

Self-timed Circuit Device Size Optimization for an Input Data Distribution

Alvernon Walker

Dept. of Electrical and Computer Engineering
North Carolina A&T State University
Greensboro, NC 27411

Evelyn R. Sowell*

Dept. of Computer Systems Technology
North Carolina A&T State University
Greensboro, NC 27411
sowells@ncat.edu
336-285-3145

Abstract— New design techniques with energy-delay characteristics that are superior to that of the synchronous timing and control approach are needed today because the throughput of systems realized with this method is limited by the power dissipation of nanometer scale devices and the power management strategies developed to insure that they do not exceed device thermal constraints. A circuit timing approach that is not dependent only on the propagation delay of the critical path is required to achieve this for a specified technology and supply voltage. Optimized self-timed circuits have this characteristic and therefore outperform synchronous designs for a given energy dissipation. A novel self-timed circuit device sizing approach that is based on the circuit input data distribution is proposed in this paper. The analysis is based on the Logical Effort RC model [1] of a ripple-carry adder. The model was extracted from SPICE simulation for the TMS320C64 0.18 μ m process. The performance and energy dissipation of circuits implemented with this approach is 13% and 16% respectively better than circuits designed with previously proposed approaches.

Keywords- self-timed circuits, energy dissipation, ripple carry adder, energy-delay product, and asynchronous circuits.

I. INTRODUCTION

The central focus of digital system design engineers over the past two decades has been on the trade-offs between the power/energy and performance of the circuits implemented in current and emerging nanometer-scale VLSI technologies. A number of techniques have been developed to address this design challenge; one approach is based on a class of asynchronous pipelined digital circuit structures that are called self-timed [2]. The dynamic power/energy dissipation is reduced in this realization, relative to synchronous implementations, because all clocks are generated locally and circuit timing and control is event driven. The performance of these circuits can exceed synchronous realization because it is based on the average intrinsic timing of the circuit instead of its worst case timing that is used to set the clock frequency in synchronous systems. The circuit design process used to determine the device sizing in self-timed circuits/systems is typically the same as that used for synchronous realizations [3,4,5]. The input distribution is not considered in this process. A novel self-timed circuit design technique that out performs previously proposed approaches is presented in this paper. The input data distribution is used in the proposed technique to optimize the circuit performance for the respective input data set probability distribution.

The performance and energy dissipation of synchronous and asynchronous digital system is determined in part by the geometry of the devices used to realize the system embedded gates. The device geometry is set in the design process to minimize the propagation delay along all the paths in the systems. This approach maximizes the performance of synchronous systems because the propagation delay of the circuit critical path is also minimized. However the performance of asynchronous circuits is not maximized because the average propagation delay is not minimized. The performance and energy dissipation of asynchronous circuits that are optimized for the average delay of the completion detection circuit are maximized and minimized respectively. The proposed technique achieves this because it is based on the average completion circuit propagation delay and the circuit input data distribution.

A novel self-timed circuit device sizing approach is presented in this paper. The analysis and methodologies used to develop the approach is covered in section 2. The performance and energy dissipation of the proposed approach is compared to circuits that were designed with device sizing method that are used for synchronous circuits in section 3. The conclusion is presented in section 4.

II. MATERIALS AND METHODS

A. Motivation

Power conservation without performance penalties have become an increasingly important issue among modern digital circuit designers. As the digital technology evolution continues to produce more complex circuits coupled with ground breaking system performance, the power consumed by these circuits are at record highs. In fact, power dissipation or energy loss in the form of heat is reaching levels comparable to nuclear reactors. The negative affect associated with the power dissipation compromises or in many cases, impair chip reliability and life expectancy.

B. Energy Delay Product

When we consider the energy or power with respect to performance from the prospective of a gate, there are several challenges. As Moore's Law continues to hold, the number of transistors on a chip will double every 18 months, the increasing clock frequencies and chip density have allowed designer to create more desirable architectures which run applications at ground breaking speeds. However, the micro-architecture and logic designs are stressed as frequency has increased faster than scaling. Since clock frequency is a linear function of

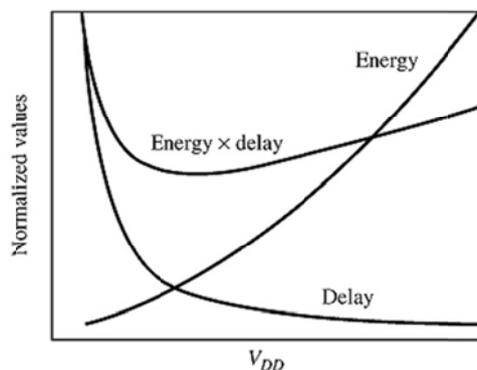


Figure: 1.1 Energy-Delay Product

power dissipation, as we increase the frequency we also increase the power dissipation. Further reducing the number of gate delays per cycle will also be difficult to achieve because the interconnect parasitics associated with the wires of a circuit are starting to dominate the speed or performance of the circuit not the gate. There are several problems that have to be resolved to build faster and more efficient chips: better chip implementation design techniques, better clock system design strategies and a more efficient micro-architecture.

Figure 1.1 illustrates as we increase the supply voltage, the delay of the gate decreases. However, the power dissipation increases, as well. This is called the energy delay product. One of the measures of efficiency for a digital system is the energy delay product, propagation delay multiplied by energy dissipation which is measured in joules. There have been several papers that investigate techniques that explore the possibilities of optimizing the energy-delay product more in depth [6, 7, 8, 9].

Modern digital designers, most often use synchronous logic to build computer systems because this logic style is more commonly accepted due largely in part to the commercial infrastructure which has already become acclimated. Traditional synchronous system designers often believe that in order to boost performance one must pay a power penalty or vice versa, which is consider power/performance tradeoff. Figure 1.3 shows the ideal energy delay product. Our challenge here is to figure out how to build a gate that is fast and power efficient. Can we increase performance without increasing power dissipation? Is it possible to have a superior power delay product?

Current trends suggest that we can. Let's take a look at some Multi-processor units (MPS) and Digital Signal Processors (DSP) which are typically the highest performing chips. In figure 1.2a, as Moore's Law remains true, more transistors are packed into a small chip. The initial effects on power dissipation are increasing most rapidly in the 1980's. This is due in part to technology that was not as power efficient as today's technology. In the early 80's the transistor sizes were much larger and the circuit operated at a lower clock frequency. The Intel's 8088 operated at 4.77 mhz as opposed to today's personal computers that can operate at 2 Ghz which is about 400 times faster. Then in the early 1990's as more power efficient architectures were introduced, (e.g. RISC, pipelining, super scalar, and branch prediction) power dissipation still increased but at a much slower rate. This is demonstrated by the difference in the data lines in the figure 1.4a. The first data line shows a four times increase in power dissipation every three years. While the second data line shows a 1.4 times increase in power dissipation every three years.

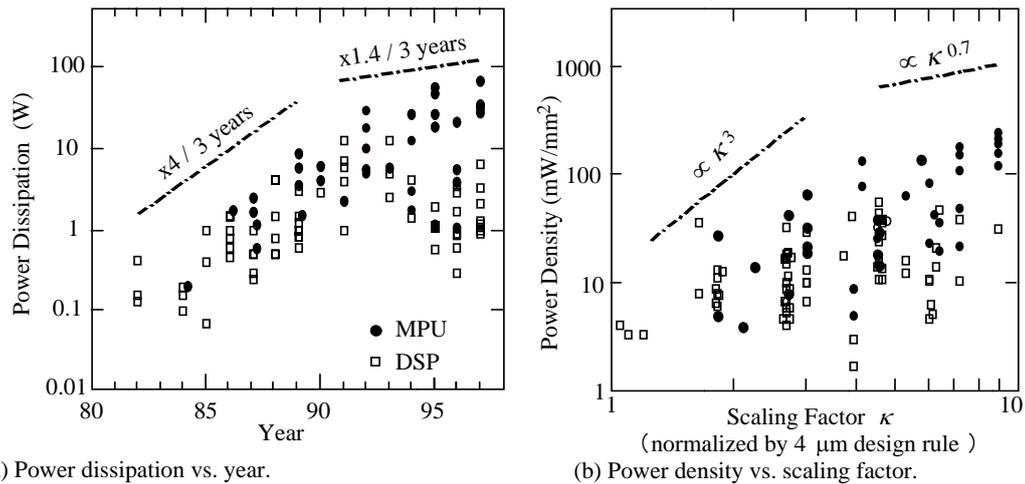


Figure 1.2: Power Dissipation with respect to year and scaling factor [3]

The same hold true with respect to scaling, figure 1.2b. As device sizes decrease, the intrinsic time constant is reduced which implies that clock frequencies and power dissipation increase. However, the more efficient architectures had the same effect on the slope to the data line. It did not increase as rapidly. This demonstrates that by building architectures that are more efficient, we get an energy penalty that is less. Furthermore, it is possible to build such systems and that different design strategies can deliver a superior energy delay product. In short, performance is constrained by power. Design choices affect the power efficiency of a circuit and can offer something more in terms of performance. By developing a circuit with a better energy delay product, we can achieve better performance per joule, which gives us new possibilities. One example would be for portable devices, the battery life can be increased and applications can run as long as possible. My goal was to build a system that yields more performance per joule.

C. Logic Gate Delay

Now that we understand how self-timed circuits are realized, let’s review how we model the timing process. The delay in a logic gate is determined by the topology of the gate (fan in) and the capacitive load that the logic gate drives (fan out). Logical effort is a term coined by Ivan Sutherland and Bob Sproull in 1991 which is a method that is used to model the delay of a single logic gate. Logical effort method provides a technique to determine the most efficient transistor sizing on the critical path to minimize the delay, as well as, providing an estimation of that delay. The delay of a logic gate using logical effort is given as:

$$d = f + p \quad (1)$$

where p is the parasitic delay which is the intrinsic delay of the gate driving no load, and f is the stage effort. The stage effort is defined as:

$$f = gh \quad (2)$$

$$d = gh + p \quad (3)$$

where g is logical effort which is the ratio of the input capacitance of a given gate to that of an inverter capable of delivering the same output current and h is effective fan out C_{out}/C_{in} . The dependency is demonstrated in figure 1.3.

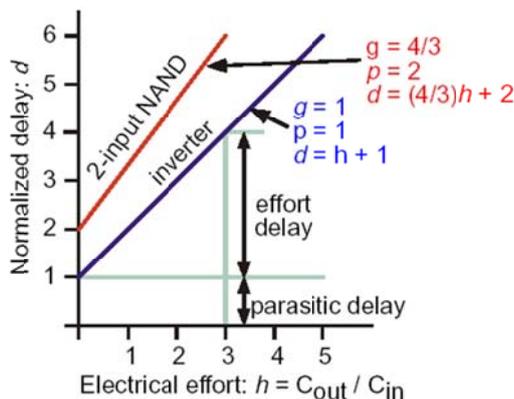


Figure 1.3: Delay expressed in terms of a minimal sized inverter [1]

The delay is a function of electrical effort of and inverter for a two input NAND gate. The slope of each line is the logical effort and the y-intercept is the parasitic delay. As shown, we can adjust the total delay by adjusting the electrical effort or by choosing a logic gate with a different logical effort [1].

D. Circuit Device Sizing with Input Distribution Data

To achieve high performance and manage power loss, designers should consider non-traditional levels of abstraction, in particularly, input data profiling. Since the switching activity of a logic gate is a strong function of the input signal statistics, system designers can use this knowledge to exploit power delay capabilities of a circuit. In this dissertation, a pipelined architecture that intersects the timing function of the circuit itself and the data that it is processing is utilized. Using input data distribution to increase self-timed circuit performance and decrease energy dissipation is novel because the timing is determined locally, which is a function of the circuit and the input data.

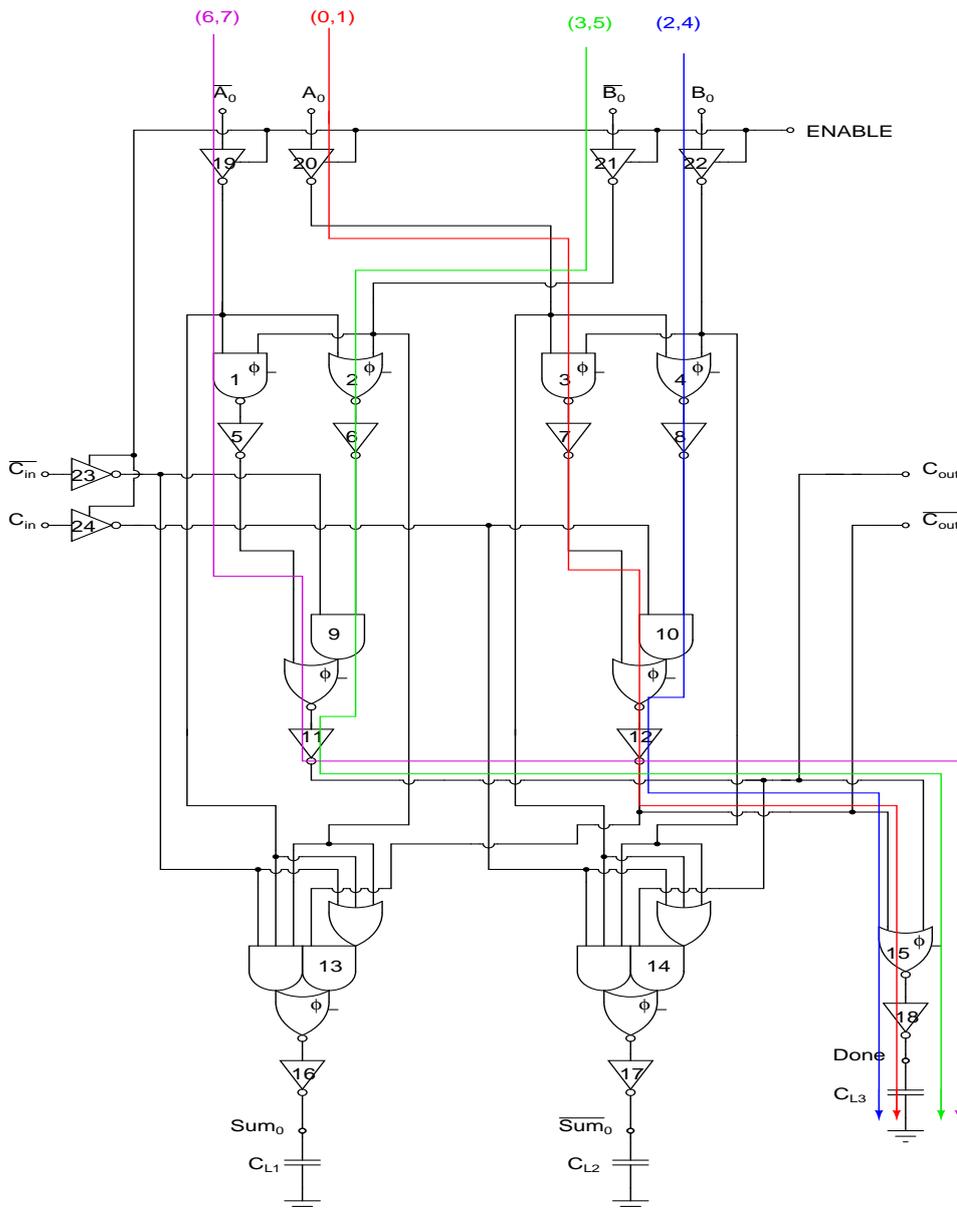


Figure 1.4: Circuit Path Activation Probability

A few advantages of this proposed technique is the decreased circuit area. This is realized when the probability of a path being used is very low then the transistors on the path will be sized smaller. There is also an increase average circuit performance because when you include data profiling, performance is even better than self-timed alone. The average energy dissipation is decreased since energy is only consumed when and event happens. The decrease circuit noise is due in part by the fact that fewer transistors are used which decreases circuit activity. The local clock distribution alleviates the greedy global clock network and hazards that can be introduced by clock skew. This technique is less sensitive to changes to process variation because timing is generated locally. Figure 1.4 gives a graphical illustration of a one bit self-timed RCA circuit path activation probability with eight different input distributions (0-7) and four different activation or critical paths illustrated by the different colors along the path.

There are a few disadvantages. There are very few Computer Aided Design development tools for design a verification. Sensitive to charge sharing is another concern that is just the nature of dynamic logic which can be offset by circuit design that is sized to minimize the effect.

The performance and energy dissipation of synchronous and asynchronous digital system is determined in part by the geometry of the devices used to realize the system embedded gates. The device geometry is set in the design process to minimize the propagation delay along all the paths in the systems. This approach maximizes the performance of synchronous systems because the propagation delay of the circuit critical path is also minimized. However the performance of asynchronous circuits is not maximized because the average propagation delay is not minimized. The performance and energy dissipation of asynchronous circuits that are optimized for the average delay of the completion detection circuit are maximized and minimized respectively. The proposed technique achieves this because it is based on the average completion circuit propagation delay and the circuit input data distribution.

A self-timed full adder is used in this section to demonstrate the proposed device sizing approach. The adder is implemented with domino logic and dynamic input latches. It is shown in fig.1.4. The time between the start signal (i.e. self-timed circuit local clock) rising transition and the rising transition on the *Done* node in fig. 1.4 is defined as the completion time of the adder. It is a function of the execution time of the self-timed circuit/system functional block. It depends on the circuit inputs and therefore it is the average of all the active critical path delays for the circuit input space. The active critical path delay is the propagation delay along the longest signal path for a given circuit input over the 2^n valid input combinations of a self-timed circuit with n primary input bits. The circuit in fig. 4.24 contains four active critical paths. The circuit four active critical paths from the primary inputs (i.e. A_o , B_o and C_{in}) to the output of the completion detection circuit (i.e. node *Done*) are shown in fig. 1.4 with the respective inputs that activate the paths. The bits that define the numbers in fig. 1.4 are organized as follows: $A_0B_0C_{in}$ where A_0 is the MSB. All equation are normalized with respect to the average intrinsic time constant, i.e. $\tau = 17.527$ pSec for TSMC process, of a CMOS process.

Recall the formula that was used to calculate the delay, $d=gh+p$. Shown below in equations are the estimated delay associated with the four active paths for input distributions, where,

$C_{AOI21B10}$ is input capacitance of AOI21 gate on input B, labeled 10 in fig. 1.4,

g_{AOI21C} - logical effort of AOI21 gate from input C,

P_{NOR} - NOR gate parasitic effort

LD_{EN} -input latches and

W_0 - probability circuit input is 000,

W_1 - probability circuit input is 001,

W_2 - probability circuit input is 010,

W_3 - probability circuit input is 011,

W_4 - probability circuit input is 100,

W_5 - probability circuit input is 101,

W_6 - probability circuit input is 110,

W_7 - probability circuit input is 111.

The expected completion time of the full adder is the average of the active critical path delays $D_0, D_1, D_2, D_3, D_4, D_5, D_6$ and D_7 . It equals equation (4). The unknown parameters in Fig. 1.4 related to the device geometry is $C_{NAND1}, C_{NOR2}, C_{NAND3}, C_{NOR4}, C_{invh5}, C_{invh6}, C_{invh7}, C_{invh8}, C_{AOI21B9}, C_{AOI21B10}, C_{invh11}, C_{invh12}, C_{SUMA13}, C_{SUMA14}, C_{NOR15}, C_{invh16}, C_{invh17}$ and C_{invh18} . The average is:

$$D_{avg} = W_0D_0 + W_1D_1 + (W_2 + W_4)D_{2,4} + (W_3 + W_5)D_{3,5} + W_6D_6 + W_7D_7 \tag{4}$$

completion time of the adder is minimized if these values are set such that,

$$\begin{aligned}
 0 &= \frac{\partial D_{avg}}{\partial C_{NAND1}} \\
 0 &= \frac{\partial D_{avg}}{\partial C_{NOR2}} \\
 0 &= \frac{\partial D_{avg}}{\partial C_{NAND3}} \\
 &\vdots \\
 0 &= \frac{\partial D_{avg}}{\partial C_{invh18}}
 \end{aligned}$$

The Newton-Raphson method is used to find the circuit parameters (i.e. unknown capacitances above) when the expressions in the equation above vanish.

The equation for branching effort is $B = \frac{OnPath+OffPath}{OnPath}$

Let,

$$\begin{aligned}
 B_0 &= \frac{C_{NAND3} + C_{NOR4} + C_{SUMA14}}{C_{NAND3}} = \frac{C_{NAND3} + C_{NOR4} + C_{SUMA14}}{(C_{NAND3} + C_{NOR4})(1-S)} \\
 B_0' &= \frac{C_{NAND3} + C_{NOR4} + C_{SUMA14}}{C_{NOR4}} = \frac{C_{NAND3} + C_{NOR4} + C_{SUMA14}}{(C_{NAND3} + C_{NOR4})S} \\
 B_1 &= \frac{C_{NOR15} + C_{SUMD13}}{C_{NOR15}}
 \end{aligned}$$

The stage effort is:

$$f = \sqrt[7]{g_{LD(en)} \cdot g_{NAND} \cdot g_{invh}^3 \cdot g_{AOI21C} \cdot g_{NOR} \cdot B_0 \cdot B_1 \cdot \left(\frac{C_{L3}}{C_{LD(EN)}} \right)}$$

The propagation delay along the path is:

$$D_{110(111)} = 7f + P_{LD(EN)} + P_{NAND} + P_{NOR} + P_{AOI21C} + 3P_{invh} \quad (\text{path though NAND gate 1})$$

The load at the input of high-skew inverter 12 is:

$$C_{invh12} = \frac{C_{L3} \cdot g_{invh}^2 \cdot g_{NOR} \cdot B_1}{f^3}$$

The propagation delay through the NOR gate is:

$$f' = \sqrt[4]{g_{LD(EN)} \cdot g_{NOR} \cdot g_{invh} \cdot g_{AOI21B} \cdot B_0' \cdot \left(\frac{C_{invh12}}{C_{LD(EN)}} \right)}$$

$$D_{010(100)} = 3f + 4f' + P_{LD(EN)} + 2P_{NOR} + P_{AOI21B} + 3P_{invh} \quad (\text{path though NOR gate 2})$$

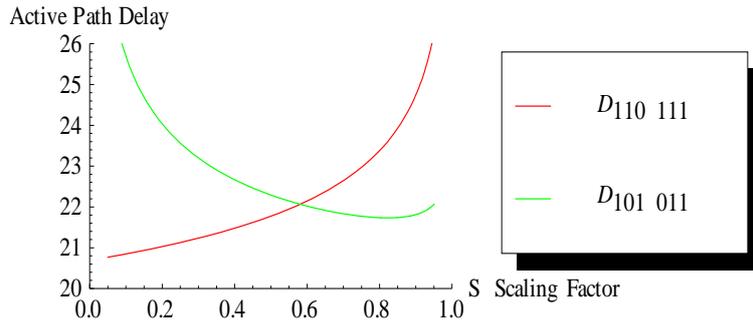


Figure 1.5: Trading delay in one path for delay in another

The inverting logic in the full adder shown in fig. 1.5 is a mirror image of the un-inverted logic. If the input probability distribution is not symmetrically distributed then the delay associated with each side of the adder should be different. This is achieved in the proposed approach by adjust the input capacitance of the sum gates. The branching effort in the circuit associated with path is:

$$B_0 = \frac{C_{NAND} + C_{NOR} + C_{SUMA}}{C_{NAND}}$$

$$B_1 = \frac{C_{NOR} + 2sC_{SUMD}}{C_{NOR}}$$

$$B_1' = \frac{C_{NOR} + 2(1-s)C_{SUMD}}{C_{NOR}}$$

where s is a user defined scaling factor.

The stage effort of the left and right path in fig. 1.5 is:

$$f_{Left} = \sqrt[7]{g_{LDu} \cdot g_{NAND} \cdot g_{invh}^3 \cdot g_{AOI21C} \cdot g_{NOR} \cdot B_0 \cdot B_1 \left(\frac{C_{L3}}{C_{LD(EN)}} \right)}$$

$$f_{Right} = \sqrt[7]{g_{LDu} \cdot g_{NAND} \cdot g_{invh}^3 \cdot g_{AOI21C} \cdot g_{NOR} \cdot B_0 \cdot B_1' \left(\frac{C_{L3}}{C_{LD(EN)}} \right)}$$

The path delay of the left and right sides is:

$$D_{Left(001)} = 7f_{Left} + P_{LD(EN)} + P_{NAND} + P_{AOI21C} + P_{NOR} + 3P_{invh}$$

$$D_{Right(110)} = 7f_{Right} + P_{LD(EN)} + P_{NAND} + P_{AOI21C} + P_{NOR} + 3P_{invh}$$

The delay associate with each of these paths as x is swept from 0.1 to 0.9 is shown in fig. 1.6.

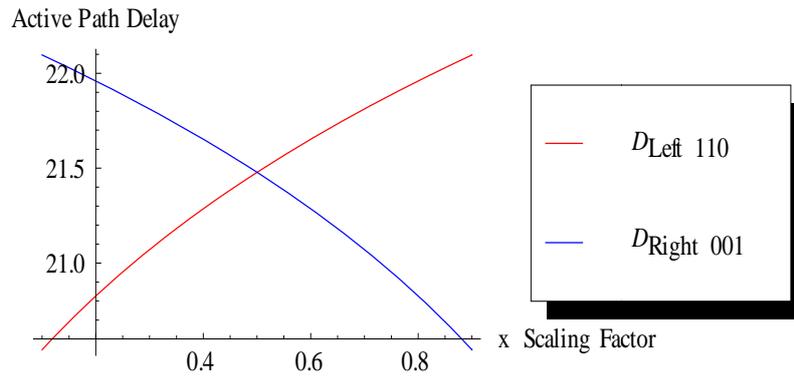


Figure 1.6: Left and Right circuit propagation delay for scaling factor x

Now let's optimize the scaling factors for the circuit shown in fig. 1.4 for the following input distribution. The input distribution is shown in fig. 1.7.

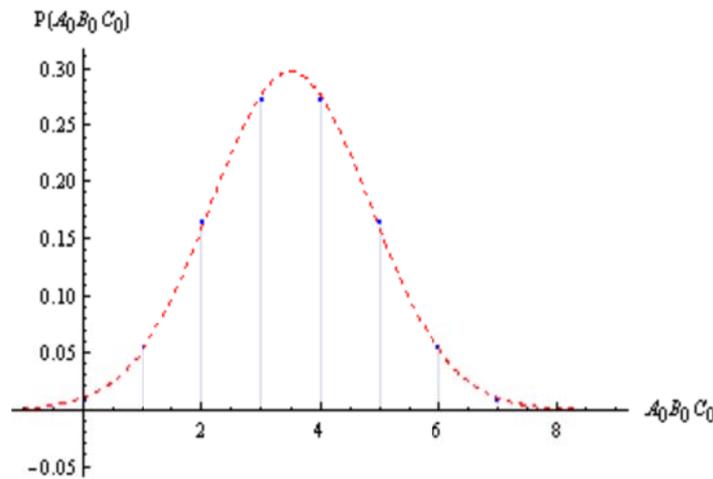


Figure 1.7: Full Adder input Distribution

Therefore, the expected completion time of the full adder is the average of the active critical path delays $D_0, D_1, D_2, D_3, D_4, D_5, D_6$ and D_7 . It equals equation (7). The unknown parameters in fig. 1.4 related to the device geometry are:

$C_{NAND1}, C_{NOR2}, C_{NAND3}, C_{NOR4}, C_{invh5}, C_{invh6}, C_{invh7}, C_{invh8}, C_{invh11}, C_{invh12}, C_{invh16}, C_{invh17}, C_{invh18}, C_{NOR15}, C_{AOI21B9}, C_{AOI21B10}, C_{SUMA13}$ and C_{SUMA14} .

$$D_{avg} = W_1 D_{001} + (W_0 + W_2 + W_4) D_{010} + W_6 D_{110} + (W_5 + W_3 + W_7) D_3 \tag{7}$$

The average completion time of the adder is minimized if these values are set such that,

$$0 = \frac{\partial D_{avg}}{\partial C_{NAND1}}$$

$$0 = \frac{\partial D_{avg}}{\partial C_{NOR2}}$$

$$0 = \frac{\partial D_{avg}}{\partial C_{NAND3}}$$

$$\vdots$$

$$\vdots$$

$$0 = \frac{\partial D_{avg}}{\partial C_{invh18}}$$

The Newton-Rapson method is used to find the circuit parameters (i.e. unknown capacitances above) when the above expression in equation vanishes.

III. RESULTS

The adder device sizing information computed for a bimodal distribution is shown in table 1. The device sizing used in traditional proposed realizations is also in the table. The calculations in the table are based on a circuit load capacitances C_{L1} , C_{L2} and C_{L3} of 100, 100 and 100 device width, which in turn affects the performance and power dissipation of a circuit since smaller transistors can use less power. Results show that if the input distribution is known, the circuit can be optimized with respect to size (transistor width). The input distribution technique allows for smaller transistor sizes for data paths that are not begin use and the larger transistor are used for data paths that have a higher probability of being used. This, in essence allows the circuit designer to boost performance and save power dissipation at the same time.

TABLE 1: Device capacitance in terms of transistor width

Device	Traditional Sizing Approach	Distribution Based Approach
C_{NAND1}	11.45	8.15
C_{NOR2}	14.95	4.75
C_{NAND3}	8.8	9.64
C_{NOR4}	11.4	2.5
C_{invh5}	26.16	17.23
C_{invh6}	39.42	17.3
C_{invh7}	15.42	20.2
C_{invh8}	22.93	13.5
$C_{AOI21B9}$	59.82	35.68
$C_{AOI21B10}$	25.99	41.74
C_{invh11}	58.49	44.78
C_{invh12}	59.0	44.93
C_{SUMA13}	17	9.36
C_{SUMA14}	17	17.07
C_{NOR15}	59.0	49.21
C_{invh16}	57.34	42.47
C_{invh17}	57.34	57.34
C_{invh18}	101	92.14

The performance and energy dissipation of the traditional and distribution based full adder device sizing approach is shown in table 2. The characteristics in this table were computed using logical effort and model parameters from SPICE simulation of the adder gates in the TSMC 0.18um process with a supply voltage V_{DD} of 1.8volts.

TABLE 2: The Performance and Energy of Traditional and Distribution Based Device Sizing

Circuit Characteristics	Traditional Approach	Distribution Based Approach	Characteristic Δ Difference	Characteristic percentage Difference
Average Completion Time [pSec]	465.727	432.673	33.054	7.1%
Expected Dynamic Energy Dissipation [pJoules]	1.90598	1.64919	0.25679	13.5%

One can further optimize the circuit by manipulating different sides of the circuit. We closely examine data paths that are more active and allow larger transistor widths within the active data path by trading size in one side of the circuit for another. This is accomplished by using the branching effort equation which is a part of calculating the logical effort. Table 3 below shows the results of distribution based device sizing with branching effort giving a 13% decrease in delay and 16% improvement in power dissipation.

TABLE 3: Device capacitance in terms of transistor width

Full Adder Side	Branching Effort	Nominal	Bimodal[]	Binomial[]
Left-Side	B0	3.889	9.22128	2.7882
	B0'	5.8335	3.12388	14.3051
Right-Side	B2	3.889	9.22128	2.7882
	B2'	5.8335	3.12388	14.3051
	B1	2.3335	2.3335	2.3335
	B1'	2.3335	2.3335	2.3335
	Average Propagation Delay		21.7258 (22.9488)	21.2617 (24.0977)
	Speedup		5.629%	13.39%
	Energy % Reduction		11.567%	16.78%

IV. CONCLUSION AND DISCUSSION

The performance and energy dissipation of self-timed circuits/systems depend on the circuit gate-level implementation, device sizing and input distribution. The device sizing approach used in previously proposed self-timed circuits is identical to that used for synchronous realizations. Therefore it is only optimized to minimize the propagation delay of all circuit signal paths. The performance and energy dissipation, i.e. average completion time and energy dissipation, of the proposed approach for a self-timed circuit is optimized, with respect to device sizing, for a given input distribution. It is less than realizations that do not consider this feature of the input space. This design process causes the active critical path delay of the circuit paths with the highest probability of being active to be less than the path delay in a realization that does not use input data. It also generates delay paths with larger propagation delay than that in previously proposed self-timed circuits design for path that are rarely used, i.e. paths associated with low probability. Both the performance and energy dissipation of self-timed circuits are reduced if the device sizing is optimized for the input distribution.

In short, performance is restricted by power and as chip density and frequency increase, synchronous designers try to figure out ways to deal with power/performance tradeoff. Can we get a better Energy Delay Product? Asynchronous designers do not have to deal with this tradeoff because of the nature of the logic design; we can use fewer transistors and operate at faster speeds.

Using self-timed circuits coupled with data profiling, one can exploit the natural properties --faster speeds, less transistors and path sizing-- to optimize power dissipation and performance. This gives us a superior Energy Delay Product. This technique is novel because there has been no research that alters the logical effort formula by manipulation the branching effort to trade delay in one part of the circuit for another. We can essentially control the flow of data by allowing highly probable paths to be sized larger and vice versa. With a 13% increase in performance and 16% decrease in power dissipation.

V. REFERENCES

- [1] Sutherland, I., Sproull, B. and Harris D.: "Logical Effort: Designing Fast CMOS Circuits," (Morgan Kaufmann Publishers, San Francisco, CA, 1999)
- [2] Pollack, F. 1999. New microarchitecture challenges in the coming generations of CMOS process technologies. *International Symposium on Microarchitecture*.
- [3] Moore, Gordon. "Cramming More Components onto Integrated Circuits," *Electronics Magazine* Vol. 38, No. 8 (April 19, 1965).
- [4] Rabaey, J.M., Chandrakasan, A. and Borivoje N.: "Digital Integrated Circuits: A Design Perspective, 2nd Ed.," (Prentice Hall, Upper Saddle River, N J, 2003).
- [5] Gonzales, R., AND Horowitz, M. Energy dissipation in general purpose microprocessors. *IEEE Journal of Solid-State Circuits* 31, 9 (September 1996), 1277-1284.
- [6] V. Oklobdzija, B. Zeydel, H. Dao, S. Mathew, R. Krishnamurthy, "Energy-Delay Estimation Technique for High-Performance Microprocessor VLSI Adders," Proceedings of the 16th IEEE Symposium on Computer Arithmetic (ARITH'03) 2003.
- [7] H. Zhang, P. Mazumder, "Design of a new sense amplifier flip-flop with improved power delay product," IEEE International Symposium on circuits and System, ISCAS 2005, p. 1262- 1265, Volume 2, 2005.
- [8] M. Ghasemzar, M. Pedram, "Power delay product minimization in high-performance 64-bit carry-select adders," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. Vol. 12, Issue 3, p. 235-244, 2004.
- [9] J. Choi, K. Lee, Design of CMOS tapered buffer for minimum power-delay product," IEEE Journal of Solid-State Circuits. Volume 29, Issue 9, p. 1142-1145. 1994.