Circuit Driven Delay Optimization of EMODL Carry Lookahead Adders

J. Wang, Z, Wang, G.A. Jullien, S.S. Bizzan, W. Luo and W.C. Miller

VLSI Research Group
University of Windsor, Windsor, Ontario, Canada N9B 3P4
jullien@engn.uwindsor.ca

Abstract

Delay minimization of carry look-ahead adders using the recently reported Enhanced Multiple Output Domino Logic (EMODL), is investigated. Delay versus tree height, using an analytical transistor sizing technique, is analyzed, and the trade-off between the tree height and the number of stages is discussed. Four architectures for a 32-bit adder are compared at the layout level and experiments show that the number of stages is more critical for delay optimization. Mask level simulations predict an aggressive 2.1ns critical path for the best architecture using a 1.2 micron CMOS technology. The simulation procedure is verified by fabrication.

1. Introduction

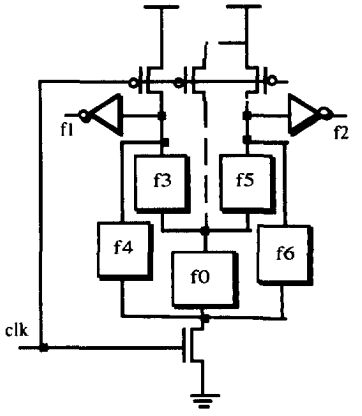
The design of fast adders still continues to hold a fascination for researchers in the area of VLSI implementation of computer arithmetic circuits. Optimal performance (i.e. minimization of the adder critical path) is, in the final analysis, a direct function of the specific transistor level details of the circuitry. To this end, we use a recently introduced domino logic circuit style (EMODL) to drive the design of fast adders for CMOS technology, and for this study we use the carry lookahead (CLA) adder architecture. There is recent evidence that the CLA Adder represents a competitive architecture at the VLSI implementation level [6], despite the proliferation of other adder architectures.

In this paper, a CLA architecture is designed and implemented in 1.2μ CMOS technology. The major features of this design are the utilization of a pseudo complement decomposition technique, that is matched to the EMODL circuit style, and a sparse carry chain. The worst case delay path of a target 32-bit adder is reduced to three gate stages compared to five as reported in a recent publication [3]. The EMODL gates are optimized using a modification to an approximate analytical technique developed for single path chains [5]. The major factors affecting the critical path of the adder are: fan-in and fan-out of each gate; the dimension of each MOS transistor; the number of cascaded gates in the critical path. We investigate the trade-off of these factors on area and critical path delay and analyze four different 32-bit adder architectures. Our SPICE simulation techniques are verified by comparing simulations of a complete adder vs. fabrication results for the critical path. Our results show considerable improvement over published designs using similar technology and circuit styles.

2. Enhanced multiple output domino logic

Domino logic has become the de facto standard for dynamic logic design since its introduction in 1982 [7]. Modifications to Domino Logic include the recent concept of Multiple-Output Domino Logic (MODL) which relies on the matching of boolean decompositions to stacked circuits, each with its own inverter output [3]. We have expanded this concept to the more general case where an intermediate logic function, which is not necessary an output, is the common base of two or more logic functions (see Figure 1); we call this circuit style *Enhanced MODL* (EMODL).

Figure 1 EMODL Circuit Concept



In this paper we will discuss a special CLA decomposition which takes advantage of this circuit style.

3. CLA Adder Circuit Design

We briefly review the CLA algorithm, and then discuss the concept of *pseudo-complements*, which is the backbone of our EMODL circuit architecture.

1058-6393/95 $4.00 © 1995 IEEE

CLA algorithm

Given an n-bit adder with two binary summands a_i and b_i, $i = 1, 2, ...,n$, with carry-in $c_0 = 0$, three auxiliary logic variables, carry generate g_i, carry propagate p_i and half sum, or the exclusive OR (XOR), x_i, are defined as:

$$g_i = a_i \cdot b_i \tag{1}$$

$$p_i = a_i + b_i \tag{2}$$

$$x_i = a_i \oplus b_i \tag{3}$$

An operator, o [2], is defined as follows:

$$(g_i, p_i) o (g_j, p_j) = (g_i + p_i \cdot g_j, p_i \cdot p_j) \tag{4}$$

$$(g_i, p_i) o c_{i-1} = g_i + p_i \cdot c_{i-1} \tag{5}$$

eqn. (5) is limited to the carry-in case. Note that the operator is associative but not commutative.

For a group, starting from the bit position i, and ending at the bit position k ($i<k$), we define the group generate, $g_{i,k}$, and group propagate, $p_{i,k}$, as given by:

$$(g_{i,k}, p_{i,k}) = (g_k, p_k) o (g_{k-1}, p_{k-1}) o ... o (g_i, p_i) \tag{6}$$

The ith carry can be computed as:

$$c_i = [p_{1,i}, g_{1,i}] o c_0 \tag{7}$$

and the sum by $s_j = c_j \oplus x_j$. In our CLA design we produce $g_{j,k}$ and $p_{j,k}$ directly from the first stage without generating g_i and p_i. We require, however, to take some care to prevent false evaluation due to *sneak paths* (paths that cause currents to flow in opposite directions through MOSFETs in the trees, and form unintended closed paths to ground).

Pseudo complements

Since we are considering a full domino logic design for the adder, both true and complements of x_i and c_i have to be created in order to allow evaluation of the sum s_i. We requires extra transistors to form the complement circuits of $g_{j,k}$ and $p_{j,k}$, since direct inversion of Domino Logic is not allowed in the middle of a Domino Chain. In order to generate the complement carry chain in parallel with the true carry chain, while retaining the same circuit speed and silicon area, we define a pseudo-complement generate $\hat{g}_i = \bar{a}_i \bullet \bar{b}_i$, and a pseudo-complement propagate $\hat{p}_i = \bar{a}_i + \bar{b}_i$. We now re-use the operator, o:

$$(\hat{g}_i, \hat{p}_i) o (\hat{g}_j, \hat{p}_j) = (\hat{g}_i + \hat{p}_i \cdot \hat{g}_j, \hat{p}_i \cdot \hat{p}_j) \tag{8}$$

Then the pseudo-complement group generate $g_{i,k}$, and the pseudo-complement group propagate $p_{i,k}$ can be generated in parallel:

$$(\hat{g}_{i,k}, \hat{p}_{i,k}) = (\hat{g}_k, \hat{p}_k) o (\hat{g}_{i-1}, \hat{p}_{k-1}) o ... o (\hat{g}_i, \hat{p}_i) \tag{9}$$

The complement carry chain can now be obtained:

$$\bar{c}_i = (\hat{g}_i, \hat{p}_i) o \bar{c}_{i-1} \tag{10}$$

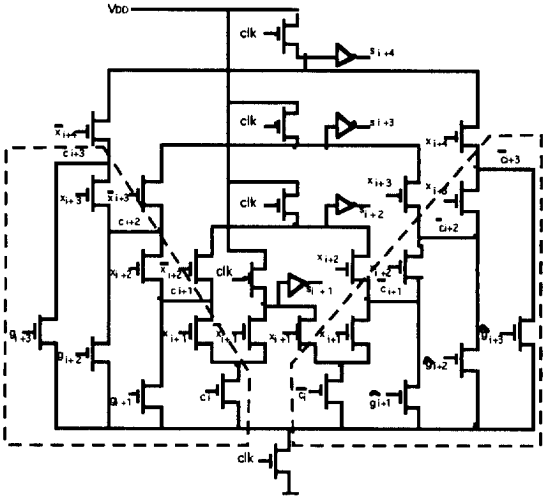
$$= (\hat{g}_i, \hat{p}_i) o (\hat{g}_{i-1}, \hat{p}_{i-1}) o ... o (\hat{p}_1, \hat{g}_1) o \bar{c}_0$$

$$= (\hat{p}_{j,1}, \hat{g}_{j,1}) o \bar{c}_0 \tag{11}$$

EMODL Circuits

Figure 2 shows an EMODL circuit, which generates four consecutive sums (outputs) from a single carry-in. The two blocks (shown in broken lines) are the carry chain (LHS) and the complement carry chain (RHS). The EMODL circuit enables the generation of several consecutive sums with a single carry-in., with two changes to the CLA adder design:

Figure 2 EMODL circuit for four consecutive sums



1. There is no need to generate the bit level generate and propagate of the conventional CLA adder, and the preliminary stage can be eliminated.
2. The full carry chain, which contains carries for all bit positions has been replaced by a sparse carry chain.

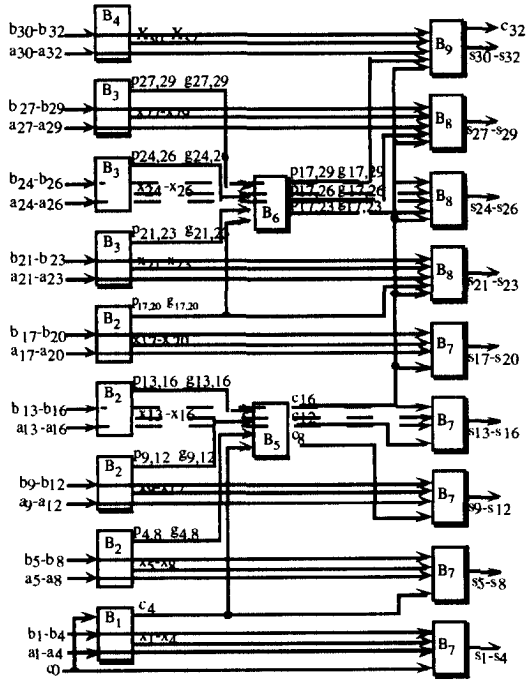
32-bit adder design

Figure 3 shows the block diagram for a 32-bit adder [1] which we later show is optimal in terms of delay and area.

The adder is constructed using 9 different block types, $\{B_k\}$. In the first stage, the XOR for each bit x_i, along with

group carry generates/propagates, and carry, c_4, are generated. The group size is 4 for bit positions below 20, and 3 for bit positions above 20. In the second stage, only c_8, c_{12} and c_{16} are produced in the sparse carry chain, to feed to the third stage. Group carry generates/propagates with group width of 3 and 4 are also produced for the third stage, which consists of three types of EMODL gates for generating the final sum for each bit position.

Figure 3: Block diagram for a 32-bit adder



4. Transistor sizing technique

As the logic block complexity increases, logic height (fan-in) increases, and transistor sizing becomes essential to reduce pull-down delay. Transistor sizing, in some cases, is the critical factor in choosing one circuit configuration over the other. In [5] analytical formulae are introduced to size single NFET chains with a single evaluation node. This formulation has to be modified to size the more complex EMODL.

The basic single chain formulation is contained in eqn. (12) to eqn. (14), where *PURS* and *PURL* are per unit resistance for saturated and non-saturated channels, W_i is the width of the ith transistor in the chain, the $\{C_i\}$ are node capacitors and the $\{K_i\}$ are constants based on SPICE measurements.

The definitions and the methods to evaluate the parameters in eqn. (13) and eqn. (14) are given in [5].

$$W_N = \frac{K_{1,N} PURS}{\frac{T_D}{N+1} - K_2 PURS} \tag{12}$$

$$W_i = \frac{PURL(K_{3,i} + K_5 W_{i+1} + C_{i+1} + \ldots + C_N)}{\frac{T_D}{N+1} - K_4 PURL} \tag{13}$$

$$C_i = K_{3,i} + K_4 W_i + K_5 W_{i+1} \tag{14}$$

Note that $K_{1,N}$ and $K_{3,i}$ replace K_1 and K_3, respectively. Each new capacitance constant contains the original value in addition to the source/drain capacitance of those transistors connected to the node under consideration. The pull-down delay of the evaluation node is a function of the structure and the sizes of the NFET circuitry and also a function of the number and sizes of the driven FETs of the subsequent blocks. For feed-forward architectures, such as the CLA adder, the loading affect can be removed if transistor sizing originates from the output stage blocks.

The following is a brief description of the new algorithm used to size the EMODL logic blocks:

1 Choose a desired pull-down delay.

2 Select a suitable precharge PFET size and evaluation node inverter sizes based on the driven load.

3 Set all NFETs to a minimum size.

4 Search for the worst discharge path using the RC model and Elmore's delay formula.

5 Size the NFET chain of the worst path using eqn. (12) through eqn. (14).

6 Repeat step 4 and 5 until all discharge paths satisfy the delay requirement or no improvement can be achieved.

7 Check if the PFET size is suitable. If not, resize the PFET and go to step 4.

It is important to mention that during the search for the worst discharge path, the top evaluation node should be considered first and then the lower nodes. If all discharge paths of the lower evaluation node are connected to the upper one, then the lower evaluation node may be ignored during the search process. This is possible due to the fact that, for optimally sized NFET chains, the potential profile of the nodes increases monotonically with node height during the discharge process [8].

5. Tree Height vs. Number of Stages

In the previous sections, we have shown that the EMODL circuit concept enables us to build CLA adders with fewer numbers of stages compared to the conventional approach. If no restriction is imposed on the fan-out, a much lower number of stages can be achieved. Table 1 gives maxi-

mum word lengths possible using a two stage structure.

Table 1 Word length for 2-stage implementations

Fan-in	2	3	4	5	6	7	8
max length	2	5	9	14	20	27	35
fan-out	1	2	3	4	5	6	7

Table 1 shows that it is possible to implement a 32-bit adder by two stages of logic gates, but the maximum tree height, including the ground switch, is 9, which may produce considerable charge sharing problems. For a three stage implementation, the maximum tree height is 6, with a fan-in pattern of 5, 4, 5, and a four stage implementation has maximum tree height of 5, with a fan-in pattern of 4, 3, 3, 4.

To investigate the relationship of the gate delay with fan-in, we simulated optimally sized EMODL circuits, with results shown in Table 2 for the worst case pull-down delay for the sum with the largest index.

Table 2 Relationship between delay and fan-in

Fan-in	3	4	5	6
Delay (ns)	0.325	0.392	0.514	0.629

An important observation from Table 2 is that the delay relationship for sized circuits is not quadratic, and so optimization procedures based on this assumption [4] will fail to produce optimal results.

There are many factors affecting the delay of the complete adder. The most important are: number of stages; delay of each stage (fan-in); and the fan-out. Although we cannot precisely predict which combination of the number of stages and the tree height provide the shortest delay based on individual block delays, e.g. table 2, we still can make comparative judgements by approximating complete adder delays by a summation of block delays. Based on such a comparison, we conclude that the delay for the three stage 32-bit adder is slightly lower than the four stage implementation.

In order to confirm our judgement, we experimented with the critical path of four 32-bit adder architectures. For III and IV we required to add a preliminary stage for the bit level generate and XOR required for later circuits. The fan-in patterns for these two additional configurations are 2, 4, 4, 5, and 2, 3, 3, 3, 4, respectively.

Table 3 shows the HSPICE simulation results from mask extracted circuits. The transistor counts are also reported.

Table 3 Architectural Comparison

Scheme	I	II	III	IV
Type of p_i	a_i+b_i	a_i+b_i	$a_i \oplus b_i$	$a_i \oplus b_i$

Table 3 Architectural Comparison

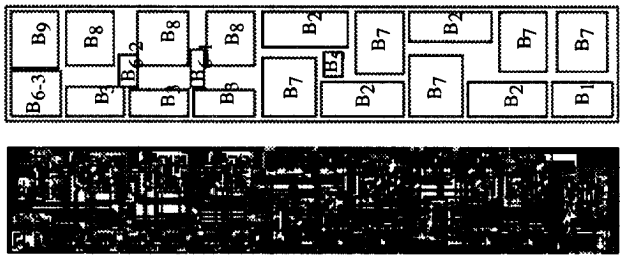
Scheme	I	II	III	IV
Fan-In Pattern	5, 4, 5	4, 3, 3, 4	2, 4, 4, 5	2, 3, 3, 3, 4
Outputs in Critical Path	c_4, c_{16}, c_{32}, s_{20}	c_3, c_9, c_{27}, s_{30}	c_1, c_4, c_{16}, c_{32}, s_{20}	c_1, c_3, c_9, c_{27}, s_{30}
Max. Fan-Out	5	7	7	5
No. of stages	3	4	4	5
HSPICE Delay	2.10ns	2.36ns	2.37ns	2.55ns
Device Count	1397	1537	1691	1826

From Table 3 we conclude that our original three stage implementation of the 32-bit adder provides minimum total delay and device count. Our experiments also show that device count and area are closely related.

6. layout regularity

As an experiment to observe the effect of irregular block architectures on silicon area, the *Scheme I* 32-bit adder was laid out. The best layout procedure is to separate the first and the last stage into 2 separate rows, with the second stage smaller blocks used to fill in space. The block distribution and layout are shown in Figure 4.

Figure 4: *Scheme I* 32-bit Adder Layout



The LSB of the adder is at the right. The carry-in is zero. The total width of the adder area is the sum of the widths of the first and last stage cells, with a total area of $337 \times 2169\mu^2$.

For comparison, the delay of the 32-bit adder given by Hwang and Fisher [3], which was fabricated in a 0.9μ CMOS technology, is 3.1ns. Even without applying scaling criteria our design is considerably faster than theirs, and our design uses fewer devices. Table 4 compares the main char-

acteristics of our design with that of [3].

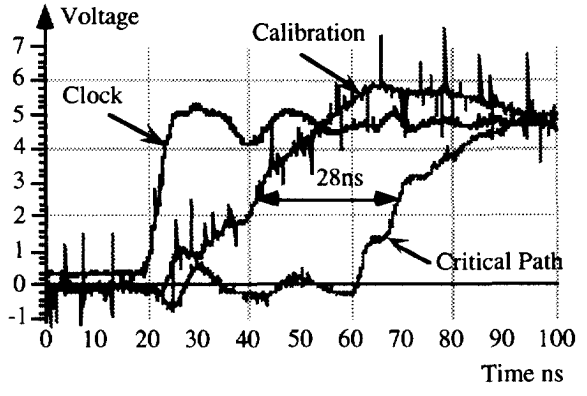
Table 4 Comparison between the two designs

Design	Process	Delay	Stages	Devices
[3]	0.9μ CMOS	3.1ns	5	1325
New	1.2μ CMOS	2.1ns	3	1269

7. Verification of simulations

In order to verify the accuracy of our simulations we fabricated a test chip early into the project. The transistor sizing was sub-optimal in that we did not have available the EMODL sizing algorithm at the time of the fabrication. Simulation results of the critical path produced a 2.7ns delay (measured at the 50% waveform point). In order to measure the delay from a fabricated circuit, we built 10 cascade stages of the critical path and provided a calibration input/output pad. The test results are shown in Table 5.

Table 5 Fabrication Tests



From the results we can make an estimate of the time delay per stage by subtracting the calibration output delay from the critical path output delay and dividing by 10 (the number of critical paths in series). At a 50% point on the waveforms we measure the critical path delay at about 2.8ns. This is in remarkable agreement with the simulations. There is clearly some uncertainty in the delay measurement but our simulation procedure appears to be sound. It is interesting that this sub-optimally sized adder critical path can result in a considerable increase in the delay; this points to the need to very carefully size transistors for time-critical performance.

8. Conclusions

In this paper we have discussed the design of CLA Adders driven from EMODL circuits. Our decomposition technique results in the interesting concepts of pseudo complements and sparse CLA carry chains.

A recently introduced analytical transistor sizing technique has been modified for multiple EMODL transistor logic tree paths. We show, from simulations, that for optimally sized trees, the quadratic delay rule with tree height does not apply. We discuss the various factors associated with obtaining minimum time designs, and explore the trade-off between block fan-in and the number of stages. By simulating four 32-bit adder critical path schemes, we have been able to empirically compare the speed and area characteristics; these characteristics are functions of gate size, fan-out load, maximum number of cascaded gates, etc., with intractable analytical solutions. The comparison study, however, reveals that the number of stages is the most crucial factor in designing a low critical path adder.

Based on this analysis, a fast dynamic adder with only three gate stages in the critical path is designed in a 1.2μ CMOS technology. Simulation results show an addition time of 2.1ns; a very aggressive result compared to recently published designs in similar technologies. The validity of our simulation results are verified from test chip measurements.

9. Acknowledgments

The authors acknowledge financial support from the Natural Sciences and Engineering Research Council of Canada and the Micronet Network of Centres of Excellence and fabrication and tools support from the Canadian Microelectronics Corporation.

10. References

[1] Wang, Z., Jullien, G.A., Miller, W.C., and Wang, J., "New Concepts for the Design of Carry Lookahead Adders", *Proc. of 1993 ISCAS*, pp. 1837-1840, 1992.

[2] Ladner, R.E. and Fischer, M.J., "Parallel Prefix Computation", *J. ACM*, vol. 27, No.4, pp831-838, 1980 New York: Wiley & Sons, 1979.

[3] Hwang, I.S. and Fisher, A.L., "Ultrafast Compact 32-bit CMOS Adders in Multiple-Output Domino Logic", *IEEE J. Solid-State Circ.*, 24, No.2, pp.358-369, 1990.

[4] Chan P.K. and Schlag, M.D.F., "Analysis and Design of CMOS Manchester Adders with Variable Carry-Skip", *IEEE Trans. Comput.*, vol. 39, No.8, pp.983-992, 1990.

[5] Bizzan, S.S., Jullien, G.A., Miller, W.C., "Analytical Approach to Sizing nFET Chains", *IEE Electronics Letters*, vol. 28, No. 14, pp. 1334-1335, 1992.

[6] T. K. Callaway and E. E. Swartzlander. "Optimizing arithmetic elements for signal processing", in VLSI Signal Processing, V (Edited by K. Yao et al) IEEE Publication, New York NY, pp. 91-100, 1992

[7] Krambeck, R.H., Lee, C.M. and Law, H-F. S. "High Seed Compact Circuits with CMOS", *IEEE J. Solid-State Circ.*, June, 1982.

[8] Shoji, M. "FET Scaling in domino CMOS Gates", *IEEE. J. Solid-State Circuits*, vol. 20, pp. 1067-1071, 1985.