A CMOS Binary Adder Using a Quaternary Ganged-Logic Internal Node

Kenneth J. Schultz and Kenneth C. Smith

Dept. of Electrical Engineering
University of Toronto
Toronto, Ontario, M5S 1A4, Canada

ABSTRACT

This paper describes the design of a novel CMOS binary full-adder structure which incorporates four-valued signalling internally. A biased CMOS pseudo-linear adder provides a quaternary signal representing the number of ones in the three binary inputs. Three area-ratioed CMOS inverters interpret this to provide three binary signals which, combined in conventional static CMOS logic, generate the sum and carry outputs. The resulting "Ganged-Logic Adder" (GLA) is competitive in situations where higher static power dissipation and reduced internal noise margins can be tolerated in exchange for much lower input capacitance and faster carry propagation.

I. INTRODUCTION

Traditionally, a major emphasis in multiple-valued circuit and system design concerns the global use of multiple-valued signals (rather than simple binary) as a means to increase information-transfer efficiency, reducing the number of interconnecting signal lines, and so on. Rather less attention has been paid to the fact that the succinct encoding that MVL implies is potentially also of use on a more local scale. While, to a degree, this has been identified in arithmetic applications [1], demonstration has generally been at the systems level [2].

While also addressing an arithmetic application, the intent of the present paper is to illustrate the relevance and utility of MVL on a much smaller scale, namely within the confines of an otherwise conventional binary full adder. For it is within such a constrained environment that the usual reservations concerning MVL noise margin can best be countered. The demonstration circuit uses binary signalling for both input and output. However, in comparison to an accepted conventional CMOS implementation with comparable layout area, it demonstrates several advantages which are made available through the internal use of MVL. Most dramatically, it provides a six-fold reduction in input capacitive loading; as a result of a short carry propagation path, it is considerably faster for large adders, for example by about 27% for a 32-bit adder.

II. GANGED CMOS AND MVL

Full complementary CMOS static circuits dissipate negligible DC power, can operate asynchronously, and do not require the routing of clock signals [3]. However, static circuits are generally slower than dynamic circuits. One way of overcoming this deficiency is to trade off stand-by power consumption for speed. Johnson [4] recently presented a novel CMOS NOR gate using inverters with their outputs shorted together. Design of such NORs involves transistor ratioing to set appropriate high and low output levels.

The above concept can be generalized to "ganged" logic, in which inverter outputs are ganged together, and their transistor ratios are chosen to implement a certain function. Ratios can also be chosen to produce a number of different stable voltages on the "ganged" node, thus producing an MVL signal from binary inputs. By using this node as the input to a number of CMOS inverters with appropriate switching thresholds, the MVL signal can be encoded back to binary. The net result is the use of an MVL node to enable "ganged" CMOS to simultaneously realize multiple functions.

III. ADDER CIRCUIT DESIGN

A full adder is an example of a binary CMOS circuit in which the above concept proves to be useful. Full complementary CMOS adders are generally large circuits, with each of the 3 binary input variables driving a number of transistor gates. These adders typically have an undesirably large carry propagation delay, which can be decreased only by employing carry-look-ahead or carry-save adders of some sort; however, this increases area requirements even more. The MVL ganged CMOS approach enables the reduction of input capacitance and carry propagation delay, at the expense of noise margins and power consumption.

Figure 1 shows the MVL stage of the adder. All inputs and outputs are binary, while the signal on node Q is quaternary. All three inputs are not only logically equivalent, but they are also electrically equivalent; this is not the case in full complementary CMOS, where the position of a transistor in a pull-up or pull-down series tree affects delay. Ideally, if the inputs are all low, node Q will be high (V_{DD}); if one of the inputs is high, the node Q voltage is 2/3 V_{DD}; if two are high, it is 1/3 V_{DD}; if all inputs are at V_{DD}, node Q is at zero. The three output inverters have different switching thresholds, enabling them to act as encoders, producing the functions 1 of 3 (INV_d), 2 of 3 (INV_e), and 3 of 3 (INV_f). Different switching thresholds could be obtained by using multiple ion implantations [5]; a technique more easily implemented, and compatible with standard CMOS, is the ratioing of transistor widths.

Unfortunately, the "ideal levels" given above are impossible to obtain in practice, and thus the 3 output functions

0195-623X/89/0000/0356$01.00 © 1989 IEEE

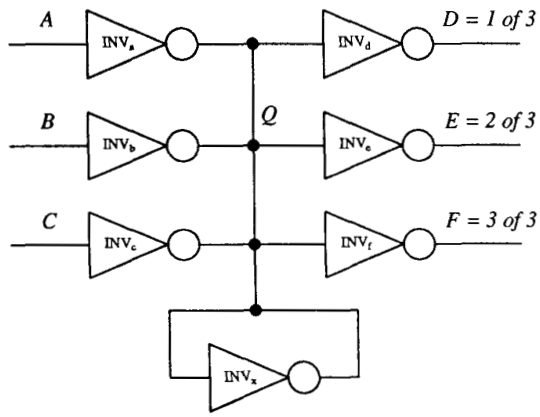

Figure 1: The basic Ganged-Logic topology operating in quaternary mode

cannot be realized with only the input inverters driving node Q. With one input at V_{DD} and two at GND, the voltage at Q will be so high that the switching threshold of INV_d will have to be unreasonably high ($W_{pd}>>>W_{nd}$) in order to differentiate this signal from V_{DD}. A similar situation occurs at the low end.

This problem is overcome by connecting INV_x as shown in Figure 1. This inverter provides negative feedback at Q, which tends to pull all stable voltages closer to a ratio-dependent central value, thus enabling differentiation between 0 and 1 of 3, as well as 2 and 3 of 3. The four possible voltages on Q can be adjusted by selecting values of W_{na}, W_{pa}, W_{nx}, and W_{px}, where $W_{na}=W_{nb}=W_{nc}$, and $W_{pa}=W_{pb}=W_{pc}$. The widths can be determined roughly by hand analysis; they must be "fine-tuned" using SPICE, or a similar circuit simulator. The same observation applies to transistor ratios of the output inverters $INV_{d,e,f}$.

A simple full complementary CMOS gate is used to realize the sum, according to the expression $S=F+D\bar{E}$, where $E=Cout$.

Before finalizing transistor sizes, the adder was laid out, following design rules of a 3 μm minimum feature size ($\lambda=1.5$ μm) process [6]. An extracted layout provides exact information pertaining to parasitics, and is thus valuable, in conjunction with SPICE simulations, in determining the effects of transistor sizes on transient performance. Noise margins can be traded off against speed by fine adjustments of the layout. Conceptually, the delay vs. transistor-width graph is concave upwards, and it is the designer's task to find the lowest point on the curve.

The following widths were chosen (all dimensions in μm, all lengths 3 μm):

$$W_{na}=7.5,\ W_{pa}=25.5,\ W_{nx}=22.5,\ W_{px}=33.0$$

These yield the following voltages at Q:

$$V_{0of3}=3.98,\ V_{1of3}=3.16,\ V_{2of3}=1.82,\ V_{3of3}=0.68$$

Table 1 presents ratios of the encoding inverters for suitable switching thresholds.

Figure 2 is a schematic showing the final 24-transistor circuit, including transistor widths. Figure 3 shows the layout.

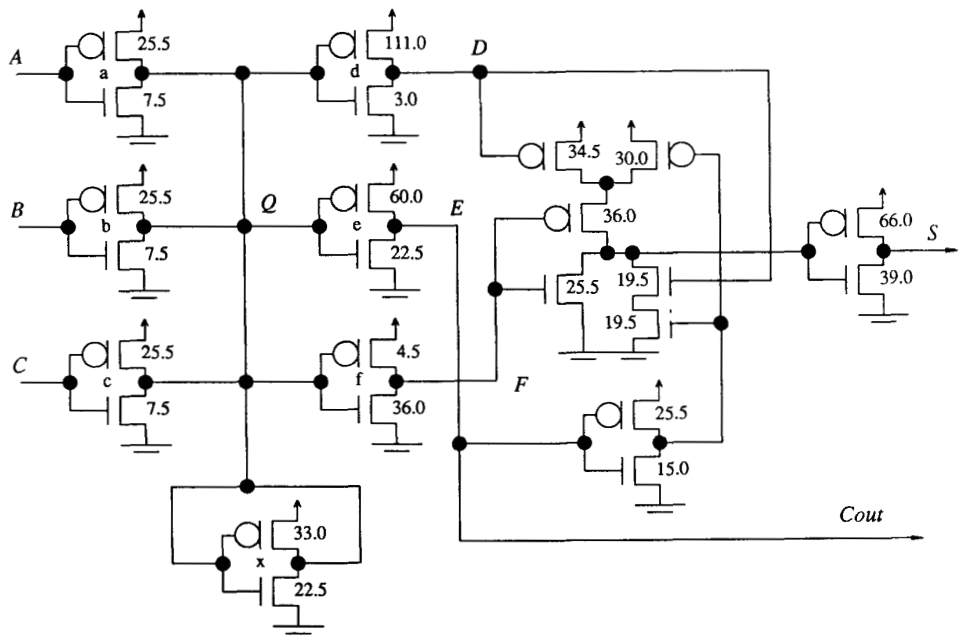

Figure 2: The CMOS full adder using a Ganged-Logic internal node

Table 1. GLA Encoding Inverter Information

inverter	V_{IL}	threshold	V_{IH}	W_n	W_p
INV_d	3.16	3.60	3.98	3.0	111.0
INV_e	1.82	2.56	3.16	22.5	60.0
INV_f	0.68	1.43	1.82	36.0	4.5

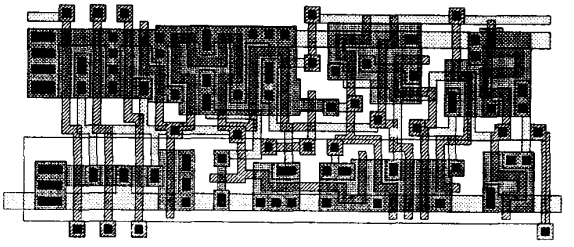

Figure 3: Layout of the GLA

IV. EVALUATION

In order to determine the merit of the circuit, its simulated performance was compared to that of a standard CMOS binary adder. The circuit chosen for comparison is the Hampel adder [7], a 28-transistor adder with a relatively short carry propagation delay, whose topology makes it easy to lay out (schematic shown in Figure 4).

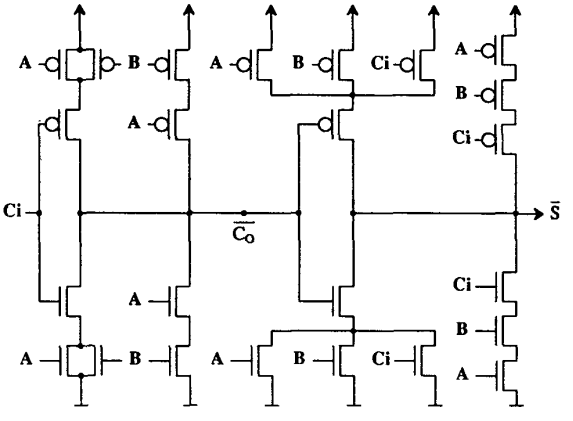

Figure 4: Hampel adder schematic; output inverters not shown

Both adders were laid out and extracted in an 8-bit wordlength, and simulations were performed using HSPICE. Results are summarized in Table 2. For a Ganged-Logic Adder, smaller size results in faster operation due to lower capacitance on node Q, as well as lower power dissipation. Because each input is applied to only one inverter, input capacitance is much lower than for the Hampel adder. Static power dissipation of the Hampel is practically zero, with an increase of 12 mW for 20 MHz operation. Static power dissipation of our adder is considerably higher, and varies with circuit inputs.

Worst case 8-bit addition time of the GLA is 10 ns faster than that of the Hampel adder. For an n-bit adder, addition time can be characterized as τ_S, the worst-case 1-bit sum delay, plus $(n-1)\times\tau_C$, where τ_C is the 1-bit carry propagation delay. Table 3 shows τ_S and τ_C for both adders, along with simulated 8-bit addition times, and extrapolated 16-bit and 32-bit addition times. It can be seen that the speed advantage of the GLA over the Hampel improves with longer wordlength, due to the shorter carry propagation delay.

Because noise margins are quite narrow compared to conventional implementations, further simulations were carried out using the Monte Carlo technique in order to ensure correct operation over a range of processing variations and supply voltages. Thirty passes were used; if the circuit functions correctly for all 30 trials, there is a 99% probability that it will also operate correctly over 80% of all component

Table 2. 8-bit Adder Comparison

property	(units)	Hampel	GLA
no. transistors		224	192
area	(mm^2)	0.144	0.141
A/B input cap	(fF)	900	105
C input cap	(fF)	620	105
8-bit time	(ns)	52	42
degraded time	(ns)	57	52
static power	(mW)	0	35-65
20 MHz power	(mW)	12	75

Table 3. Speed Comparison

(times in ns)	Hampel adder	GLA
τ_S	14	15
τ_C	5.4	3.8
8-bit	52	42
16-bit	95	72
32-bit	181	133

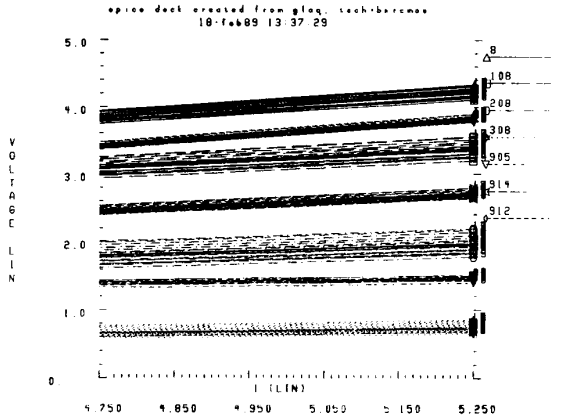

Figure 5: Monte Carlo simulation voltage distributions on the ganged node Q

values in the given distribution [8]. While V_{DD} varied from 4.75V to 5.25V (±5%), simultaneous gaussian distributions were used for transistor threshold voltages (σ=3%) and gate oxide thickness (σ=5%). Voltage distributions are shown for node Q in Figure 5. Noise margins of at least 220 mV are maintained at the ganged quaternary node. In the worst case, 8-bit addition time degrades to 52 ns. Note that addition time of the Hampel degrades to 57 ns over the same parameter range.

V. CONCLUSIONS

The Ganged-Logic Adder (GLA) presented in this paper is good in situations where high static power dissipation and narrow noise margins can be tolerated in exchange for lower input capacitance and faster carry propagation.

Its promise of success reinforces the view that MVL applications need not be on a grand scale to be useful. In particular, it suggests that it may be quite constructive to consider the creation of alternative mixed-radix standard cells for use in otherwise-binary environments.

REFERENCES

[1] K. C. Smith, "Multiple-Valued Logic: A Tutorial and Appreciation," *IEEE Computer*, vol. 21, no. 4, pp. 17-27, April 1988.

[2] M. Kameyama, S. Kawahito, and T. Higuchi, "A Multiplier Chip with Multiple-Valued Bidirectional Current-Mode Logic Circuits," *IEEE Computer*, vol. 21, no. 4, pp. 43-56, April 1988.

[3] N. Weste and K. Eshraghian, *Principles of CMOS VLSI Design*, Addison-Wesley, Reading, MA, 1985.

[4] M. G. Johnson, "A Symmetric CMOS NOR Gate for High-Speed Applications," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1233-1236, 1988.

[5] C. Zukerman, C. Afuso, M. Kameyama, and T. Higuchi, "Design of New Low-Power Quaternary CMOS Logic Circuits Based on Multiple Ion Implants," *Proc. 15th Int. Symp. on Multiple-Valued Logic*, pp. 84-90, May 1985.

[6] *Guide to Integrated Circuit Implementation Services of the Canadian Microelectronics Corporation*, Canadian Microelectronics Corporation, Kingston, Ont., Canada, January 1987.

[7] D. Hampel, K. E. McGuire, and K. J. Prost, "CMOS/SOS Serial-Parallel Multiplier," *IEEE J. Solid-State Circuits*, vol. SC-10, pp. 307-313, 1975.

[8] *HSPICE Simulation Report 10*, Meta-Software, Campbell, CA, March 1988.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the financial assistance of the Natural Sciences and Engineering Research Council of Canada through an NSERC 1967 Science and Engineering Scholarship and an NSERC Operating Grant A1753. As well, the cooperation of Professor C.A.T. Salama is truly appreciated.