Device Threshold Based CMOS Differential Logic Style for Reducing Power Delay Product

S. Hari

M.E VLSI DESIGN

SRI ESHWAR COLLEGE OF ENGINEERING

ABSTRACT

Growing need for portable devices has provoked ever increasing concern on energy-efficient design. Since the energy consumption of modern digital CMOS circuits has been traditionally dominated by switching energy having quadratic dependence on supply voltage, voltage scaling is an effective way to minimize the overall energy consumption of system-on-chip .This CMOS differential logic style is operating with the MOS threshold voltage. It can also improve the switching speed by boosting the gate–source voltage of transistors along timing-critical signal paths and minimizes area overhead by allowing a single boosting circuit to be shared by complementary outputs. Test sets of logic gates were designed in a 90 nm CMOS process, whose comparison results indicated that the Power–delay product of the proposed logic style was improved when compared with the conventional logic styles at supply voltage ranging from 0.4 to 1.2 V. The experimental result for a 64-bit adder designed using the proposed logic style is mainly applicable for the filter applications. The computational power required for the filter functions can be reduced using the proposed logic style. It can be implemented in the adder block of FIR filter operations, so that the performance factor of the FIR filter can be further improved.

Index Terms—Adder, low power, low voltage, voltage boosting.

I.INTRODUCTION

In the extreme case, circuits can be made to be operated in the sub-threshold region for maximum energy efficiency. However, the approach is limited to be only used in a low-end design, where speed is a secondary concern, because of severe speed degradation due to small switching current and high performance variability due to process, temperature, and threshold voltage variations For medium- and high-end designs, where speed performance and

energy efficiency are both important, that much aggressive voltage scaling is not acceptable, and instead, a near-threshold voltage design is more suitable for achieving relatively high energy efficiency without severe speed degradation. As the Supply voltage scales down toward threshold voltage, the speed performance of conventional CMOS circuits, such as static CMOS logic, differential cascode voltage switch (DCVS) logic in Fig. 1(a) and
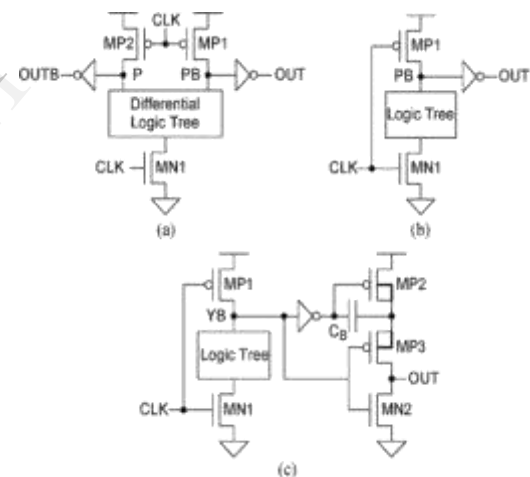

Fig. 1. Conventional digital CMOS circuits. (a) DCVS. (b) Domino CMOS logic. (c) BDL

Domino CMOS log*i*c. is still severely degraded due to the reduced overdrive voltage ($V_{GS} - V_{TH}$) of transistors. To overcome this problem, a bootstrapped CMOS large capacitive-load driver was proposed. It can improve the switching speed at low supply voltage by allowing the voltage of some internal nodes to be boosted beyond the supply rails. However, since the circuit was proposed for use as a large capacitive-load driver, logic functions cannot be efficiently embedded into the circuit. For fast logic operation at low supply voltage, CMOS bootstrapped

Dynamic logic (BDL) in Fig. 1(c) was proposed. However, the speed of this logic style was not so much improved since the latency of bulky bootstrapping circuit was superimposed on the overall latency of the circuit. Moreover, logic composition of this logic style is constrained since it is configured as a single-ended structure. Although some recent circuit techniques adopting bootstrapped operation have been proposed, they are all not for logic composition but for large capacitance driving .To overcome the aforementioned problems and to further improve the switching performance, a novel boosting CMOS differential logic style is proposed in this brief. Section II describes the circuit structure and operation of the proposed logic style. In Section III, comparison results for some representative logic gates are presented to assess the performance of the pro-posed logic style. Section IV describes the experimental result for 64-bit adders as a design example to prove the practicality of the proposed logic style. Finally, we present the conclusion in Section V.

(a)

(b)

Fig. 2. Proposed BCDL. (a) Structure. (b) Simulated waveforms

II. CIRCUIT STRUCTURE AND OPERATION

Fig. 2(a) shows a generic structure of the proposed logic style i.e., boosted CMOS differential logic (BCDL). It consists of a pre-charged differential logic block and a voltage-boosting block. The voltage-boosting block, which is shown in the dotted box at the lower part of the circuit, is composed of transistors MN2, MN3, and MP3 and boosting capacitor C_{BOOT} and is used to boost the voltage of NP below the ground. The pre-charged differential logic block, which is composed of a differential logic tree with bottom transistor MN1, pre-charge transistors MP1 and MP2, and output inverters, receives the boosted voltage at NP and swiftly evaluates the output logic values.

Let us explain the operation of BCDL. It has two phases of operation, namely, a pre-charge phase and a boosted evaluation phase. The circuit is in the pre-charge phase when CLK is low. During this phase, the pre-charged differential logic block is separated from the voltage-boosting block since MN1 is fully off. Pre-charge nodes P and PB in the differential logic block are then pre-charged to the supply voltage by MP1 and MP2, letting outputs OUT and OUTB identically low. At the same time, transistors MP3 and MN2 in the voltage-boosting block turn on, allowing NS and NP to be high and low, respectively. Then, a voltage identical to the supply voltage is applied across C_{BOOT}. When CLK changes to high, the circuit goes into the boosted evaluation phase. The simulated waveforms of BCDL in this phase are shown in Fig. 2(b), where a supply voltage of 0.5 V is used. Since CLK goes high, MN1 turns on and connects the differential logic tree to the voltage-boosting block.
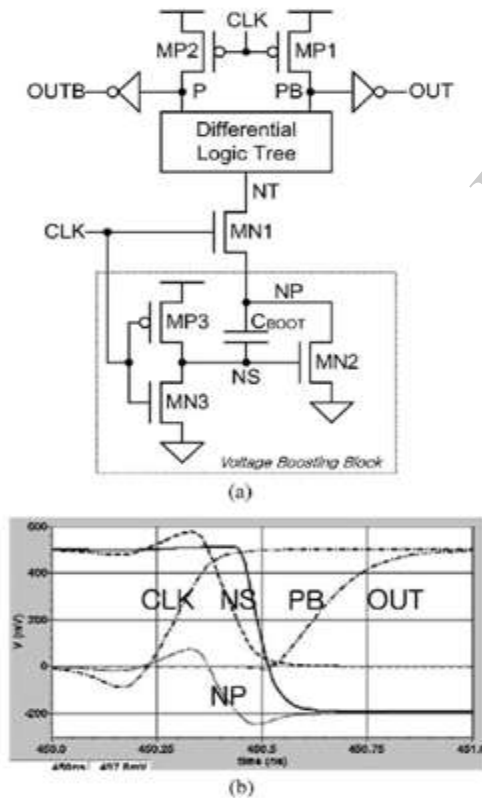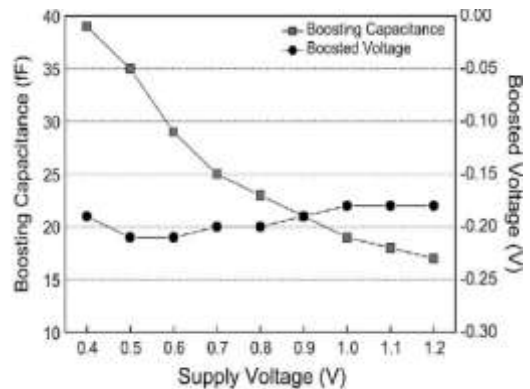

Fig.3. Two-input XOR/XNOR BCDL gate

At the same time, NS is pulled down toward the ground, allowing NP and NT to be boosted below the ground by capacitive coupling through C_{BOOT}. As shown in Fig. 2(b), NP temporally reaches -250 mV and settles at around -200 mV by the boosting action. Then, the gate–source voltages of MN1 and transistors that are on in the logic tree are enlarged, resulting in an increased driving strength of these transistors. Moreover, a slightly for-ward source–body voltage established in these transistors by boosting source voltages below the ground leads to a reduction in threshold voltages of these transistors, further increasing their driving strength. In turn, the boosted voltage at NT is then transferred to P or PB through the logic tree, depending on input data. In Fig. 2(b), input data are such that PB is pulled down below the ground. Then, the gate–source voltage of the driver PMOS transistor is also enlarged, enhancing its driving strength. All these driving strength-enhancing effects by boosting are combined together along the timing-critical signal paths from the inputs to the outputs via pre-charge nodes, resulting in significantly improved switching speed at a low-voltage region.

III. SIMULATION COMPARISON

To assess the performance of the proposed circuit technique, various multi-input logic gates are designed using the conventional and proposed logic styles in a 90 nm CMOS process. The nominal threshold voltages of p and n-channel MOS transistors are -0.45 and 0.42 V, respectively. The boosting capacitor was implemented using the gate-oxide capacitance of a PMOS transistor. Transistor widths in each logic gate and the amount of boosting capacitance were individually optimized at each supply voltage for each logic style to provide a minimum power–delay product (PDP).

Fig. 3 shows the resulting optimal boosting capacitance and the associated boosted voltage of a two-input XOR/XNOR BCDL gate for supply voltage ranging from 0.4 to 1.2 V. As shown in Fig. 3, a relatively constant boosted voltage around -200 mV was found to be optimal since it results in a minimum EDP with a good tradeoff between switching speed and noise margin. For a shallower boosted voltage, noise margin becomes better, but speed enhancement is not enough due to insufficient bootstrapping. For a deeper boosted voltage, speed becomes better, but noise margin is degraded, and energy efficiency becomes worse due to an increase in leakage. The size of the boosting capacitor then to generate the optimal boosted voltage increases as the supply voltage is lowered, as indicated in Fig. 3,

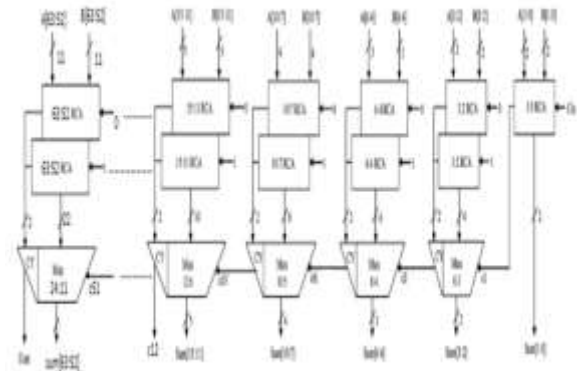since a larger-sized capacitor is required to get the same boosted voltage from a lower supply.

Fig. 4. Conventional adder-RCA diagram

As shown in Fig. the 8 bit ripple carry chain of a 64 bit BCDL adder is used. Conventional non-boosted logic gates, such as static CMOS, domino CMOS, and DCVS, have an increasing delay as the supply voltage approaches the device threshold. The BDL gate, although better than other conventional logic gates, provides not so much improvement on delay at low supply voltages due to increasing latency of bootstrapping circuit. The conventional 64 bit adder is used in our existing system design as shown in fig.4 and is shown in the following figure. The Ripple carry adder has the drawback of Propagation delay.so, we are using carry select adder instead of ripple carry adder for the efficient design. The logic values o and 1 are simultaneously used in our CSA design.
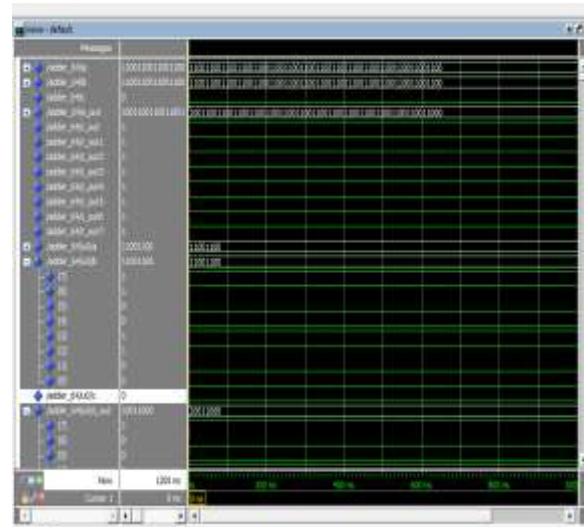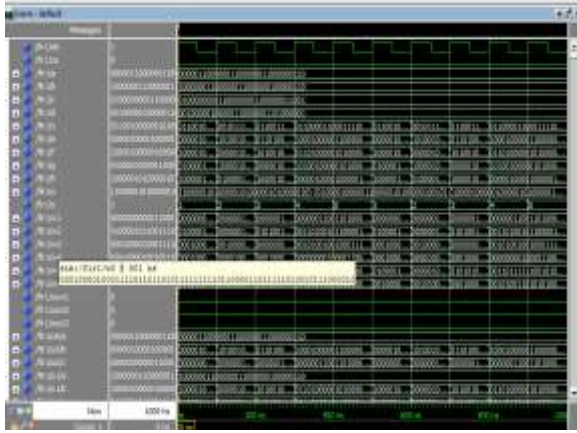
Fig.5 Simulated output of an conventional 64 bit adder

The simulated waveforms are shown in the fig.6 for our existing method.   Meanwhile, the proposed BCDL gate has a significantly reduced propagation delay compared with the static CMOS, domino CMOS, DCVS, and BDL gates, providing up to 89%, 82%, 83%, and 79% improvements, respectively. The BDL gate shows the worst performance. The BCDL gate, showing the lowest energy consumption above 0.9-V supply, becomes worse than the conventional logic gates, except for BDL under 0.5-V supply. This is attributed to the fact that a larger optimum-sized boosting capacitor was used at lower supply voltages, resulting in increased energy consumption for boosting operation.
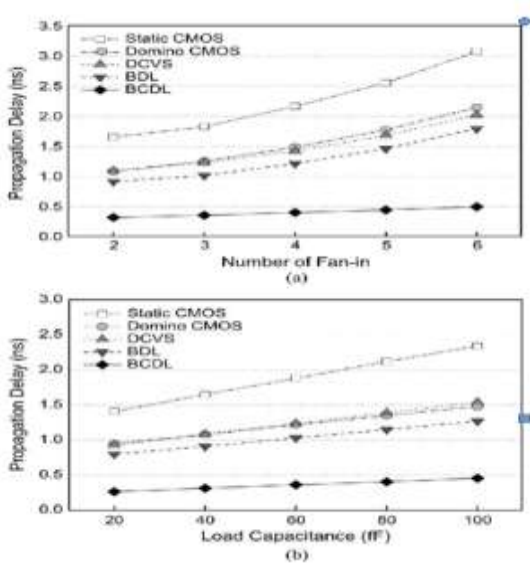

Fig 6. Simulated comparison of XOR/XNOR gates

The EDP performance is compared, the proposed BCDL gate is superior to conventional logic gates. Numerically, the BCDL gate provides up to 86%, 74%, 81%, and 86% improvements on EDP, as compared with static CMOS, domino CMOS, DCVS, and BDL gates at 0.4-V supply, respectively. The minimum EDP of the BCDL gate occurring at around 0.6 V is reduced by up to 60%, as compared with those of conventional logic gates occurring at higher supply voltages. Fig. 6 shows the propagation delay of XOR/XNOR gates depending on fan-in number and load capacitance at a supply voltage of 0.5 V. As shown in Fig. 6 BCDL gates provide up to an 84% improvement for the fan-in number ranging from 2 to 6, as compared with conventional logic gates. It shows the propagation delay of the logic gates versus load capacitance where the load capacitance ranges from 20 to 100 fF, where BCDL gates provide up to an 80% improvement. These results indicate that BCDL retains the advantage in terms of speed for increased logic complexity and load capacitance.

Table I summarizes the simulated performance of various logic gates designed with the conventional and proposed logic styles. The simulation was executed at 100-MHz frequency with 0.5-V supply. The propagation delay was measured at the worst case with 40-fF load capacitance. Clocking energy was also included in the measurement. As before, for single-ended logic styles, such as domino CMOS and BDL, only the non-inverting outputs are available. For AND/NAND and OR/NOR gates having the same input counts designed with differential logic styles, the numerical data are all the same since the structures of the logic gates are exactly identical to each other. For all the logic gate types, the delay of BCDL gates is smaller than those of conventional gates, showing up to an 81% improvement

Type of adder used	Power(mW)	Delay(nS)	PDP
Conventional	202	10.023	2024.6
BCDL	171	9.573	1636.9

Gate Type	Parameter	Static CMOS	Domino CMOS	DCVS	BDL	BCDL
2-Input AND/ NAND	Delay (ns)	1.18 (1.0)	0.81 (0.69)*	0.79 (0.67)	0.67 (0.57)	0.24 (0.20)
	Energy (fJ)	4.42 (1.0)	5.83 (1.32)	7.39 (1.67)	12.84 (2.90)	7.99 (1.81)
	EDP (10⁻²⁴ J•s)	5.22 (1.0)	4.72 (0.90)	5.84 (1.12)	8.60 (1.65)	1.92 (0.37)
2-Input OR/ NOR	Delay (ns)	1.25 (1.0)	0.78 (0.62)	0.79 (0.63)	0.62 (0.50)	0.24 (0.19)
	Energy (fJ)	5.35 (1.0)	5.06 (0.95)	7.39 (1.38)	11.69 (2.19)	7.99 (1.49)
	EDP (10⁻²⁴ J•s)	6.69 (1.0)	3.95 (0.59)	5.84 (0.87)	7.26 (1.08)	1.92 (0.29)
2-Input XOR/ XNOR	Delay (ns)	1.65 (1.0)	1.08 (0.65)	1.08 (0.66)	0.91 (0.55)	0.31 (0.19)
	Energy (fJ)	11.37 (1.0)	9.24 (0.81)	11.88 (1.04)	20.36 (1.79)	12.87 (1.13)
	EDP(10⁻²⁴ J•s)	18.76 (1.0)	9.98 (0.53)	12.88 (0.69)	18.52 (0.99)	4.04 (0.22)
3-Input AND/ NAND	Delay (ns)	1.41 (1.0)	0.92 (0.65)	0.91 (0.65)	0.75 (0.53)	0.28 (0.20)
	Energy (fJ)	10.08 (1.0)	10.31 (1.02)	12.67 (1.26)	19.23 (1.91)	11.59 (1.15)
	EDP (10⁻²⁴ J•s)	14.21 (1.0)	9.49 (0.67)	11.57 (0.81)	14.42 (1.01)	3.21 (0.23)
3-Input OR/ NOR	Delay (ns)	1.49 (1.0)	0.80 (0.54)	0.91 (0.61)	0.85 (0.44)	0.28 (0.18)
	Energy (fJ)	14.74 (1.0)	6.29 (0.43)	12.67 (0.86)	14.70 (1.0)	11.59 (0.79)
	EDP (10⁻²⁴ J•s)	21.96 (1.0)	5.03 (0.23)	11.57 (0.53)	9.56 (0.44)	3.21 (0.15)
3-Input XOR/ XNOR	Delay (ns)	1.82 (1.0)	1.25 (0.68)	1.23 (0.67)	1.01 (0.55)	0.35 (0.19)
	Energy (fJ)	31.41 (1.0)	15.06 (0.48)	21.74 (0.69)	35.92 (1.14)	23.81 (0.76)
	EDP (10⁻²⁴ J•s)	57.17 (1.0)	18.81 (0.33)	26.63 (0.47)	36.28 (0.63)	8.36 (0.15)

* Digit in parenthesis in each cell represents performance ratio with respect to static CMOS logic.

IV. EXPERIMENTAL RESULT

To demonstrate practical applicability of the proposed logic style, a set of 64-bit adders were designed. The 64-bit adder consisting of eight 8-bit adder subsections adopts the carry selection scheme for high-speed carry propagation. An 8-bit ripple carry chain was used in the BCDL adder to allow boosting operation at each Carry chain stage, whereas an 8-bit Manchester carry chain was used in the DCVS and BDL adders for high-speed carry propagation. Fig. 7 shows the structure of the 8-bit ripple carry chain used in the 64-bit BCDL adder
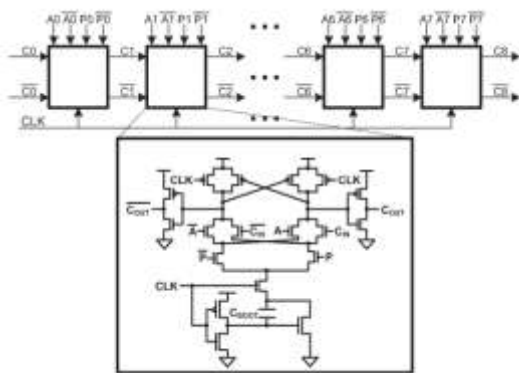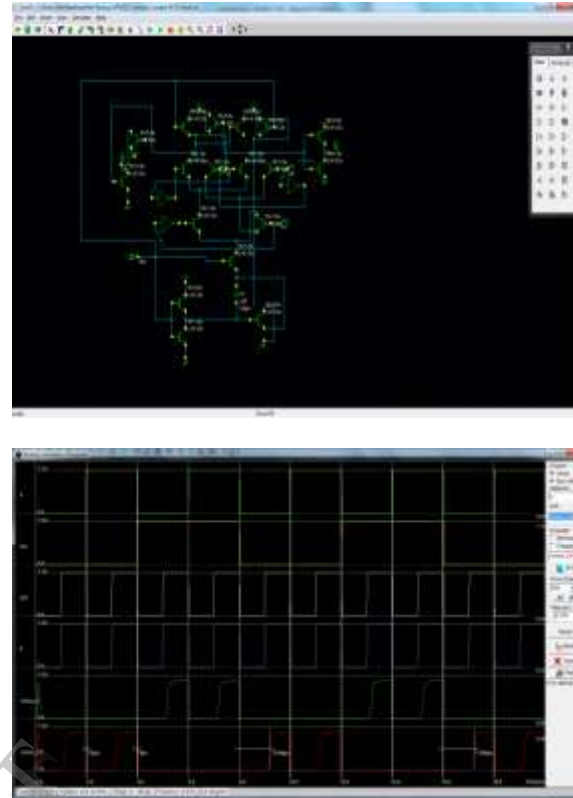

Fig.8 Simulated output of a 8 bit ripple carry chain in a 64 bit BCDL

In our proposed method, we are using Boosted CMOS Differential Logic (BCDL) style adder for our system design. It is shown in the following fig.9

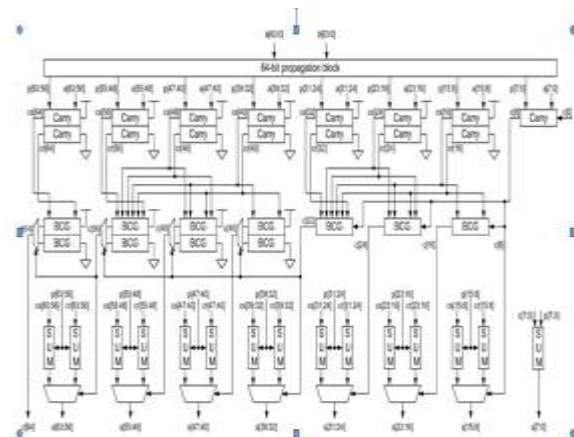Fig.7 Structure of an 8-bit ripple carry chain in a 64-bit BCDL
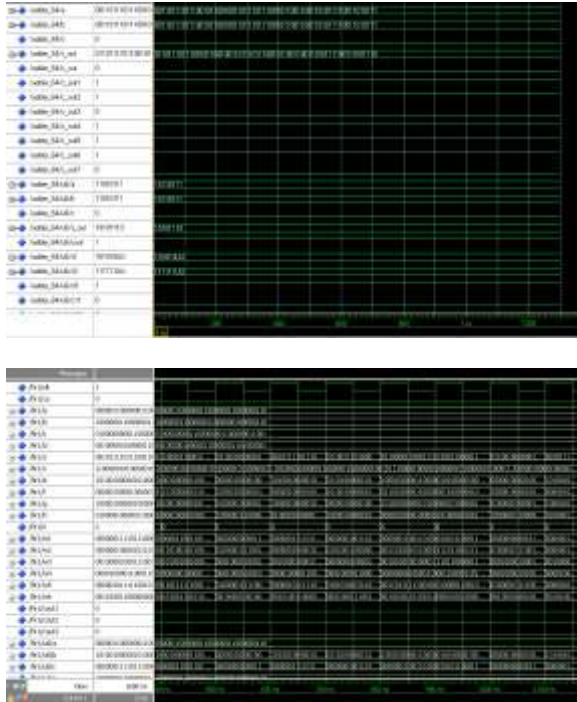

Fig.9 Block diagram of 64-bit BCDL adder

Fig.10 Simulated output of a Proposed 64 bit BCDL adder

The 64-bit adder section can be implemented in the application of FIR filter summer block operation. Finite impulse response (FIR) filter is a filter whose impulse response (or response to any finite length input) is of finite duration, because it settles to zero in finite time. This is in contrast to infinite impulse response (IIR) filters, which may have internal feedback and may continue to respond indefinitely (usually decaying).Here the main advantage of using this adder section in FIR filter is to mainly reduce the amount of computation power required to do the calculations. The measured propagation delay and switching energy of the BCDL adder depending on supply voltage are summarized in Table II.
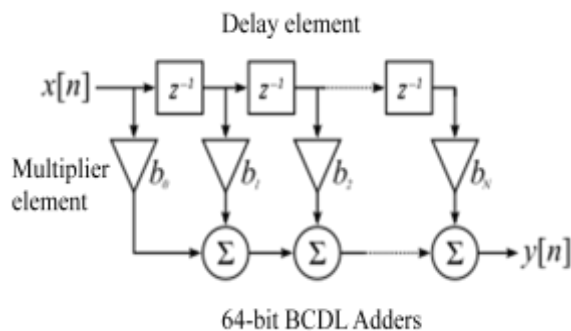

Fig.11 A discrete-time FIR filter of order N

Fig.11 shows the proposed FIR filter circuit for the implementation purpose of BCDL adder.

V.CONCLUSION

In this paper, a novel CMOS differential logic style with voltage boosting has been described. The BCDL provides higher switching speed than the conventional logic style at low supply voltage. The BCDL also minimizes area overhead by allowing a single boosting circuit to be shared by complementary outputs. Comparison results in a 90 nm CMOS process indicated that the power–delay product of the proposed logic style was improved when compared with conventional logic styles at supply voltage ranging from 0.4 to 1.2 V. The experimental result for a 64-bit BCDL adder designed with the proposed logic style in the filter applications so that the computational power required for the filter operations can be reduced. It can be implemented in the operation of FIR filters.

REFERENCES

1. M. Pedram and J. M. Rabaey, Power-Aware Design Methodologies. Boston, MA: Kluwer, 2002.

2. A. Wang and A. Chandrakasan, "A 180 mV FFT processor using sub-threshold circuit techniques," in Proc. IEEE ISSCC, 2004, pp. 292–295.

3. B. Zhai, L. Nazhandali, J. Olson, A. Reeves, M. Minuth, R. Helfand, P. Sanjay, D. Blaauw, and T. Austin, "A 2.60 pJ/Inst sub threshold sensor processor for optimal energy efficiency," in Proc. IEEE VLSI Symp., 2006, pp. 154–155.

4. D. Bol, D. Flandre, and J.-D. Legat, "Technology flavor selection and adaptive techniques for timing-constrained 45 nm sub threshold circuits," in Proc. ACM ISLPED, 2009, pp. 21–26.