of the proposed SPST-equipped ETD with some existing H.264 transform designs [19]–[21]. The design in [19] offers a hardware implementation for the transform and quantization of H.264. This design adopts a parallel architecture to unfold the 2-D transform to avoid using a transpose memory; thus, it achieves a high throughput of 16 pixels per cycle. However, it is hard for the parallel-transform design to process in full speed because the input and output data are seldom transmitted in such a high-parallelism way due to practical constraints. The design in [20] presents an H.264 transform realization that contains two 1-D transform PEs and 16 registers served as the transpose memory with hardware cost of 6538 gates. This design has a rather low throughput of only four pixels per cycle. Furthermore, the design in [21] uses only one 1-D transform PE, in a recursive manner, and 16 registers, served as the transpose memory, with the hardware cost of 3524 gates to realize the transform coding of H.264. The throughput of this design is even lower as one pixel per cycle. In both the design in [20] and [21], the transpose register array occupies a crucial portion of the total hardware expense. In Table VI, we adopt an index named *data Throughput rate Per Unit Area* (TPUA), which is computed from dividing the throughput rate of each design listed in the fourth column by the corresponding area cost listed in the second column, to objectively evaluate the hardware efficiency of the SPST-equipped ETD. The TPUA index shows that the proposed design is at least 1.6 times more efficient than the existing designs.

TABLE VI
HARDWARE EFFICIENCY COMPARISONS OF THE SPST-EQUIPPED ETD AND THE EXISTING H.264 TRANSFORM DESIGNS. "TPUA" STANDS FOR DATA THROUGHPUT RATE PER UNIT AREA

Designs	Area (gate)	Freq. (MHz)	Throughput (M pixels /sec)	TPUA (pixels/sec/gate)
Kordasiewicz [20]	77280	107	1712	22.15 k
Wang [21]	6538	80	320	48.94 k
Wang [21] redesigned	6274	100	400	63.76 k
Liu [22]	3524	129	129	36.61 k
SPST-equipped ETD	7839	100	800	102.05 k

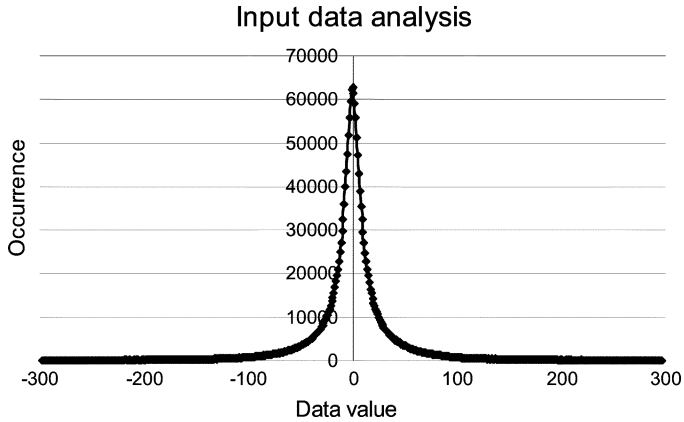


Fig. 14. Data analysis of the input data of the texture coding in H.264 where the video sequence is "Foreman," the quantization step is 20, and the total data number is over two million.

B. Evaluation and Comparison of the SPST-Equipped VMFU

The data analysis of the input data of multiplications in H.264 texture coding is shown in Fig. 14. The video sequence is "Foreman," and the quantization step is 20. The total number of the input data is more than two million. From the quantitative analysis of Fig. 14, we can tell that 95% of the input data are concentrated in the range between -95 and $+95$ among the total range between -2^{14} and $+2^{14}$[13]. The improvement in applying the SPST on the VMFU, classified into two phases, is shown in Fig. 15. In the first phase, the SPST is applied only on the compression tree of the VMFU which results in an 8% power reduction at the cost of a 9% area increment. In the second phase, the SPST is further applied on the modified Booth encoder which consumes considerable power dissipation, enlarging the power reduction to 24%. In addition, Fig. 16 shows the power results of the VMFU without and with the SPST, respectively. On the VMFU, the SPST can achieve a 24% power reduction at the cost of only a 10% area increment when compared with the original VMFU without equipping the SPST. This reveals that the SPST can save obvious power consumption of the VMFU at a rather tiny area cost, and we

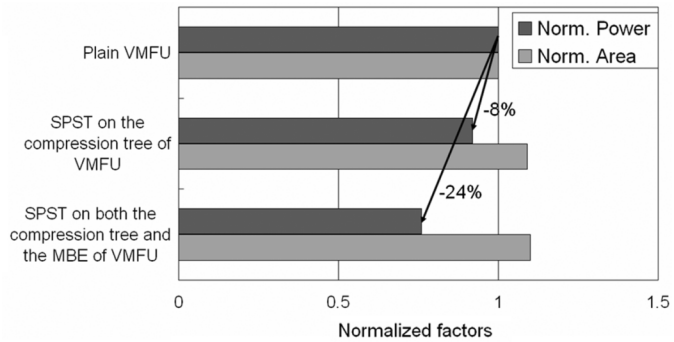


Fig. 15. Power improvements using the SPST.

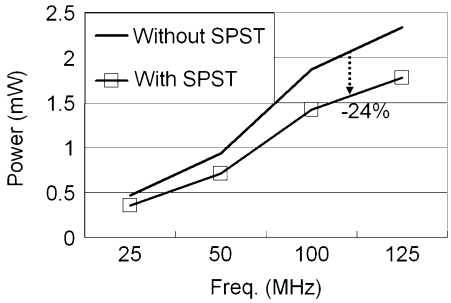


Fig. 16. Power reduction of the SPST for the VMFU.

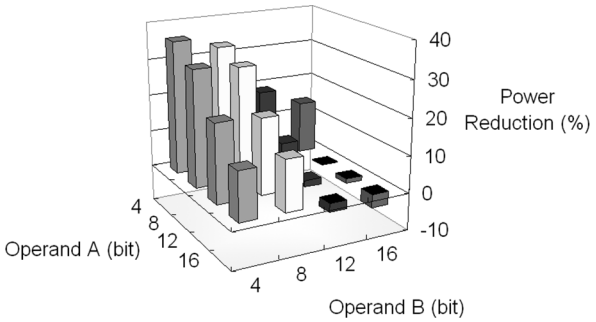


Fig. 17. Power reduction of the proposed SPST on multiplication A∗B for different effective ranges of *Normal distribution* of the operands using VMFU.

may extend the applications of the SPST to other circuits by finding a niche for the SPST in the target circuit.

Besides measuring the power dissipation in terms of multimedia applications, we provide the power analysis in terms of the inputs of different effective ranges using the *normal distribution*. Fig. 17 shows the power reduction of the proposed SPST on multiplication for different effective ranges, i.e., 4-, 8-, 12-, and 16-bit, of the operands using the VMFU. The figure reveals the following conditions.

1) The power reduction can reach up to 31%–36% if both the effective data ranges of operands A and B are smaller than or equal to 8 bit, by using the proposed SPST.
2) When the effective data range of operand B is smaller than or equal to 8 bit and that of operand A is 12 or 16 bit, the SPST also obviously saves power consumption for VMFU, e.g., around 21% or 13% power savings for 12- or 16-bit wide operand A, respectively. This achievement mainly results from operand B being Booth encoded and the proposed SPST being able to reduce the power dissipation of the Booth encoder apparently.
3) Moreover, the proposed SPST can reduce the power consumed in the compression tree besides the Booth encoder. When operand B is 12-bit wide, the power reduction is 19% or 8%, respectively, for the 4- or 8-bit wide operand A. Additionally, there is a 14% power saving in the case of multiplying 4-bit wide operand A with 16-bit wide operand B.

Furthermore, this paper examines some modern multiplier or MAC designs [22]–[29] because the basic architecture of VMFU is similar to them. The design in [22] includes three techniques, i.e., the signal flow optimization (SFO), left-to-right leapfrog (LRLF) structure, and upper/lower split structure, to optimize the array multipliers. The SFO and LRLF techniques are used for signal balancing of the PPR section in a multiplier. While the upper/lower split structure is used to shorten the path of the PPR section to prevent the snowballing glitch effect. The design [23] turns off some columns in the multiplier array whenever their outputs are known, thus, saving 10% power consumption at the cost of 20% area overhead under a 0.35-μm CMOS technology. The design [24] uses a DRD unit to detect the dynamic range of the inputs and adopts three separate Wallace trees for the 4×4, 8×8, and 16×16 multiplications. Both methods adopted in the design in [24] certainly increase the area and capacitance overheads. The design in [25] proposes a 32-bit single-instruction–multiple-data MAC unit which is a coprocessor to the Intel XScale microprocessor. Under a 0.18-μm CMOS technology, [25] dissipates 450 mW at 600 MHz with 1.3-V supply voltage. The design in [26] is a vector MAC unit that can perform one 64×64, two 32×32, four 16×16, or eight 8×8 signed/unsigned multiply accumulations and is fabricated using the 0.13-μm bulk and 90-nm silicon-on-insulator silicon technology. The design in [27] involves a fixed-width 32-bit left-to-right multiplier which obtains 8% speed improvement, 14% power reduction, and 13% area saving. Meanwhile, the design in [28] explores a design methodology for high-speed modified Booth multipliers. Finally, design in [29] adopts an advanced 90-nm dual $V_{\rm t}$ CMOS technology to implement a 16×16-bit multiplier that consumes 9 mW at 1 GHz with 1.3 V.

Table VII lists the performance of some representative existing multipliers and the proposed SPST-VMFU. For the sake of a fairer comparison in a time-economic way, we scale the power results to the same technology, i.e., the 0.18-μm technology with 1.8 V, according to the geometric feature and the well-known power approximation $P = \sum_i \alpha_i C V^2 f$[4] and [30], where P, α_i, C, V, and f denote power dissipation, switching activity, capacitance, voltage, and frequency, respectively. Under the situation that all the multiplier designs including the compared ones and the proposed one can meet the performance requirements, the factor of frequency, i.e., f, of all the designs can be assumed to be the same to avoid the frequency-scaling effects. Besides, switching activity, i.e., α_i, is also the same when all the designs are computing for the same applications. Thus, power consumption P is proportional to the product of capacitance and supply voltage only. The capacitance is assumed to be proportional to the silicon area which is proportional to the square of the linewidth in the technology scaling. Therefore, we can obtain the scaling

TABLE VII
PERFORMANCE COMPARISON OF EXISTING MULTIPLIERS AND THE PROPOSED VMFU

Design	Feature	Tech.	Power (mW)	Delay (ns)	Area
Huang [22]	(1) 32b×32b	0.18-μm	(1) 19.65 for Djpeg (2) 40.65 for Random data @ 100MHz	7.25	74598 (tr.)
Liao [25]	(1) Coprocessor (2) SIMD (3) 32b MAC	0.18-μm	900@1.6V, 800MHz	1.25	N.A.
Wang [27]	(1) 32b×32b (2) Fixed-width	0.35-μm /3.3V	79.86	14.01	19743 (gate)
Chen [9]	(1) 16b×16b	0.25-μm	17.30 for *Normal distribution* inputs	8.3	0.337 (mm^2)
Lee [24]	Scalable length of 4b, 8b, 16b	0.13-μm /1.2V	1.04@100MHz for random data	N.A.	6388 (gate)
Hsu [29]	(1) 16b×16b (2) Sleep mode (3) Duel V_T	90nm	(1) 9@1.3V, 1GHz (1) 7.9×10^{-2} @50MHz, 0.57V	1	0.03 (mm^2)
Proposed	(1) 16b×16b (2) Versatile functions	0.18-μm	(1) 4.2@100MHz,1.8V (2)1.25×10^{-1}@25MHz, 0.7V for H.264 IQ (3) 6.3@125MHz,1.8V for *Normal distribution* inputs	8	5232 (gate) or 0.178 (mm^2)

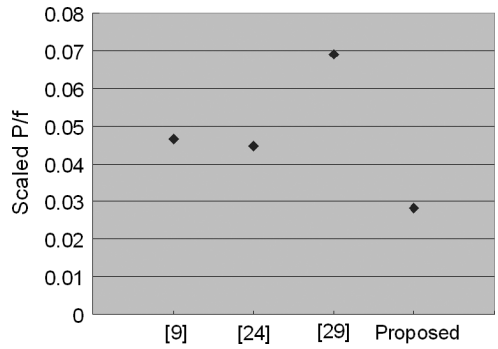


Fig. 18. Power comparison of the proposed VMFU.

formulation $P_{\rm sca} = (P_{\rm org})/((\mathrm{Tech}_{\rm org}/0.18)^2 * (V_{\rm org}/1.8)^2)$, where $P_{\rm sca}, P_{\rm org}$, $\mathrm{Tech}_{\rm org}$, and $V_{\rm org}$ denote the scaled power dissipation, original power dissipation, linewidth in the original technology, and the original voltage, respectively. Similar scaling approaches are adopted in the design in [31]. The scaled results in terms of power per unit frequency are shown in Fig. 18. Because the designs in [22], [25], and [27] are basically 32-bit multipliers, they are considered as reference indexes only. From Fig. 18, we can find that the proposed VMFU has better power efficiency than the existing multipliers although it involves the augmented circuits for versatile functions.

VI. CONCLUSION

This paper proposes a low-power technique called SPST and explores its applications in multimedia/DSP computations, where the theoretical analysis and the realization issues of the SPST are fully discussed. The proposed SPST can obviously decrease the switching (or dynamic) power dissipation, which comprises a significant portion of the whole power dissipation in integrated circuits. When applied to the H.264 multitransform coding design (ETD), the proposed SPST can save 27% power consumption at the cost of only 20% area overheads. Besides, the proposed SPST can achieve a 24% saving in power consumption at the expense of only 10% area overheads for the proposed VMFU. Both the SPST-equipped ETD and the VMFU are verified in detail and physically implemented on chips using the 0.18-μm CMOS technology. The performance comparisons also illustrate that the SPST-equipped designs are very competitive with the existing designs. Furthermore, the proposed SPST is a fully static CMOS circuit technique which does not aggravate the problems of leakage power, signal racing, and voltage dropping.

REFERENCES

[1] T. Stockhammer, M. Hannuksela, and T. Wiegand, "H.264/AVC in wireless environments," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 657–673, Jul. 2003.
[2] R. Schafer, T. Wiegand, and H. Schwarz, "The emerging H.264/AVC standard," EBU Technique Review Jan. 2003 [Online]. Available: http://www.ebu.ch/trev_293-schaefer.pdf
[3] A. Bellaouar and M. I. Elmasry, *Low-Power Digital VLSI Design"Circuits and Systems*. Norwell, MA: Kluwer, 1995.
[4] A. P. Chandrakasan and R. W. Brodersen, "Minimizing power consumption in digital CMOS circuits," *Proc. IEEE*, vol. 83, no. 4, pp. 498–523, Apr. 1995.
[5] K. K. Parhi, "Approaches to low-power implementations of DSP systems," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 48, no. 10, pp. 1214–1224, Oct. 2001.
[6] K. Choi, R. Soma, and M. Pedram, "Dynamic voltage and frequency scaling based on workload decomposition," in *Proc. IEEE Int. Symp. Low Power Electron. Des.*, 2004, pp. 174–179.
[7] J. Choi, J. Jeon, and K. Choi, "Power minimization of functional units by partially guarded computation," in *Proc. IEEE Int. Symp. Low Power Electron. Des.*, 2000, pp. 131–136.
[8] O. Chen, R. Sheen, and S. Wang, "A low-power adder operating on effective dynamic data ranges," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 10, no. 4, pp. 435–453, Aug. 2002.
[9] O. Chen, S. Wang, and Y. W. Wu, "Minimization of switching activities of partial products for designing low-power multipliers," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 11, no. 3, pp. 418–433, Jun. 2003.
[10] L. Benini, G. D. Micheli, A. Macii, E. Macii, M. Poncino, and R. Scarsi, "Glitch power minimization by selective gate freezing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 8, no. 3, pp. 287–298, Jun. 2000.
[11] S. Henzler, G. Georgakos, J. Berthold, and D. Schmitt-Landsiedel, "Fast power-efficient circuit-block switch-off scheme," *Electron. Lett.*, vol. 40, no. 2, pp. 103–104, Jan. 2004.
[12] T. Xanthopoulos and A. P. Chandrakasan, "A low-power DCT core using adaptive bitwidth and arithmetic activity exploiting signal correlations and quantization," *IEEE J. Solid-State Circuits*, vol. 35, no. 5, pp. 740–750, May 2000.
[13] K. H. Chen, J. I. Guo, and J. S. Wang, "A high-performance direct 2-D transform coding IP design for MPEG-4 AVC/H.264," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 4, pp. 472–483, Apr. 2006.
[14] K. H. Chen, K. C. Chao, J. I. Guo, J. S. Wang, and Y. S. Chu, "Design exploration of a spurious power suppression technique (SPST) and its applications," in *Proc. IEEE Asian Solid-State Circuits Conf.*, Hsinchu, Taiwan, Nov. 2005, pp. 341–344.
[15] K. H. Chen, Y. M. Chen, and Y. S. Chu, "A versatile multimedia functional unit design using the spurious power suppression technique," in *Proc. IEEE Asian Solid-State Circuits Conf.*, Hangzhou, China, Nov. 2006, pp. 111–114.
[16] H. H. Chang, S. H. Sun, and S. I. Liu, "A low-jitter and precise multiphase delay-locked loop using shifted averaging VCDL," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2003, vol. 1, pp. 434–505.
[17] Y. J. Jung, S. W. Lee, D. Shim, W. Kim, C. Kim, and S. I. Cho, "A dual-loop delay-locked loop using multiple voltage-controlled delay lines," *IEEE J. Solid-State Circuits*, vol. 36, no. 5, pp. 784–791, May 2001.

[18] K. Yano and T. Yamanaka, "A 3.8-ns CMOS 16 × 16-b multiplier using complementary pass-transistor logic," *IEEE J. Solid-State Circuits*, vol. 25, no. 2, pp. 388–395, Apr. 1990.
[19] R. Kordasiewicz and S. Shirani, "Hardware implementation of the optimized transform and quantization blocks of H.264," in *Proc. IEEE Can. Conf. Elect. Comput. Eng.*, May 2004, vol. 2, pp. 943–946.
[20] T. C. Wang, Y. W. Huang, H. C. Fang, and L. G. Chen, "Parallel 4 × 4 2-D transform and inverse transform architecture for MPEG-4 AVC/H.264," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2003, pp. 800–803.
[21] L. Z. Liu, Q. Lin, M. T. Rong, and J. Li, "A 2-D forward/inverse integer transform processor of H.264 based on highly-parallel architecture," in *Proc. IEEE Int. Workshop System-on-Chip Real-Time Appl.*, Jul. 2004, pp. 158–161.
[22] Z. Huang and M. D. Ercegovac, "High-performance low-power left-to-right array multiplier design," *IEEE Trans. Comput.*, vol. 54, no. 3, pp. 272–283, Mar. 2005.
[23] M. C. Wen, S. J. Wang, and Y. N. Lin, "Low-power parallel multiplier with column bypassing," *Electron. Lett.*, vol. 41, no. 12, pp. 581–583, May 2005.
[24] H. Lee, "A power-aware scalable pipelined Booth multiplier," in *Proc. IEEE Int. SOC Conf.*, Sep. 2004, pp. 123–126.
[25] Y. Liao and D. B. Roberts, "A high-performance and low-power 32-bit multiply-accumulate unit with single-instruction–multiple-data (SIMD) feature," *IEEE J. Solid-State Circuits*, vol. 37, no. 7, pp. 926–931, Jul. 2002.
[26] A. Danysh and D. Tan, "Architecture and implementation of a vector/SIMD multiply-accumulate unit," *IEEE Trans. Comput.*, vol. 54, no. 3, pp. 284–293, Mar. 2005.
[27] J. S. Wang, C. N. Kuo, and T. H. Yang, "Low-power fixed-width array multipliers," in *Proc. IEEE Symp. Low Power Electron. Des.*, Aug. 9–11, 2004, pp. 307–312.
[28] W. C. Yeh and C. W. Jen, "High-speed Booth encoded parallel multiplier design," *IEEE Trans. Comput.*, vol. 49, no. 7, pp. 692–701, Jul. 2000.
[29] S. K. Hsu, S. K. Mathew, M. A. Anders, B. R. Zeydel, V. G. Oklobdzija, R. K. Krishnamurthy, and S. Y. Borkar, "A 110 GOPS/W 16-bit multiplier and reconfigurable PLA loop in 90-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 256–264, Jan. 2006.
[30] Z. L. He, C. Y. Tsui, K. K. Chan, and M. L. Liou, "Low-power VLSI design for motion estimation using adaptive pixel truncation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 5, pp. 669–678, Aug. 2000.
[31] B. M. Baas, "A low-power, high-performance, 1024-point FFT processor," *IEEE J. Solid-State Circuits*, vol. 34, no. 3, pp. 380–387, Mar. 1999.

Kuan-Hung Chen (M'07) received the B.S. and M.S. degrees in engineering from National Cheng Kung University, Tainan, Taiwan, in 1998 and 2000, respectively, and the Ph.D. degree in electrical engineering from National Chung Cheng University, Chiayi, Taiwan, in 2006.

Since February 2007, he has been with the Department of Electronic Engineering, Feng Chia University, Taichung, Taiwan, where he is currently an Assistant Professor. His research interests include video/audio-processing algorithms, digital-signal processing, very large scale integration architecture design, digital IP design, and system-on-chip design.

Dr. Chen was the recipient of the Second Prize Award of the Silicon Intellectual Property Design Contest in 2004 and 2005, respectively.

Yuan-Sun Chu (M'05) received the B.S. degree in electrical engineering from Feng-Chia University, Taichung, Taiwan, in 1978 and the M.S. and Ph.D. degrees in electrical engineering from Katholieke Universiteit Leuven, Leuven, Belgium, in 1986 and 1991, respectively.

Since 1991, he has been an Associate Professor with the Department of Electrical Engineering, National Chung Cheng University, Chiayi, Taiwan. His research interest includes communication ICs, QoS engine for Internet, and low-power design.