Current Mode Multiple-Valued Adder Tree Design for High Performance Arithmetic Applications

Uğur Çini[1,2]
e-mail: ucini@dogus.edu.tr

Avni Morgül[2]
e-mail: morgul@boun.edu.tr

Mustafa Aktan[2]
e-mail: aktanmus@boun.edu.tr

[1]*Electronics&Comm. Eng. Dept. Dogus University Acibadem-34722 Istanbul, Turkey*
[2]*Electrical&Electronics Eng. Dept. Bogazici University 34342-Bebek Istanbul, Turkey*

Key words: Multiple-Valued Logic, Adder Trees, Computer Arithmetic

***Abstract* — Adder trees are the crucial design blocks of many arithmetic VLSI circuits. Alternative implementations can solve the large area requirements and complicated interconnection scheme of these building blocks. Multiple Valued Logic (MVL) implementation of an adder tree can be an alternative design style, which can save from area and interconnect.**

I. INTRODUCTION

Adder trees are the crucial design blocks for high speed arithmetic circuits, especially for multipliers. In an arithmetic circuit adder trees contain large number of logic gates and interconnections. Alternative implementations might solve the large area requirements and complicated interconnection schemes of these arithmetic building blocks. Multiple-valued arithmetic and logic circuit design is a challenging topic for high performance [1–4]. In this design, Multiple Valued Logic (MVL) implementation of an adder tree is designed. The design saves from area and interconnect whereas the performance of the system is still kept at the same level when compared to static CMOS implementation having equivalent functionality. Here, a multiple valued adder tree is implemented which is suitable for arithmetic applications such as general purpose multipliers and digital filters. (7,3) counter is one of the most basic elements of adder tree designs [5]. In a (7,3) counter, seven equal weighed binary inputs are added up together to have a three bit binary output. Fig. 1 shows the (7,3) counter circuit composed of four full adder blocks.

In this work, binary input and multiple valued output circuit of an adder tree is implemented. The output of the digits is in radix-4 form. MVL implementation of the system saves from area and interconnect, whereas the speed of the system is equivalent to the binary implementation. Current mode design is selected for high dynamic range since there would appear seven discrete levels at the input stage and four levels at the output.

Voltage mode implementations of multiple valued circuits for high radix is quite difficult, almost impossible to realize in today's low voltage and high speed requirements. The proposed custom designed multiple-valued adder tree is especially suitable for large bit width systems such as 32 or 64 bit adders or multipliers for its compactness. Since the system has quasi-analog behavior, scaling to different technology sizes can be made by small modifications in the geometry of the transistors. Selection of the unit current is also an important task where excessive currents would result high static power dissipation with very small speed improvement. In this design, 5 µA is selected as unit current. Current mode design is used for the design where all logic levels are represented as the increments of the selected unit current. A sample MVL circuit having equivalent function of (7,3) counter is designed and compared with its conventional implementation in binary logic. Since the design has a differential current mode style, there is no more dynamic power dissipation. Moreover, the design does not contain switching noise since it is implemented in a way that is similar to source couple logic technique. The circuit has been designed with Taiwan Semiconductor Manufacturing Company (TSMC) 0.25µm technology and operates with 2V single supply voltage. The adder tree circuit is simulated using Hspice and the worst-case delay is 1.5 nanoseconds. The designed MVL adder tree has 380mW static power consumption.

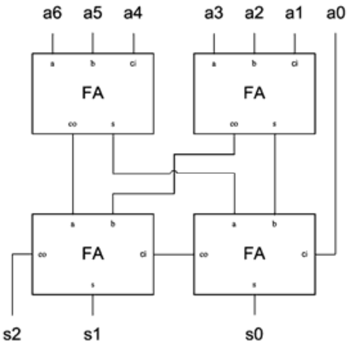


Figure 1. Binary (7,3) Counter

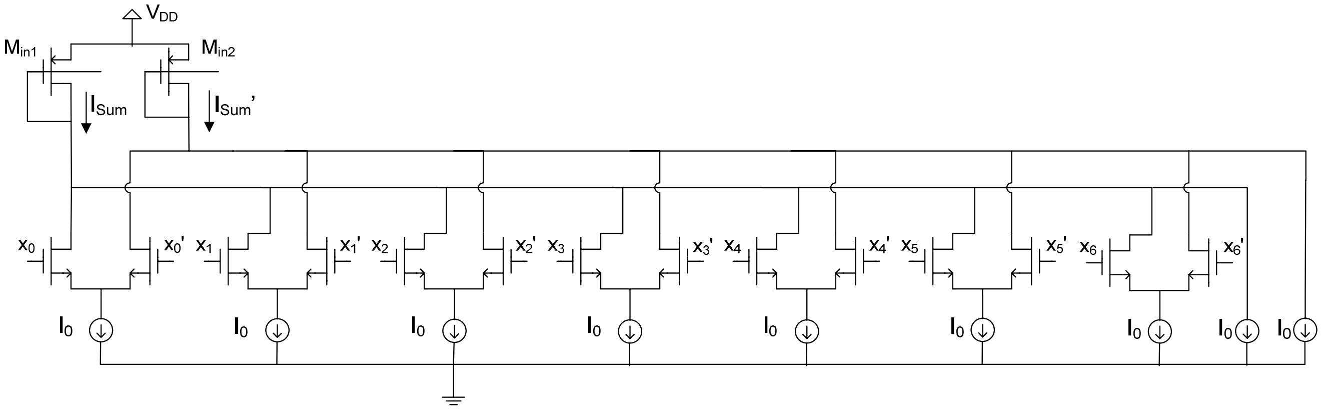


Figure 2. The input block of the adder tree. I_{sum} and I_{sum}' shows complementary current input values

II. MULTIPLE VALUED ADDER TREE

The adder tree is designed as a binary voltage input and multiple valued voltage or current output style. Binary input stage make the circuit compatible with other binary circuits in the system. At the end of the addition process, the output can be fed into another multiple valued system, or converted to binary with comparator circuits. Since the output has only four different discrete levels, building comparator circuits for this kind of system is not a difficult task. An efficient differential mode comparator circuit appears at [3], which is similar to a flash ADC converter.

Fig. 2 shows the input block of the adder tree. Here, x_0 … x_6 are the binary inputs of the circuit whereas x_0' … x_6' are the complementary inputs. As an example, whenever x_0 is high (x_0=V_{DD}), x_0' is low (x_0' ≈ V_{DD} - 0.5V) and I_0 current passes through the transistor M_{in1}. On the other hand, whenever x_0 is low, then x_0' is high and I_0 current this time passes through the transistor M_{in2}. In this way, the voltage mode binary inputs are converted to current values via current sources and the summation of the input currents are accumulated over diode connected transistors. The diode connected transistors carry I_{sum} and I_{sum}' that are complement of each other.

The circuit operates fully differential for better noise immunity and glitch-free operation. Differential design is actually essential because of the low noise margins of multiple levels in each wire, where differential design alleviates the design problems.

The unit current I_0 is selected to be 5 µA as stated before. A bias current of 5 µA is also supplied to both complementary input transistors which can be seen in Fig. 2. This bias current is used for better operation at the output stage, where 0 µA is quite difficult to acquire at the output. With this bias current source, the zero logic level is represented as 5 µA, and each logic level increases with 5 µA unit current levels. The corresponding current levels of each logic level of the input stage can be seen in Table 1. I_{sum} values increase by 5 µA steps whereas I_{sum}' decrease by 5 µA steps when the logical level increases. In radix-4 form, only the values between '0' and '3' are defined, since the output is higher than 3, a carry out signal must be generated, and the output must be kept between the defined logic levels.

III. CARRY GENERATION

For the addition operation in radix-4, a carry-out must be generated to make the number system operational when the input summation value exceeds '3'. Transistors M_{in1} and M_{in2} carry the I_{sum} and I_{sum}' current values respectively. Since M_{in1} and M_{in2} are diode connected, their input resistances are equivalent to:

$$r_i = \frac{1}{g_m + g_{ds}} \cong \frac{1}{g_m} = \frac{1}{\sqrt{2\mu_n C_{ox}\frac{W}{L}I_D}} \qquad (1)$$

and the voltage generated on the input transistors are:

$$V_{DS(M_{in1})} = \frac{I_{sum}}{\sqrt{2\mu_n C_{ox}\frac{W}{L}I_{sum}}} = \sqrt{\frac{I_{sum}}{2\mu_n C_{ox}\frac{W}{L}}} \qquad (2)$$

$$V_{DS(M_{in2})} = \sqrt{\frac{I_{sum}'}{2\mu_n C_{ox}\frac{W}{L}}} \qquad (3)$$

Table 1. Logic levels of I_{sum} and I_{sum}' currents at the input stage

Logic Level	0	1	2	3	4	5	6	7
I_{sum} (μA)	5	10	15	20	25	30	35	40
I_{sum}'(μA)	40	35	30	25	20	15	10	5

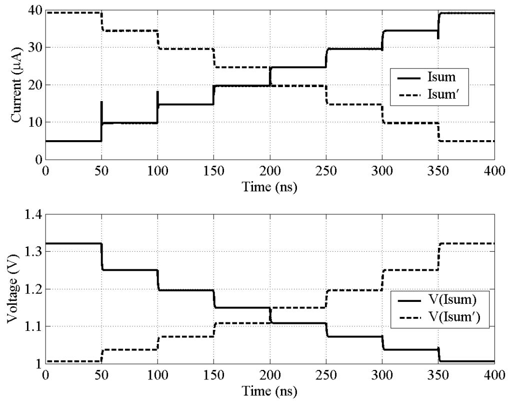


Figure 3. The input currents I_{sum} and I_{sum}' and generated voltages over the drains of the input transistors M_{in1} and M_{in2}

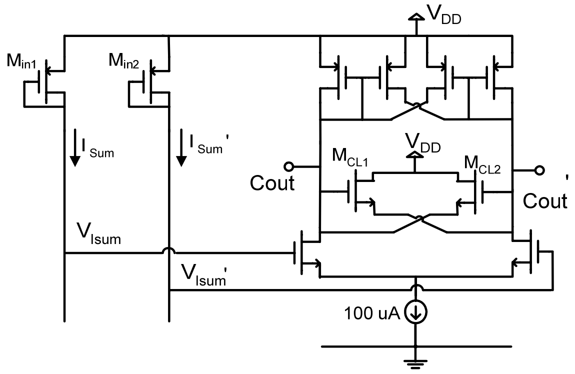
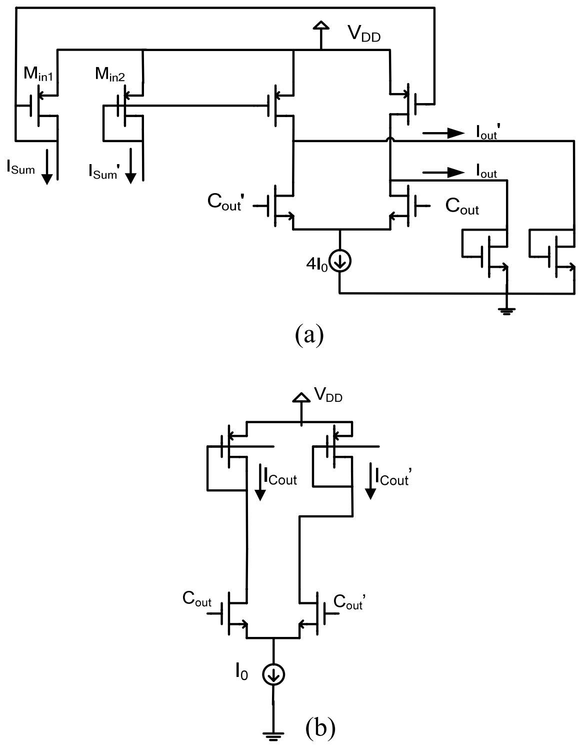


Figure 4. Voltage-mode carry generation circuit

Both of these diode connected transistors accumulate current inputs and copy these currents to other stages, at the same time the voltage drop generated over these transistors are fed into the differential amplifier for sensing the carry-out condition.



(a)

(b)

Figure 5. a- Output current, b- current mode carry-out generation circuit

As the current level over one of M_{in1} or M_{in2} increases, drain to source voltage over the transistor also increases. Fig. 3 shows the drain voltages of the transistors M_{in1} and M_{in2} as $V(I_{sum})$ and $V(I_{sum}$'). In the figure, the logic levels are swept from logical '0' to '7'; i.e. x_0 to x_6 are activated one after another. The critical point is the transition from '3' to '4' where the carry signal should be generated. As the figure is explored, it can be seen that the drain voltage level of the M_{in2} exceeds M_{in1} at the critical point. The comparison is made by complementary pairs that is in the decision boundary where I_{sum} increases from 20 to 25 μA and at the same time I_{sum}' decreases from 25 to 20μA.

Fig. 4 shows the carry-out circuit which is actually a differential amplifier with active load. Whenever the I_{sum} value is greater than I_{sum}', it means that the logic value of I_{sum} is greater than '3', i.e. there must be carry generated. In this condition, V_{Isum} voltage becomes smaller than V_{Isum}', which is fed into the differential amplifier, acting as a comparator. The comparator result represents the carry-out condition. M_{CL1} and M_{CL2} transistors are used to clamp the output voltages of the carry-out and carry-out' in certain levels [6]. The cross coupled load pair in the differential stage boosts the gain for carry generation [7,8].

When the carry is generated, logic '4' is subtracted from the Isum value and transferred to output as I_{out}. The current subtraction is also done by a differential pair, where I_{out} and I_{out}' are generated. As it can be seen in Fig. 5a, $4I_0$ is subtracted either from I_{out} or from I_{out}' depending on the carry out occurs or not for complement ary operation. The current values corresponding to logical values for I_{out} and I_{out}' are given in Table 2. A current mode carry-out signal may easily be generated from Cout signal of Fig.5a by using the simple circuit shown in Fig.5b.

Fig. 6 shows the $I(C_{out})$ and I_{out} plots. The complementary output, I_{out}', is not shown in the figure for clarity. Whenever the addition is completed at the adder tree stage, the output of the adder tree can be converted to

Table 2. Logic levels of I_{out} and I_{out}' currents at the output

Logic Level	0	1	2	3
I_{out}(μA)	5	10	15	20
I_{out}'(μA)	20	15	10	5

binary system by a four valued comparator block to connect the system to next binary blocks. After the binary conversion, the odd digits must be shifted left and added up with the even digits for compatibility between radix-2 and radix-4.

IV. CONCLUSION

A multiple valued adder tree that has equivalent function of (7,3) counter is designed. The system has less transistor count, less interconnect and consequently less area compared to its binary counterpart. After layout extraction, the circuit is simulated using HSpice and the delay of the system is found to be 1.5 nanoseconds. The binary implementation of the system is also designed and simulated, the delay is found approximately same as the MVL circuit. The designed circuit gives an alternative solution for high performance arithmetic systems. The power consumption of the circuit is 0.38 mW and does not change depending on the switching activity. The MVL adder tree contains 43 transistors, whereas binary adder tree has 112 transistors. In addition to area efficient implementation and fast operation, differential implementation gives the circuit analog friendly behavior where the switching activity is almost invisible on power supply lines. The system can be used together with source coupled logic (SCL) circuits seamlessly for the generation of input signals, which can provide fully differential design with lower area requirements. Fig. 7-a shows the layout of the MVL adder tree which is 15x33 μm^2. Fig. 7-b is the layout of the binary version of it having an area of 24x35 μm^2.

ACKNOWLEDGEMENT

This work is sponsored by Bogazici University Scientific Research Projects Fund: 05HA201.

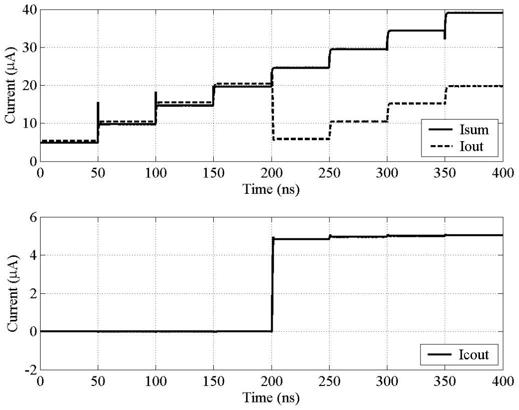
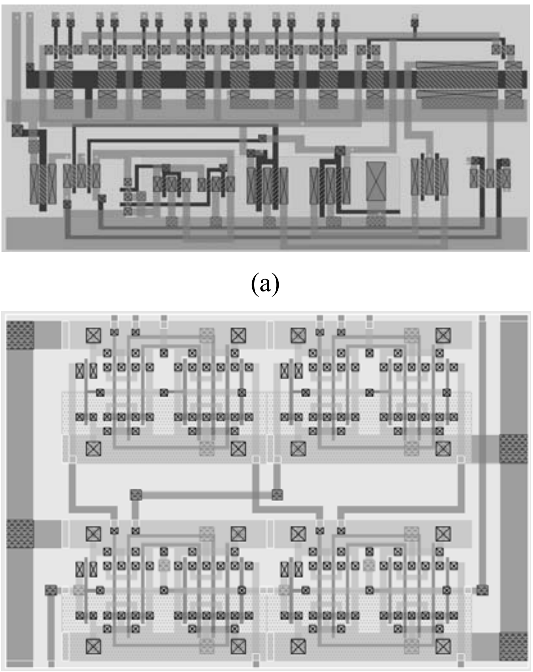


Figure 6. Output current and carry-out



(a)

(b)

Fig 7. Adder tree layouts; a-MVL circuit, b-binary circuit

REFERENCES

1. T. Hanyu, M. Kameyama, "A 200 MHz Pipelined Multiplier Using 1.5 V-Supply Multiple-Valued MOS Current-Mode Circuits with Dual-Rail Source-Coupled Logic", IEEE JSSC, Vol. 30, No. 11, pp. 1239-1245, Nov. 1995.
2. S. Kawahito, M. Ishida et. al., "High-Speed Area-Efficient Multiplier Design Using Multiple-Valued Current-Mode Circuits", IEEE Computers, Vol. 43, No. 1, pp. 34-42, Jan. 1994.
3. A. Mochizuki, T. Hanyu, "Highly Reliable Multiple-Valued Circuit Based on Dual-Rail Differential Logic", ISMVL 2006. Proceedings of the 36th IEEE International Symposium on Multiple-Valued Logic, May 2006.
4. T. Ike, T. Hanyu and M. Kameyama, "Fully Source-Coupled Logic Based Multiple-Valued VLSI", ISMVL 2002. Proceedings 32nd IEEE International Symposium on Multiple-Valued Logic, pp. 270 – 275, May 2002.
5. I. Koren, Computer Arithmetic Algorithms, AK Peters, Natick, MA, 2002.
6. B. Razavi, Design of CMOS Integrated Circuits, McGraw-Hill, 2000.
7. D. J. Allstot, "A Precision Variable-supply CMOS Comparatoı", IEEE JSSC, SC-17(6), 1080-1087, 1982.
8. R. Gregorian, Introduction to CMOS OP-AMPs and Comparators, 1999.