A New Design of a Fast Barrel Switch Network

G. M. Tharakan and S. M. Kang

Abstract—A new implementation of the barrel switch with a speedup of 20-30% as compared to that of the prior barrel switch design is proposed in this paper. Domino CMOS is used to implement the design, which can shift or rotate 32-b data to the left or right by any number of bit positions between 0 and 31. The principles behind the design, timing and area comparisons, layout, simulation, and test results will be presented.

I. INTRODUCTION

Shift left of 32-*n* bits in a 32-b machine. This leads to an increase in the critical path length and hence decreases the speed. Specifically, the prior design in [2] requires six stages (one for right shift and five for left shift) to implement the barrel switch. In comparison, the proposed new design requires only four stages to do the same operations, resulting in a speedup of 20-30%. This paper will also discuss the principles behind the design, the logic design, and an area estimate.

II. DESIGN PRINCIPLE

To reduce the number of stages in the data path, a novel concept is introduced. In general, computer operations are based on binary logic, but some operations lend themselves naturally to ternary or more valued logic as in multiplication [3], division, shifting, and rotation. Consider shifting, for instance, we have three alternatives, namely, 1) shift right, 2) shift left, and 3) do neither. Such operations can be more easily implemented with ternary logic (-1, 1, 0). Hence, doing the above operations with binary values would lead to extra stages. On the other hand, since the prevalent hardware technology can only support binary values, implementing ternary logic would required additional logic circuitry.

Applying the above ternary concept to the 32-b barrel switch, we can reduce the number of stages to four. The first stage shifts the data 1 b right/left, the second stage shifts the data 3 b right/left, the third stage 9 b right/left, while the fourth stage shifts data 18 b right/left. Hence,

Manuscript received June 22, 1991; revised September 26, 1991. This work was supported in part by the Semiconductor Research Corporation under Contract SRC 90-DP-109 and by the Joint Services Electronics Program under Contract N00014-90-J-1270.

IEEE Log Number 9105084.

to shift data by 11 b to the right, the control vector would be (S3, S2, S1, S0) = (0, -1, -1, 1) where -1, 1, and 0 imply right shift, left shift, and no shift, respectively. The first stage shifts the data 1 b to the left, the second stage shifts them 3 b to the right, the third stage shifts them 9 b to the right, and the last stage does no shift, resulting in the data being shifted right by 9 + 3 - 1 =11 b. It is shown in the Appendix that all possible shifts can be achieved in this way. The inputs to the barrel switch will be binary, of which the control vector has to be converted to the ternary form. This conversion logic is discussed later on.

III. TIMING ANALYSIS

Consider the critical path of the design in [2] shown in Fig. 1. It consists of six stages with each stage having a fan-out of 2. Assuming that the drain capacitance of each stage is $2C_d$, the wiring capacitance of the interconnect metal is C_w , and the channel resistance in each stage is R, then by applying the Elmore's time constant method, the total worst-case delay through the critical path would be

$$\tau_{\text{old}} = R(2C_d + C_w) + 2R(2C_d + C_w) + 3R(2C_d + 2C_w) + 4R(2C_d + 4C_w) + 5R(2C_d + 8C_w) + 6R(2C_d + 16C_w) = 42RC_d + 161RC_w. (1)$$

Now consider the critical path of the new implementation as shown in Fig. 2. It consists of four stages with each stage having a fan-out of 3. The drain capacitance of each stage is now increased to $3C_d$ while the wiring capacitance C_w and the channel resistance are the same, which will result in a total critical delay of

$$\tau_{\text{new}} = R(3C_d + C_w) + 2R(3C_d + 3C_w) + 3R(3C_d + 9C_w) + 4R(3Cd + 18C_w) = 30RC_d + 106RC_w.$$
(2)

Hence, an effective speedup of $(42 - 30) \times 100/42$ = 29% is to be expected if the RC_d term is dominant; on the other hand, if the RC_w term is dominant, then (161 -106) × 100/161 = 33% is expected. A similar analysis can be done for the design in [4] over which a speedup of 12% can be achieved.

IV. CONVERSION LOGIC FOR CONTROL SIGNALS

The conversion logic for control signals can be implemented with a PLA or using standard cells. It is expected

0018-9200/92\$03.00 © 1992 IEEE

The authors are with the Department of Electrical and Computer Engineering and Coordinated Science Laboratory, University of Illinois, Urbana, IL 61801.



Fig. 2. Critical path of the new implementation.

that in a microprocessor the logic evaluation would either precede or occur simultaneously along with the data shifting and, hence, will not cause a slow down of the critical path.

Typically, the control inputs to the barrel switch may consist of 7 b: 1 b for shift/rotate, 1 b to indicate left/ right shift, and 5 b to indicate number of shifts. The outputs of PLA would consist of 12 b (3 b for each stage) indicating left, right, or no shift. The PLA implementation requires only five inputs to specify the number of shifts (this reduces the number of rows in the PLA) and the outputs are gated to the appropriate signals to derive the left/right shift.

The outputs will be sent through a multiplexer which selects L_k/R_k depending on the left/right-shift input and passes it to the barrel switch. For left-shift operations, L_k is sent onto the pass gate implementing left shift while $\overline{L_k}$ is sent to the pass gate for right shift. The resulting outputs, which will be fed directly to the shifter, are shown in Table I for various shift combinations.

V. AREA ESTIMATE

For a 32-b machine, there will be 32 horizontal bit paths and, in addition, eight more paths are required to handle overshoot in the first two stages. Each path has four stages

TABLE I CONTROL VECTORS FOR SHIFT OPERATIONS

Operation	CONTROLS											
	SHIFT 1			SHIFT 3			SHIFT 9			SHIFT 18		
	L0	S 0	R0	LI	S 1	RI	L2	S 2	R2	L3	\$3	R3
Left Shift												
0	0	1	0	0	1	0	0	1	0	0	1	0
4	1	0	0	1	0	0	0	1	0	0	1	0
7	1	0	0	0	0	1	1	0	0	0	1	0
14	0	0	1	0	0	1	0	1	0	1	0	0
16	1	0	0	0	0	1	0	1	0	1	0	0
23	0	0	1	0	0	1	1	0	0	1	0	0
31	1	0	0	1	0	0	1	0	0	1	0	0
Right Shift												
2	1	0	0	0	0	1	0	1	0	0	1	0
5	1	0	0	1	0	0	0	0	1	0	1	0
11	1	0	0	0	0	1	0	0	1	0	1	0
13	0	0	1	0	0	1	0	0	1	0	1	0
17	1	0	0	0	1	0	0	1	0	0	0	1
22	0	0	1	0	0	l	0	1	0	0	0	1
29	1	0	0	0	0	1	0	0	1	0	0	1

and each stage has three transistors, but the eight additional paths require only two transistors in each stage. From the above data, a rough area estimate can be made: the number of transistors is equal to $32 \times (3 \times 4) + 8$

1



Fig. 3. Detailed floorplan for the new barrel switch.

 \times (2 \times 4) = 448, while the interconnect area is proportional to the number of connecting lines (number of shifts) which is equal to (1 + 3 + 9 + 18) \times 2 = 62.

In the previous design there are 32 horizontal bit paths; each path has six stages and each stage has two transistors. There are no extra paths since no overshoot occurs. Hence, the number of transistors is equal to $32 \times (6 \times 2) = 384$, and the interconnect area is again proportional to the number of shifts, i.e., $(1 + 2 + 4 + 8 + 16) \times 2 = 62$.

Therefore, the new implementation has a slight increase in area due to the overshoot handling, and a worst-case estimate of this increase is $(448 - 384)/384 \times 100 = 17\%$.

VI. FLOORPLAN AND LAYOUT OF BSW

A more detailed floorplan of the barrel switch (BSW) is shown in Fig. 3. The input register contains 32 latches while the control signals are fed in from the top, out of the PLA and the multiplexer. There are 12 CONTROL

signals as well as four ROTATE signals which are laid out vertically. The shifter requires a two-phase clock for precharging and evaluation. The shifter along with the PLA was laid out in the MAGIC System [5] with 3.0- μ m design rules. The entire layout was also plotted and is shown in Fig. 4. Two-layered metal technology, using domino CMOS, was used to reduce layout and interconnection area.

VII. SUMMARY

The new implementation of the BSW was designed using $3-\mu m$ double-metal domino-CMOS technology. Logic verification of the BSW has been done by using the MUSA switch level simulator [6]. The area required for laying out the entire circuit was 4.2 mm \times 3.3 mm, and the circuit was simulated using SPICE and MOSIS model parameters. The critical path delay of the circuit was 27.53 ns. This compares very favorably with the 62 ns that was reported in [2]. A novel feature of the design is the re-



Fig. 4. Layout of the new barrel switch.

duction in the number of stages for shifting from six to four, resulting in a 30% increase in speed.

APPENDIX

Since there are 3 degrees of freedom, i.e., shift left, shift right, and no shift at each stage, shifts are done in powers of 3 at each stage. Therefore, shifts of 1, 3, 9, $27, \cdots$, are implemented in the design. To prove that any number of shifts can be achieved by this configuration, we resort to mathematical induction.

Assume that it is possible to achieve *n* shifts; therefore $a(m)3^m + a(m-1)3^{(m-1)} + \cdots + a(0)3^0 = n$ (A1) where $a(k) \in \{1, 0, -1\}$.

The result is easily demonstrated for the first few integers (e.g., n = 0, 1, 2, 3). Now assume that (A1) is true for some arbitrary n.

To achieve n + 1 shifts, find j such that $a(j) \neq 1$ and a(i) = 1 for all i < j. Assign a(j) = 1 + a(j) and all other a(i) = -1 where i < j. This has the effect of adding 3^{j} and subtracting the 3^{i} terms twice (for each i < j). This

is equivalent to adding

$$3^{j} - 2(3^{j-1} + 3^{j-2} + \cdots + 3^{2} + 3 + 1)$$

= 3^{j} - (3^{j} - 1) = 1 (A2)

to n. Here we have used the well-known identity

$$1 + r + r^{2} + \cdots + r^{n-1} = (r^{n} - 1)/(r - 1).$$
 (A3)

Therefore, if the equation is true for n, we can find a solution set $\{a(m), m = 0, 1, 2, \dots\}$ for n + 1. It is true for n = 1 by assigning a(0) = 1. Therefore, it is true for all natural n. Also with 1, 3, 9, and 27, we can achieve a total of 40 shifts, but for a 32-b BSW, we only need 1, 3, 9, and 18 shifts in stages. In general, for a B-bit machine, the number of stages J will be

$$J = \lfloor \log_3 \left(2B + 1\right) \rfloor \tag{A4}$$

where |x| stands for rounded-up integer value of x.

ACKNOWLEDGMENT

The authors are grateful to the reviewers for their helpful comments and suggestions, especially in shortening the proofs in the Appendix.

References

- [1] R. S. Lim, "A barrel switch design," Computer Design, pp. 76-78,
- Aug. 1972.
 [2] S. M. Kang, "Domino-CMOS barrel switch for 32-bit VLSI processors," *IEEE Circuits Devices*, pp. 3–8, May 1987.
 [2] Mech.
- [3] A. D. Booth, "A signed binary multiplication technique," J. Mech.
- Appl. Math, vol. 4, pt. 2, pp. 236–240, 1951.
 [4] E. Hokenek et al., "Second-generation RISC floating point with multiply-add fused," *IEEE J. Solid-State Circuits*, vol. 25, pp. 1207– 1213, Oct. 1990.
- [5] MAGIC Layout System Manual, Univ. Calif., Berkeley, Dec. 1985.
- [6] MUSA SIMULATOR, Univ. Calif., Berkeley, July 1989.