A STUDY ON THE MAXIMUM SPEED AND RELIABILITY

ASSURANCE IN WAVE PIPELINE-BASED

COMBINATIONAL CIRCUITS

By

TAO FENG

Bachelor of Science
Northwest A&F University
Xi’an, China
1996

Master of Science
Oklahoma State University
Stillwater, Oklahoma
2003

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
December, 2006

A STUDY ON THE MAXIMUM SPEED AND RELIABILITY ASSURANCE IN WAVE PIPELINE-BASED COMBINATIONAL CIRCUITS

Thesis Approved:

Dr. Nohpill Park

Thesis Advisor

Dr. K. M. George

Dr.Venkatesh Sarangan

Dr. Guoliang Fan

Dr. A. Gordon Emslie

Dean of the Graduate College

ACKNOWLEDGMENTS

I sincerely thank my research advisor, Dr. Nohpill Park, for his excellent advice, guidance, and direction. Also I gratefully acknowledge all the members of my committee, Dr. K.M. George, Dr. Venkatesh Sarangan, and Dr. Guoliang Fan for their valuable advice. I would also like to give special thanks to my family for their love, support, and guidance: my wife, Melissa Feng; my parents, Guangsheng Feng and Sumei Wang; and my parents in-law, Mark and Nancy Torbert.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Clockless wave pipeline is a cutting-edge and innovative technology as an alternative to traditional pipeline, and a promising computing model moving towards ultra-high throughput and speed. The basic computational components of a clockless wave pipelines are data waves in association with request signals and switches as proposed in [23]. The key to success of clockless wave pipeline is how to coordinate and ensure the processing of datawaves throughout the pipeline in association with the request signals without relying on any intermediate access points under clocked control. Due to the complication of clockless operations, an efficient and effective method to model and analyze the confidence level (referred to as reliability or yield) of clockless operations within wave pipeline is exigently demanded. However, this has not yet been adequately addressed on an integrated level, such as datawaves in association with request signals, leaving this an innovative challenge. The confidence level is primarily determined by the reliability of the clockless orchestration of datawaves and their associated request signals under switch arrangements for datawave alignment.

In this regard, out-of-orchestration between datawaves and request signals, referred to as delay fault, is the major concern in assuring and optimizing the reliability of the system. New faults, referred to as Intra-Wave Fault [13], Inter-Wave Fault [14], and further Request Signal Faults [15], have been proposed and addressed in this dissertation. The proposed new fault models help reveal fundamental yet essential characteristics of the reliability of the clockless wave pipeline, and enable efficient and effective reliability assurance and optimization. The intra/inter-wave faults are used as the primary drivers to maneuver the evaluation method of the fault impacts on the reliability with respect to extensive delay-sensitive design parameters such as delay distribution, delay variation, crosstalk noise, intra/inter-wave of datawaves, request signal, and other elements. This dissertation specifically addresses and resolves the followings for clockless wave pipeline: extensive and practical clockless-induced datawave fault modeling, yield and reliability modeling, assurance and optimization, and clockless-oriented fault tolerant design methods. The proposed methods will establish a sound and adequate theoretical foundation for development of innovative yet practical test/diagnosis/fault-tolerant design methods in the early design stages of clockless wave pipeline-based logic.

All conventional wave pipeline techniques have been practiced within the long-believed theoretical and technological constraint limited by $D_{max} - D_{min}$, where D_{max} and D_{min} stands for maximum and minimum path delay in a combinational circuit, respectively; however, we concluded that this limit is a very passive and loose constraint and our proposed new wave pipeline technique can further push the limit farther beyond that level.

A new approach to dramatically enhance the circuit speed is proposed in the context of wave pipeline. The proposed approach is based on the fact that there is more temporal space in combinational circuits to push the frequency of datawaves by allowing superimposition between adjacent datawaves. By allowing the datawave-superimposition, still with the high assurance and certainty of mutual exclusion between adjacent datawaves, a faster clock frequency with the clock cycle equals to $D_{max} - D_{min} - \delta_{push}$ is achieved in the proposed new wave pipeline, where δ_{push} stands for the amount of space that can be further pushed between adjacent datawaves.

Also, a temporal/spatial bounded system delay model, which expands the current bounded path delay model into the system level and into spatial area, is proposed in order to fully address the practical circuit-delay issues and adequately characterize the datawave processing in the context of wave pipeline. The complete set of paths existed in the circuit are extracted out and relocated into a 2-dimensional coordinate system, with the y-axis representing the path delays and x-axis representing the order of paths. Each datawave can be represented by a polygon within the system, and the Global Maximum Polygon (GMP) is the very last polygon in the system. A new output register architecture with delay buffers is designed and presented in order to achieve the resulting performance and reliability of the proposed approach.

The performance advantage of the proposed technique over conventional wave pipeline is theoretically demonstrated with the cost for a higher reliability requirement. The simulation and verification of the proposed new wave pipeline are experimented against the ISCAS benchmark circuits. The results obtained have shown a faster frequency circuit achieved by the proposed new wave pipeline technology. A MTTF (mean time to failure) model and a reliability model for pipelining circuits are theoretically proposed and applied to conventional pipeline, conventional wave pipeline, and the proposed new wave pipeline. They are also experimented against ISCAS benchmark circuits. The simulation results have shown that the new proposed wave pipeline has a much higher reliability requirement compared with conventional pipeline and conventional wave pipeline.

The specific objectives of this dissertation are:

1. To establish clockless wave pipeline-specific fault models as the primary drivers for yield and reliability modeling, assurance and optimization.

2. To demonstrate a theoretical yet thorough characterization and parameterizations of the representative clockless wave pipeline-specific faults (intra-wave fault, inter-wave fault, and request signal fault).

3. To demonstrate fault tolerant design methods with respect to datawave faults and request signal faults. Also, to derive a novel yet solid yield model for the proposed fault tolerance for efficient and effective manipulation of extensive fault tolerance.

4. To propose the new bounded system delay model based on the current bounded path delay model, and to further propose the new wave pipeline-based architecture, by allowing the datawave-superimposition between adjacent datawaves, which can theoretically yet practically push the performance of of wave pipeline to its limit.

5. To conduct the reliability analysis of the proposed new wave pipeline, and compare it with the reliability of conventional pipeline and conventional wave pipeline.

The following sections in this chapter provide a general introduction to and review of the current wave pipeline techniques and others related.

1.1 Conventional Pipeline

In computing, a pipeline is a set of data processing elements connected in series. The output of one element is being processed as in the input of the next one. The elements of a pipeline are executed in parallel or in time-sliced fashion.

In conventional pipeline, there are registers or latches between any two stages. At any instant of time, there is at most one data active in a single stage as shown in Figure1.1.

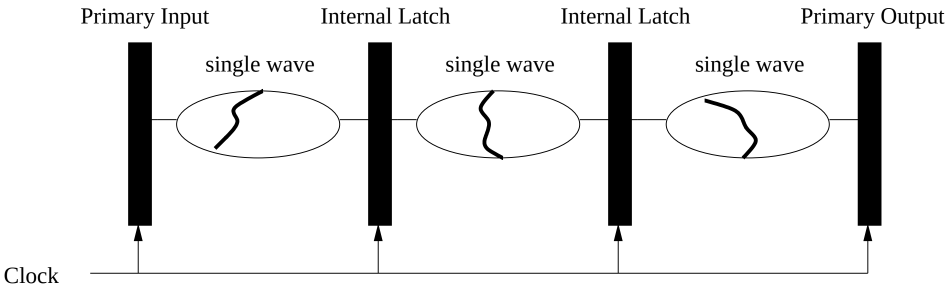


Figure 1.1: Conventional pipeline

As a concrete example of conventional pipeline, instruction pipeline is a realization of linear synchronous pipeline in which performance is improved through instruction-level parallelism by allowing to start execution of an instruction before the previous ones already in the pipeline are finished. The architecture of conventional synchronous linear instruction pipeline is shown in Figure 1.2.

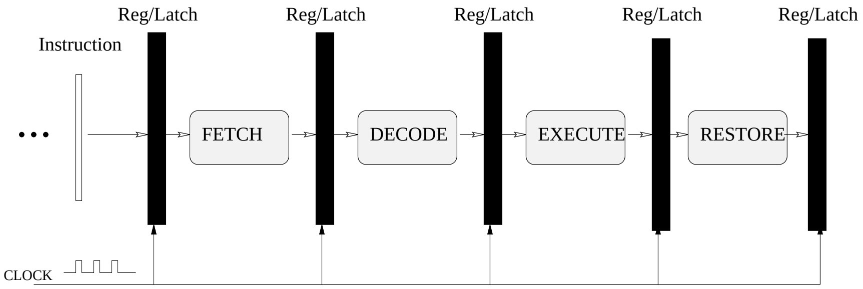


Figure 1.2: Conventional synchronous linear instruction pipeline

1.2 Wave Pipeline

Wave pipeline is a pipeline processing technique that can increase the throughput without increasing internal storage space and power consumption [18, 48, 51]. Multiple datawaves can propagate through the wave pipeline from the PI (Primary Input) to the PO (Primary Output) simultaneously without internal latching. It can ideally achieve the theoretical maximum performance, and draws lots of attentions in the industry today.

The wave pipeline technology was introduced by Cotton [7] in 1969 in order to increase the throughput of the pipeline. This pipeline was also referred to as the maximum rate pipeline. Cotton observed that the rate at which logic can propagate through the circuit depends not on the longest path delay, but on the difference between the longest and the shortest path delays. Hence, by removing the internal latches, balancing all the logic paths within the circuit, and allowing multiple waves, the logic bits can propagate through the combinational circuit in each stage without any delay for register-latching. Thus, a clock period much shorter than the one for the longest path in the circuit can be obtained. To achieve this it must be guaranteed that no fast datawave overruns a previous slow datawave that would result in data loss. Hence, it is critical and the major challenge in wave pipeline design that all paths in the combinational logic be well balanced. The balancing of paths can be implemented in two ways: rough tuning and fine tuning. Rough tuning is to equalize path delays by inserting delay elements to the fast path, and fine tuning is by adjusting gate delays to achieve path equivalence.

Wave pipeline has been extensively researched in both academic and industrial sectors [27], and three to four times of speed-up has been reported in [18]. A few research efforts of wave pipeline have been focused on synchronous wave pipeline (SWP), which is a wave pipeline using clock to control the latches operating in parallel. Many works have been done to research and enhance SWP into an effective and reliable computer technology [76], such as modeling and analysis of correct timing [51], [17]; development of logic synthesis and computer-aided design (CAD) tools for SWP circuits [74],[58]; and development of new wave pipeline-specific circuit [39].

A conventional synchronous instruction wave pipeline architecture is shown in Figure1.3. Each active instruction in the pipeline can be regarded as a wave of input data. By removing the internal register-latches, multiple instructions can propagate through the same logic stage simultaneously at a higher clock frequency. Note that this wave pipeline is still synchronous because the primary input (PI) and the primary output (PO) operate synchronously under the control of the clock.

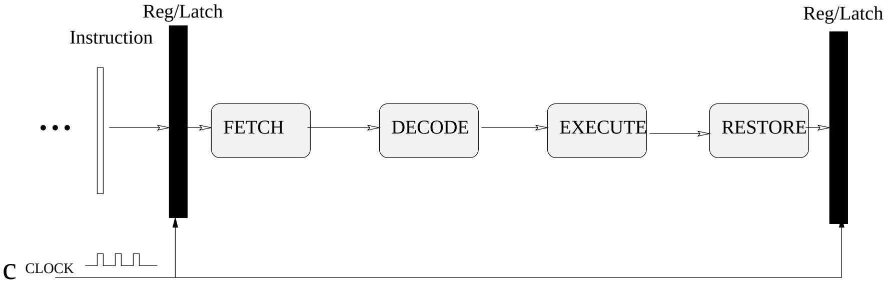


Figure 1.3: Instruction wave pipeline

In conventional pipeline, there are registers or latches between any two stages. At any instant of time, there is at most one data active in a single stage as shown in Figure1.1. Whereas, in wave pipeline, the internal registers or latches are removed, and multiple data can be active in the same logic stages simultaneously as shown in Figure1.4. The clock cycle time in conventional pipeline is determined by the maximum stage delay, that is $T_{ck} \geq D_{max}$ (where T_{ck} is the clock cycle time and D_{max} is the maximum stage delay). In wave pipeline, the time constraint is more stringent. The clock cycle time is determined not by D_{max} but by the relative difference between D_{max} and D_{min}, that is $\frac{T_{max}}{N} < T_{ck} < \frac{T_{min}}{N-1}$ [19] (where N is the number of waves in the circuit, T_{max} is the maximum path delay and T_{min} is the minimum path delay.

A wave pipeline can be built either in a synchronous or an asynchronous manner. Synchronous wave pipeline uses a clock signal to synchronize the movement of datawave bits. It

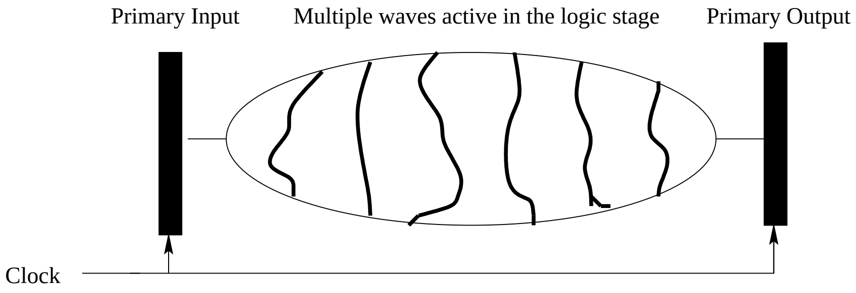


Figure 1.4: Typical synchronous wave pipeline

has been successfully deployed in several commercial processors such as the floating point unit of IBM $360/91$ [1] and external caches in the HP $PA8000$ [33].

1.3 Clockless Wave Pipeline

Asynchronous wave pipeline uses request and acknowledgement signals (or only a request signal) instead of a global clock to serve as a reference signal. Asynchronous wave pipeline is relatively more difficult to deploy than synchronous wave pipeline due to its explorativeness and a few technological hurdles as mentioned before.

Clockless wave pipeline circuits have only been experimented in non-commercial sectors such as the two-phase clockless wave pipeline, which uses only a single request signal line [26, 23]. The specific architectural model investigated in this dissertation is the two-phase clockless wave pipeline [23]. To the best of our knowledge, it is known to be the best in the context of clockless wave pipeline.

The *two-phase asynchronous wave pipeline* was proposed by Hauck and Huss in [23] as shown in Figure(1.5), which has the two-phase operation by alternating positive and negative level-sensitive switches, and it employs a request signal only, no acknowledgement signal. Note that the asynchronous wave pipeline with request signal only is referred to as clockless wave pipeline in this dissertation.

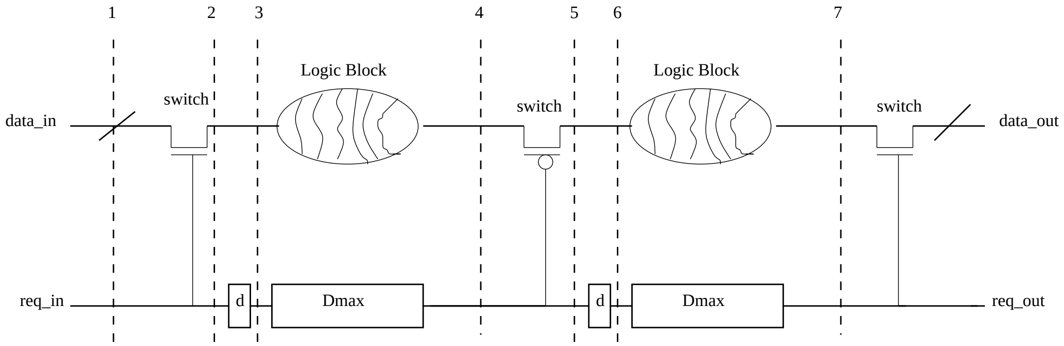


Figure 1.5: Two-phase asynchronous wave pipeline [23]

As shown in Figure(1.5) [23], two types of switches, namely positive and negative switches, logically and physically partition the circuit into several pipeline stages for datawave progress alignment purposes, and a request signal controls the switches. A pair of a datawave and a request signal level (either a high or low pulse of the request signal) enters the clockless wave pipeline at the same time in full association, and they must stay so throughout the propagation of the datawave.

The switch can be opaque or transparent to the datawaves. If opaque, the switch latches the datawave and, if transparent, the datawave passes the switch without latching. Specifically, N-type switches (negative switches) are made to be opaque to the datawaves associated with a low request signal, and transparent to those associated with a high request signal. Symmetrically, P-type switches (positive switches) are made to be transparent to the datawave associated with a low request signal, and opaque to those associated with a high request signal. The propagation of the datawave within a pipelined circuit is intermittent. Once the datawave enters the circuit, it is assigned a request signal, at either high or low level. Then, by the design rule, as defined, the datawave associated with a high request signal propagates through N switches without stopping for alignment, but has to be latched at P switches. Symmetrically, the datawave associated with a low request signal propagates through P switch without stopping, but has to stop and be latched at N switches. Figure(1.6) depicts the case of a datawave passing through a transparent switch, and Figure(1.7) the case where a datawave is aligned at an opaque switch.

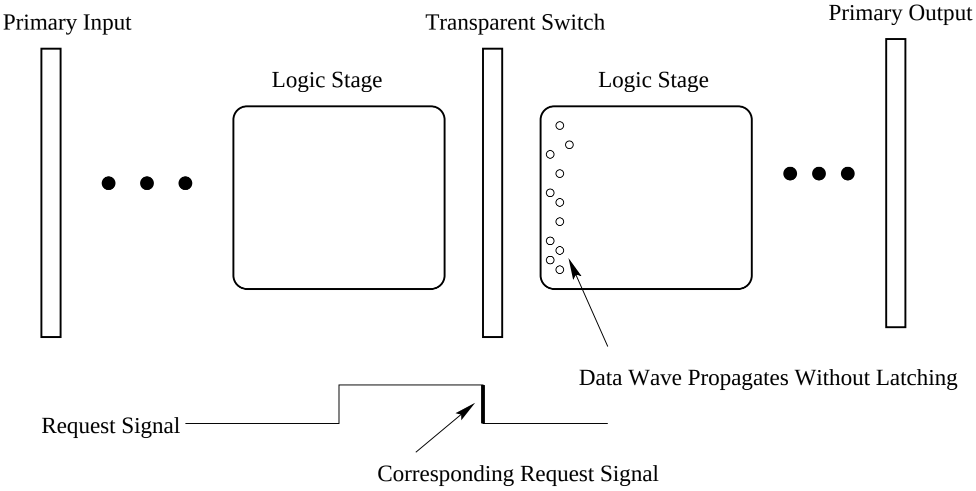


Figure 1.6: Transparent switch

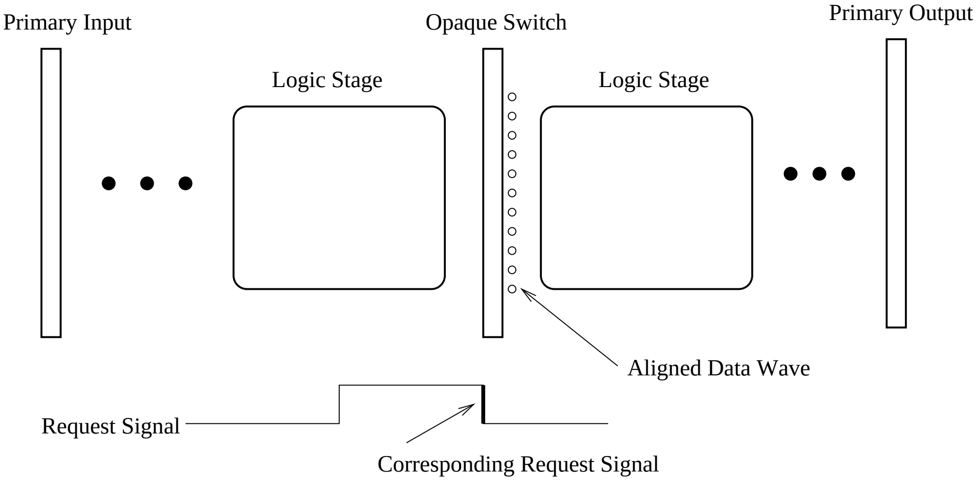


Figure 1.7: Opaque switch

Ideally, there are multiple datawaves populated and propagating through the combinational circuit simultaneously, and traditional delay-fault testing, modeling and assurance techniques are not readily able to handle and evaluate the faults of successive transitions at the datawave level in the circuit. Therefore, assurance and optimization of delay-oriented yield and reliability is exigently demanded, and a key to the success of clockless wave pipeline technique. Ideally, all path delays from PI to PO are to be equally or near-equally balanced. However, equal-balancing of path delays is hard to realize even with the help of extensive tuning [68] for various fabrication and runtime variations (such as power consumption, thermal distribution and design errors to mention a few) to account for delay variations. Furthermore, clock skews due to variations in the rise/fall time and the setup and hold time of the storage elements will limit the clock frequency to within an increase only by a factor of 2 to 3, even by using the best known design tuning method [75].

Two most representative delay-fault models for synchronous wave pipeline are as follows. These fault models provide an important basis and guidance to theoretical characterization and parameterization of delay faults for clockless wave pipeline in this dissertation as well as that of wave pipeline in general. There have been a few works proposed on synchronous wave pipeline design for reliability [57, 45, 12, 77], in which testing, characterization and parameterization of datapath-oriented delay faults were proposed. However, there is no adequate and extensive characterization and parameterization of the faults and yield for synchronous wave pipelines, nor for clockless wave pipelines either. This is a great hurdle for reliable wave pipelines to be realized. Therefore, it is imperative to develop formal methods for adequate and extensive modeling of delay-faults precisely from a datawaves' standpoint and at an integrated datawave level to be able to assure and optimize the yield in early design stage. Without such theoretical reliability assurance and optimization methods, ones precisely on integrated system level with focus on the datawaves, there is no efficient and effective way of designing a reliable wave pipeline with fault tolerance.

1.4 Other Asynchronous/Clockless/Delay-Insensitive Circuit Design Techniques

Today, ever increasing attention is being paid to clockless circuits for the following outstanding advantages: saving on circuit area due to no global clock-timing; reduced power and heat dissipation; eliminating clock skew and global timing constraints; potentially improved performance over synchronous worst-case; and potentially more efficient migration and adaptation to new technology.

Various techniques have been proposed and researched to realize clockless (or asynchronous, delay-insensitive) circuits. Some of the most outstanding techniques are reviewed in the following.

Null-Convention-Logic is a technique to completely cover the entire range of signal state by introducing null state in addition to conventional true and false states. Having null state incorporated, NCL defines a new Boolean logic such that if any of inputs to a device is sensed as inactive (as defined by null state), its result outputs an inactive or null signal. This way, data signals can be self-synchronized by being able to discriminate active (or valid) and inactive (or null) data without reference to a clock (i.e., asynchronously) [44, 11, 40, 60, 36].

Self-Resetting is a technique to control data-in and out of latches via a self-resetting feedback signal self-generated by combinational circuit instead of via a clock-controlled synchronization signal. In entry-latch, a reset and reset-detection circuitry is placed to coordinate data-in into the combinational circuit, and a feedback line is connected from exit-latch at the end of the combinational circuit [41, 69, 46, 53, 32, 39].

Opportunistic-Time-Borrowing is a semi-asynchronous technique to maximize the utilization of clock signals. Having clock-latch and no-clock-latch laid out alternately for data-signal synchronization, clock-latch (no-clock-latch) borrows a certain extent of no-clock (clock) period of time from subsequent no-clock-latch (clock-latch) as needed if the subsequent no-clock-latch (clock-latch) requires less no-clock (clock) period of time. To maintain the original functionality intact, any borrowed clock (no-clock) period of time is supposed to be paid back [21, 3, 9, 37, 62].

Micro-pipeline is a wave-pipeline technique to overcome the theoretical limit in speed and throughput of conventional clock-synchronous pipeline technique. The basic asynchronous operation is accomplished by a handshaking protocol in which two main signal lines (i.e., request and acknowledgement) coordinate and control datawaves without relying on any main clock. This technique is more widely understood and practiced for its simplicity of control [22, 55, 35, 64, 61, 63, 4, 47]; however, the overhead for heavy signaling for handshaking and only marginal performance gain opens an opportunity for the clockless wave pipeline technique under investigation in this chapter to achieve ultra-high performance through more technical challenge.

Regardless of the outstanding benefits asynchronous circuits can offer, there are still a few problems to be resolved before asynchronous circuits are extensively deployed in the commercial market. The following are the hurdles for asynchronous circuit designers to address and overcome: signaling overhead and complication for handshaking protocol in generic-type asynchronous circuits such as micropipelines [63]; less availability of CAD tools and standard for design, testing, and manufacturing, which hinders the asynchronous circuit design paradigm from extensive deployment and expansion [49, 50, 59]; potential algorithmic and hardware overhead and complication to resolve hazards for asynchronous and non-monotonic processing; lack and difficulty of theoretical method development to adequately address, assure, and optimize the performance; and no adequate theoretical modeling and assurance of reliability of asynchronous process on integrated system SoC level. The modeling, assurance and optimization technique for reliability in early design phase is key to the success of asynchronous circuits. In this context, this dissertation gives a great emphasis on the reliability modeling, assurance, and optimization of clockless circuits integrated at the datawave level in the early design phase.

CHAPTER 2

FAULT MODELS AND YIELD MODELING/ASSURANCE

In order to efficiently and effectively depart from the delay fault models of conventional synchronous wave pipelines, we propose two new fault models. The proposed Intra-Wave Fault Model is based on the Pulse Fault Model as proposed in our previous work [13]. Based on the intra-wave fault model, we further propose the Inter-Wave Fault Model [16] to provide a comprehensive yet essential understanding of the fault mechanism of a clockless wave pipeline. The proposed fault models are used as the theoretical basis for the proposed yield modeling, assurance and optimization method. Also, we take into account a synergistic fault model, in which specifically, delay faults on request signal lines are modeled together with the datawave delay faults, because correct association between datawaves and their request signals primarily determines the delay-oriented yield and reliability.

2.1 Intra-Wave Fault and Yield

The path delay of a partial path can be represented as x. Suppose there are n such partial paths (i.e. x_1, $\dots x_n$) and assume those partial paths are assumed to be well balanced in the clockless wave pipeline under investigation. Also, without loss of generality, we may assume the path delay for x_i follows a Gaussian distribution with a mean μ and a standard deviation σ as follows $x \sim N(\mu\sigma^2)$, relying on each path being composed of at least several logic elements contributing delay. Let $\bar{x}$ be the sample mean, and s be the sample standard deviation.

Then, μ and σ can be substituted for with $\bar{x}$ and s, respectively, because $\bar{x}$ and s are unbiased estimators of μ and σ $x \sim N(\bar{x}s^2)$. Suppose $f_x(x)$ is the *p.d.f.* (probability density function) of x as follows, $f_x(x) = \frac{1}{\sqrt{2\pi}s}e^{-\frac{(x-\bar{x})^2}{2s^2}}$. By definition, the intra-wave fault is the probability that some bits in the datawave proceed too fast and then overstep their associated request level (i.e., the probability of a setup time fault); or some bits in the datawave are so slow that they lag behind the associated request level (i.e. a hold time fault) leaving some bits out of the range of the supposed-to-be-associated request level interval. Figure 2.1 shows the rate of the intra-wave fault as the integral of the shaded areas.

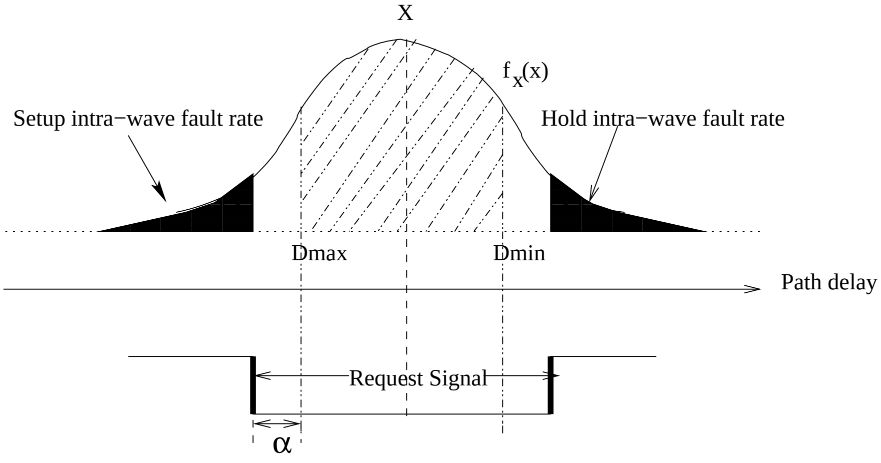


Figure 2.1: Intra-wave fault rate

The request signal and all the data bits enter the circuit at the same time, and for proper operation the request signal should be slower than the slowest bit of the datawave, and reach the switch after the slowest bit. Likewise, the previous request signal should reach the switch before the fastest bit of the associated datawave. The data skew (represented by $\triangle$ in Figure 2.3) is supposed to be covered by the associated request level properly. Therefore, the coverage of $\triangle$ by the request level—i.e., the relative position between the datawave and its associated

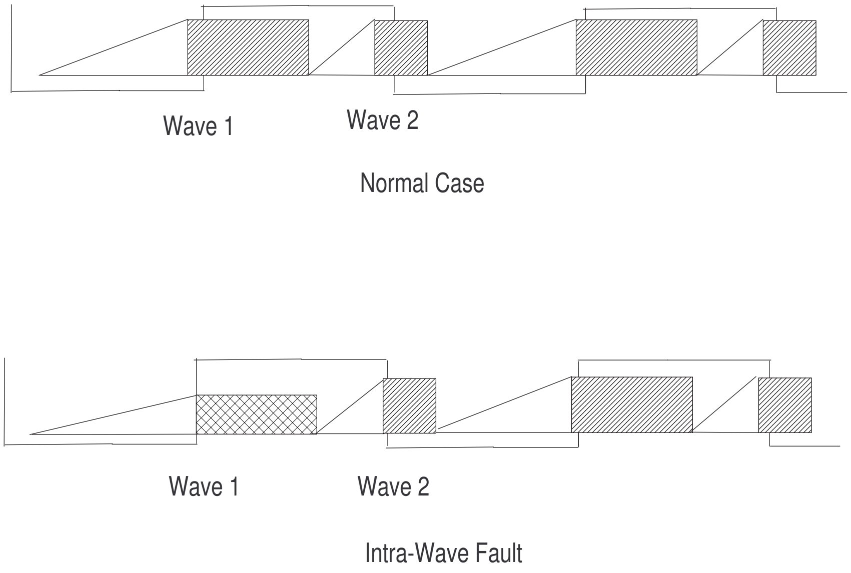


Figure 2.2: Detailed intra-wave fault

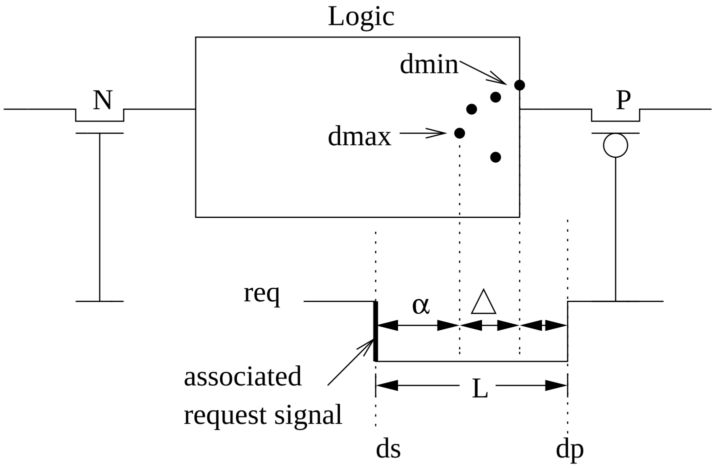


Figure 2.3: Relative position between the request signal and datawave

low or high request level—may influence the intra-wave fault rate to a great extent.

α refers to the difference of propagation time between the slowest bit of the datawave and the request level as shown in Figure 2.3. Thus, the associated request signal propagation delay (denoted as d_s) can be expressed as $d_s = d_{max} + \alpha$. The propagation delay of the request signal pulse through the switch (denoted as d_p) is $d_p = d_{min} - (L - \alpha - \triangle)$. Thus, $L = d_s - d_p$. The intra-wave fault rate at switch i is as follows:

$$P_i = 1 - \int_{d_{min_i}-(L-\alpha_i-\triangle_i)}^{d_{max_i}+\alpha_i} \frac{1}{\sqrt{2\pi}s} e^{(-\frac{(x-\bar{x})^2}{2s^2})} dx \tag{2.1}$$

Placing n switches in the circuit, each of which with P_i, the intra-wave fault rate at each switch where $1 \leq i \leq n$, the total intra-wave fault rate (P_{total}) is as follows:

$$P_{total} = 1 - \prod_{i=1}^{n}(1 - P_i) = 1 - \prod_{i=1}^{n}(1 - \int_{d_{min_i}-(L-\alpha_i-\triangle_i)}^{d_{max_i}+\alpha_i} \frac{1}{\sqrt{2\pi}s} e^{(-\frac{(x-\bar{x})^2}{2s^2})} dx) \tag{2.2}$$

Therefore, the overall yield $Y = 1 - P_{total}$. Also, note that by using the proposed yield model, at each switch i, we see that theoretical optimum is achieved by setting α_i to $\frac{1}{2}(L - \Delta_i)$.

2.2 Inter-Wave Fault and Yield

In order to extensively assure and optimize the reliability of the clockless wave pipeline, delay faults between datawaves must be characterized and parameterized beyond the scope of the Intra-Wave Fault. A preliminary result was reported in [14]. The proposed inter-wave fault model reveals the effect of the proposed intra-wave fault in association with other primary delay-oriented factors, such as the request signal and inter-datawave relation.

Delay faults can be viewed and modeled either vertically or horizontally at the moment when the bits of a datawave reach the opaque switches. Vertically, datawave-delay faults can occur within the scope of a datawave if its associated request signal goes out of association;

this has the effect of an intra-wave fault. Horizontally along each data-path, if path delay faults occur, then data bits of two adjacent datawaves may collide and get invalidated across the scopes of the datawaves; this type of fault is referred to as Inter-Wave Fault, as proposed here.

The inter-wave fault observes the bits of datawaves at each switch, and thus facilitates the controllability and observability of datawave testing. Figure 2.4 demonstrates a snapshot of an occurrence of a delay fault hit on two adjacent bits on a data path (horizontally)—that is, an inter-wave fault.

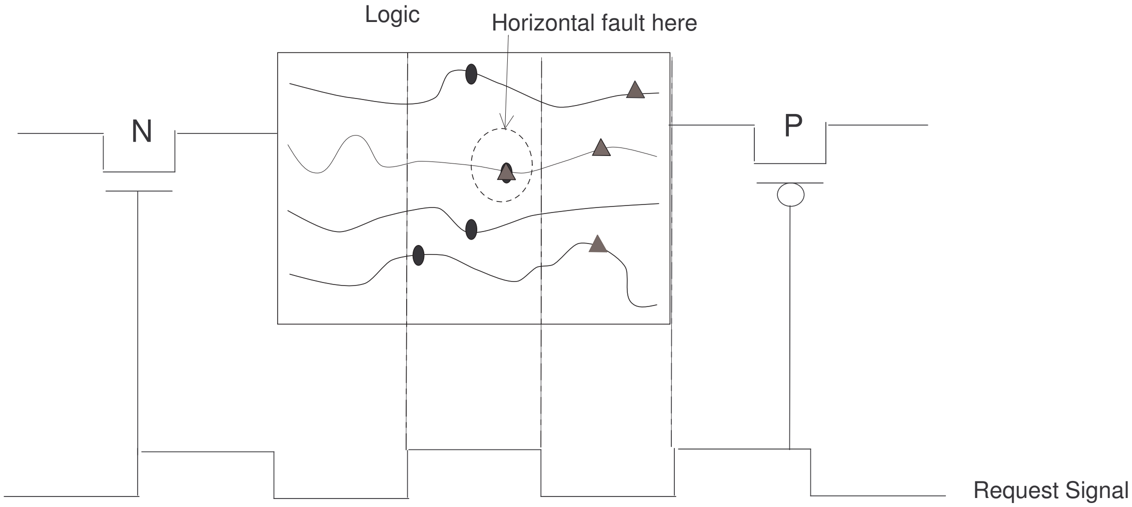


Figure 2.4: Inter-wave fault as horizontal delay fault

Figure 2.5 demonstrates a detailed occurrence of an inter-wave fault in a data signal pulse diagram between two datawaves (i.e., wave 1 and wave 2). Some bits of datawave 1 are overlapped with some other bits of datawave 2. Consequently the shape of datawave 1 becomes smaller than normal indicating that these bits are invalidated and lost. Notice that as shown in Figure 2.5, all the bits of datawave 1 arrive at the observation point (i.e., a switch) before the request signal arrives. Hence, the shrink in the shape of the datawave is not caused by out-of-

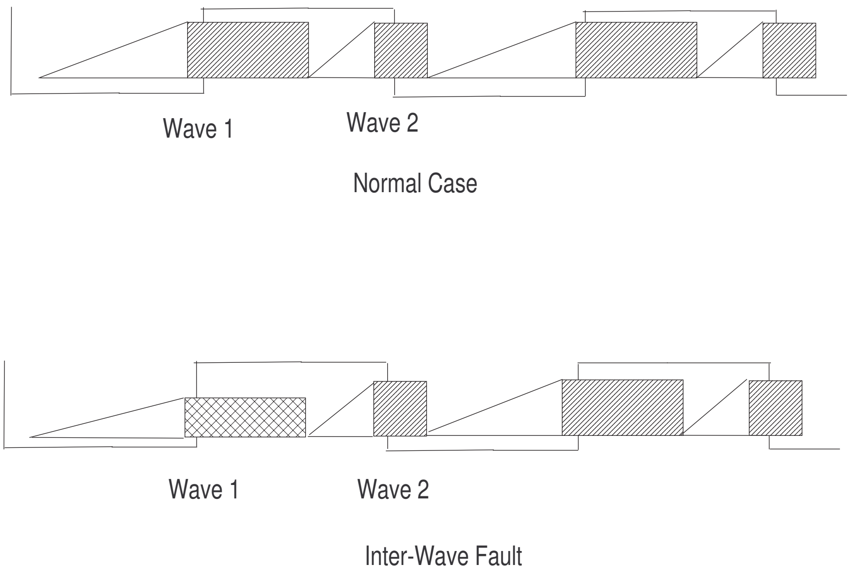


Figure 2.5: Signal transition diagram of inter-wave fault

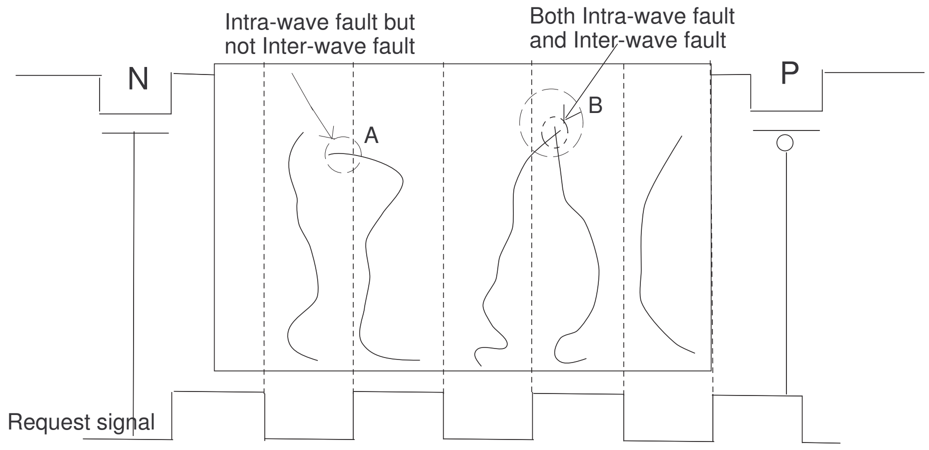


Figure 2.6: Inter-wave fault as a subset of intra-wave fault

association between the datawave and its request signal, but by the as-observed overlapping of the bits in opposite transition. This distinguishes the inter-wave fault from the intra-wave fault, which is an out-of-association between a datawave and its supposed-to-be-associated request signal.

Notice that according to the theoretical definition of intra and inter-wave faults, we may state that existence of an intra-wave fault is a necessary but not sufficient condition for an inter-wave fault to occur (i.e., inter-wave fault is a subset of intra-wave fault). Therefore, an accurate modeling and analysis of the co-relation and co-effect between the two datawave faults is needed for efficient and effective clockless wave pipeline design decisions and performance optimizations. If an inter-wave fault occurs, some of the faulty bits have to be out of association with its request level. This is illustrated in Figure 2.6. At A, the case where the intra-wave fault doesn't cause any inter-wave fault. At B, the two datawaves collide and cross each other to cause an inter-wave fault, where some bits of the datawave on the left hand side run too fast and break the association with its request signal and moreover collide and cross its neighboring datawave on the right had side.

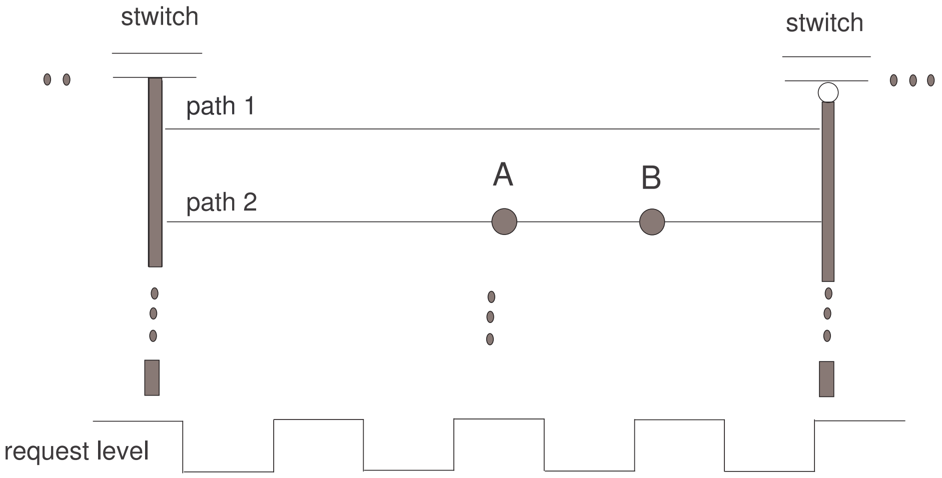


Figure 2.7: Two bits, A (in falling transition) and B (in rising transition) on the same datapath

In Figure 2.7, we show two bits A and B on a data path, each of which belongs to a different datawave respectively and those two datawaves are adjacent in their sequence. Suppose bit A is the front-most bit in the datawave which is supposed to propagate behind the datawave with bit B as its rear-most bit. Then, a collision between bits A and B is equivalent to the Inter-Wave

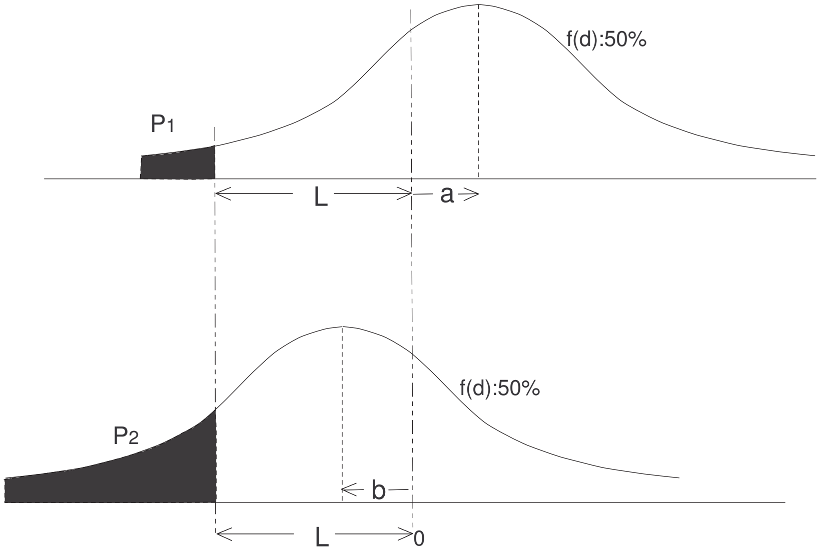


Figure 2.8: Probability for two bits on the same datapath causing an inter-wave fault

Fault by our definition. Without loss of generality, we make an assumption that the probability for each switch to observe an Inter-Wave Fault is independent from each other since inter-wave fault depends on the path delays through each partial circuit in between switches and the partial circuits are functionally and physically independent. Under the assumption, we can assume a random variable x with a Gaussian distribution to represent the delay of bit A; and another random variable y also with a Gaussian distribution to represent the delay of bit B. Then based on x and y, we can further assume a random variable $d = x - y$, and d also follows a Gaussian distribution without loss of generality. Thus, d represents the status of collision between bits A and B such that: (a) if $d > 0$ it indicates that there is no collision between bits A and B; (b) if $d = 0$ it indicates that bits A and B just collide at the intersection of the two datawaves they belong to; and (c) if $d < 0$ it indicates bit A surpasses bit B resulting in overlap between the two datawaves they belong to, that is the Inter-Wave Fault we propose.

Given the above random variables assumed, Figure 2.8 demonstrates the probability density functions for inter-wave fault. Notice that the p.d.f. on the top represents the probability distribution to have the two bits A and B getting farther from each other due to an arbitrary delay variation as indicated by the shift of the average point in the positive direction from the original point at 0 by a; while the p.d.f. on the bottom represents the probability distribution to have the two bits A and B coming closer to each other due to another arbitrary delay variation as indicated by the shift of the average point in the negative direction from the original point at 0 by b. The L indicates the maximum difference, or the maximum value d, between bits A and B, either in positive or negative direction. Also, note that each p.d.f. accounts for 50% of the entire probability because without loss of generality we can assume that the probabilities for the two bits to get farther or closer are equal unless otherwise specified. The dark area in the p.d.f. on the top (i.e., P_1) represents the probability that bit B is going beyond the left-edge of its request level in the negative direction and thus will intersect and overlap with bit A; and vice versa (i.e., P_2) in the the p.d.f. on the bottom. Thus, it shows that the farther the two bits get away, the smaller is the probability for inter-wave fault; and vice versa. Therefore, summation of the dark areas of each p.d.f. represents the total probability for the inter-wave fault (i.e., $\Pr$(Inter-wave fault/switch) at each switch. The probability for inter-wave fault in the p.d.f. on the top is $f_x(d_1) = \frac{1}{\sqrt{2\pi}S_{d_1}} e^{-\frac{(d_1-\bar{d_1})^2}{2S_{d_1}^2}}$ and that in the p.d.f. on the bottom is $f_x(d_2) = \frac{1}{\sqrt{2\pi}S_{d_2}} e^{-\frac{(d_2-\bar{d_2})^2}{2S_{d_2}^2}}$ where note that d_1 and d_2 represent the d in each p.d.f.). Thus, P_1 and P_2 are as follows: $P_1 = \int_{-\infty}^{-L-a} \frac{1}{\sqrt{2\pi}S_{d_1}} e^{-\frac{(d_1-\bar{d_1})^2}{2S_{d_1}^2}} dd_1$ and $P_2 = \int_{-\infty}^{-L+b} \frac{1}{\sqrt{2\pi}S_{d_2}} e^{-\frac{(d_2-\bar{d_2})^2}{2S_{d_2}^2}} dd_2$.

Therefore, the total probability $\Pr(Interwave\ fault/switch) = P_1 + P_2$. Suppose there are n such paths in a partial circuit, then the total probability for the inter-wave fault in each partial circuit (i.e., $\Pr$(Inter-wave fault/partial circuit)) can be expressed as follows: $\Pr(Inter$-

$wave\ fault/partialcircuit) = \prod_{i=1}^{n} P_i$ because delays on each data path are independent. Further, suppose there are m such partial circuits and let the $\Pr$(Inter-wave fault/partial circuit) be P_j, then the total probability for the inter-wave fault in the whole clockless wave pipeline can be expressed as $\Pr(Interwave\ fault/pipeline) = 1 - \prod_{j=1}^{m}(1 - P_j)$. Therefore, the resulting yield $Y = 1 - \Pr(Interwave\ fault/pipeline)$.

2.3 Request Signal Fault and Yield

Correct operation of a clockless wave pipeline is mainly determined not only solely by intra/inter-wave fault but also by the request signal faults. Datawaves are guided by the request signal; aligned at and by the P- and N-switches; and hence the request signal determines the correct propagation of datawaves. Therefore, the request signal is the primary signal in control of clockless wave pipeline processing. Any incorrect control over datawaves by request signal may cause a fatal error leading to data corruption. Hence, ensuring a fault-free request signal is a key to the successful realization of reliable clockless wave pipeline.

The Request Signal Fault of our interest is a glitch hit on the request signal due to crosstalk or power pulse noises [2, 17, 30]. In [23], it was reported that glitches can hit on the request signal created by a request signal generator. A glitch on a request signal breaks the datawave in association with it (either transient, permanently, or intermittently); the faulty request signal may instantaneously break the datawave at the instant the glitch hits, and let the head portion of the broken datawave propagate through the next switch where the broken datawave will be released earlier than the normal switch-blocking period of time. This broken datawave will consequently proceed towards the datawave ahead of that and result in an inter-wave fault.

Likewise, the tail portion of the broken datawave may suffer from any glitch hit on the request signal associated with the following datawave, also resulting in another inter-wave fault. The yield with consideration of the inter-wave fault caused by request signal faults can be evaluated by extending the proposed inter-wave fault model in the previous section. In this context and based on the proposed fault models and yield modeling/assurance methods, a solid method to ensure a fault-free request signal is proposed in the following section.

CHAPTER 3

FAULT TOLERANT DESIGN FOR REQUEST SIGNAL FAULTS

We propose two fault tolerant designs for clockless wave pipeline [15]: one targeting at request signal faults because the request signal is the most critical element for correct control over the datawaves and overall computation; and the other targeting at datawave faults because intra/inter-wave faults may still hit on the datawaves due to delay variations independent of request signal faults. The fault tolerant design for request signal will be addressed in Chapter 3, and the one for datawaves will be addressed in Chapter 4. The efficiency and effectiveness will be demonstrated by the proposed novel reliability and yield modeling and assurance techniques.

Request signal is a crucial control part of the pipeline. In practice, a request signal is highly sensitive and vulnerable to electronic crosstalk or power pulse noise (referred to as glitch). As the technology allows ultra-smaller device geometries, millions of closely spaced interconnections, and higher switching speeds, electronic crosstalk noise is very likely to occur; and it appears to be a major problem in the development of next generation high-speed integrated circuits [42][72][78] [79]. In general, the request signals are caused by various electrical and environmental factors such as voltage, temperature, humidity, etc. Crosstalk noises on request signals could be permanent, transient, or intermittent. Any type of glitch on the request signal is deleterious and may result in fatal datawave corruption.

In this chapter, inspired by the intrinsic properties of the two-phase pipeline, a unique, simple, but effective and low overhead technique is proposed to fault tolerate crosstalk noises on request signal lines, to ensure fault-free request signals. Redundant hardware are introduced. Theoretical formulations are probed and explored to explain the technique mathematically and the simulation is conducted to demonstrate the effect of the proposed fault tolerance technique at the end.

3.1 Possible Effects of Crosstalk Noise on the Datawaves

In clockless wave pipeline circuit, crosstalk noises can occur on low or high request signals, and they could effect the regular control of request signal to N or P switches. There are four possible combinations between types of switch and noise altogether. In this section, each possible combination and their effects is analyzed in-depth, thus the solution proposed later to solve this request signal introduced fault problem can be easily comprehended.

3.1.1 High Request Signal with Crosstalk Noise to Pass N Switch

Figure3.1 shows the instance of crosstalk noise resided high request signal and its associated datawave propagating a N switch. Part A depicts the fault-free situation before the datawave and request signal propagating through the N switch. As mentioned earlier, datawave associated with high request signal has to be latched at N switch, hence at part B in Figure3.1, part of data bits are latched as normal until the crosstalk noise reach the N switch. The crosstalk noise has the complementary electronic signal value to the regular request signal and it mislead the switch, therefore, the switch switches from opaque state to transparent state immediately. Then the latched first half bits of the datawave are permitted to start to propagate in the next stage of circuit, which is improper since the first half bits start to propagate a lot earlier than designed. At this time, these bits could invalidate the previous datawave because the present bits propagation

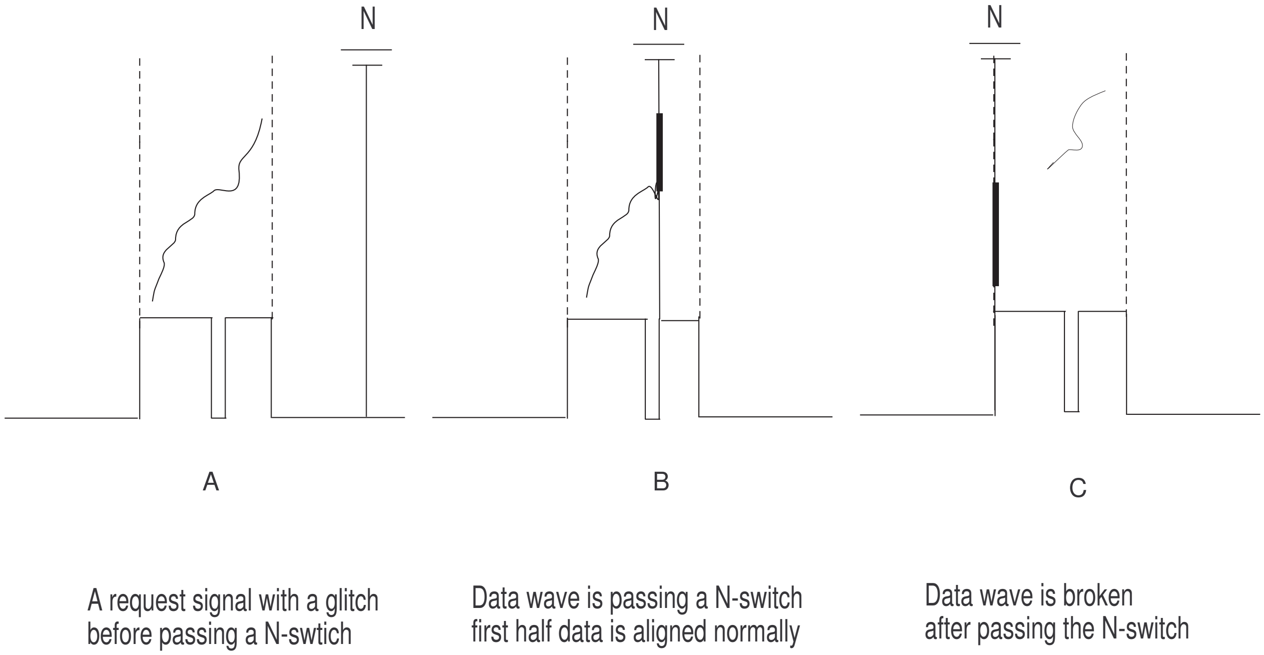

N
N
N
A
B
C
A request signal with a glitch
before passing a N-swtich
Data wave is passing a N-switch
first half data is aligned normally
Data wave is broken
after passing the N-switch

Figure 3.1: Crosstalk noise on a high request signal with the N switch

conflicted with the design specification.

The width of crosstalk noises are very short compared with the length of request signal level, so the control signal switches back to normal request signal swiftly. The switch becomes opaque again and the remaining bits of the current datawave are latched by the opaque switch. At the end of this process, half of the data bits are propagating through the next circuit stage and half of them are latched at the current switch. There is a big gap between them and this data broken could causes fault because there may exist bits dependent in the process of bits propagation. Part C in Figure 3.1 portray this situation.

Broken datawave with big gap eventually causes Intra Wave Fault [13] (the CWP delay fault caused by improper cover of the datawave by the associated request signal level). Figure

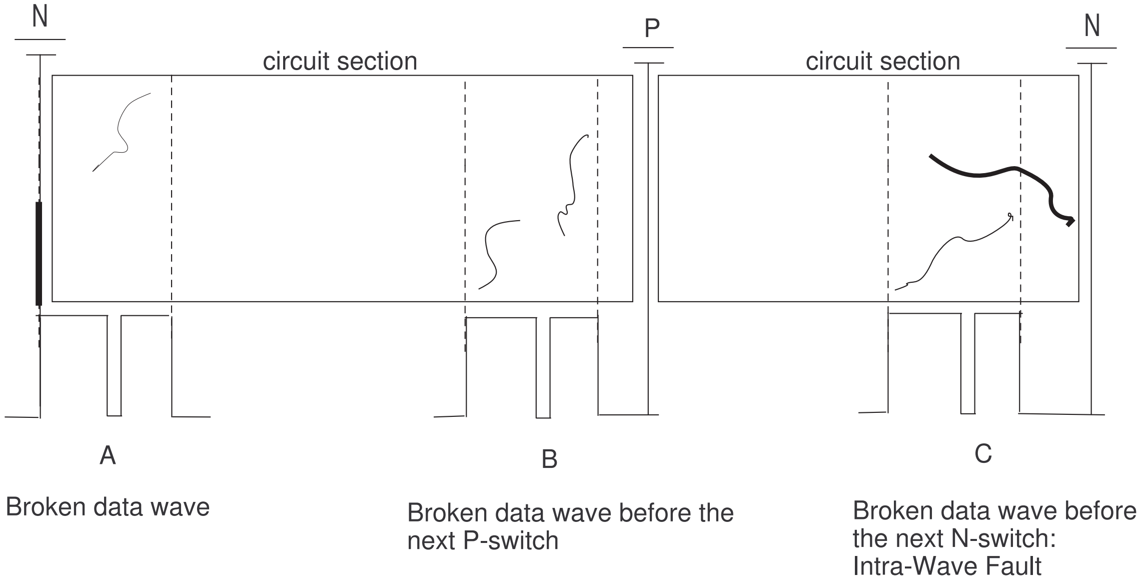

N
P
N
circuit section
circuit section
A
B
C
Broken data wave
Broken data wave before the next P-switch
Broken data wave before the next N-switch: Intra-Wave Fault

Figure 3.2: Broken data on the N switch causing intra-wave fault

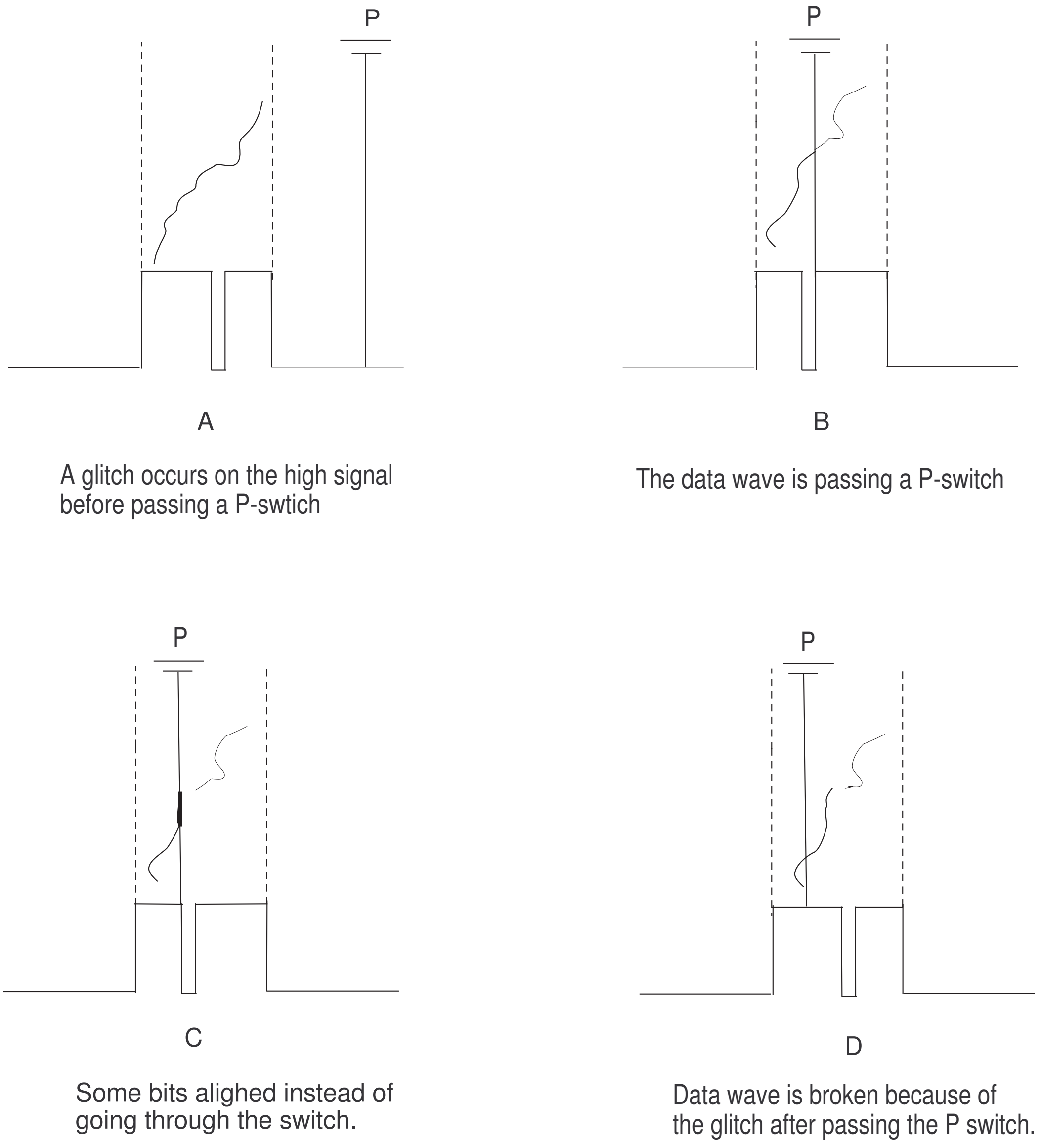
3.2 traces the broken datawave propagation from the break point to the fault point. Site A is position where the broken datawave was generated as discussed earlier. Site B displays the situation where the broken datawave reaches the next switch: a transparent P switch. Site C portrays the intra wave fault situation, where the first half datawave propagates too far and goes beyond its limit because of the unexpected earlier startup. Thus it can not be covered by its associated request signal level properly. The more specific explanation is that the first half data wave starts to propagate earlier than designed and its propagation speed remains unchanged despite the crosstalk noise and data broken, thus it reaches the next N switch earlier than normal after propagating through two circuit sections, while in the normal case, all data bits must arrive after the arrival of the frontier of the high request signal.

3.1.2 High Request Signal with Crosstalk Noise to Pass P Switch

The second combination is the crosstalk noised resided high request signal and P switch. Figure3.3 traces the propagation process. The crosstalk noise occurs at site A, before the datawave propagating through the P switch. According to the clockless wave pipelined circuit design rules, the P switch is transparent to the datawaves associated with high request signals. Thus, at site B, the datawave is propagating through the P switch without being latched. Site C depicts that the P switch switches to opaque to data wave and recovers rapidly since the crosstalk noise take the control of the P switch instantaneous. The temporary opaque section results a very small section of momentary data latching as shown in the site C, and in site D, this small section of latching datawave causes the tiny datawave broken.

It is important to notice that this tiny broken data does not bring any faults. The difference between this and the situation in the first combination is that this broken is caused by temporary latching other than temporary transparent, and temporary latching does not produce datawave's earlier startup as been discussed in the previous part. Since the width of the crosstalk noise is really small, the gap between the two parts of the datawave is also too small to cause any faults, even if bit dependence applied.

Note also if the crosstalk noises rate becomes excessive high, multiple single noises may unite into one big glitch. In that case, the temporary latching time is enlarged , and consequently the gap between the two parts of the broken data is broadened, then intra-fault could be possibly occurs. Fortunately, in reality crosstalk noise density is hardly to be so high. So, in this chapter, this combination is considered to be faulty free.



A glitch occurs on the high signal before passing a P-swtich

The data wave is passing a P-switch

Some bits alighed instead of going through the switch.

Data wave is broken because of the glitch after passing the P switch.

Figure 3.3: Crosstalk noise on a high request signal with P switch

3.1.3 Low Request Signal with Crosstalk Noise to Pass P Switch

This combination can cause fault. It is very similar to the first combination. Datawave associated with low request signal is latched by the P switch, and the crosstalk noise can broken the datawave with a big gap because of temporary transparent. At the next successive P switch, intra-wave fault and bits dependent fault then occur. This combination is described in Figure

3.4 and 3.5.

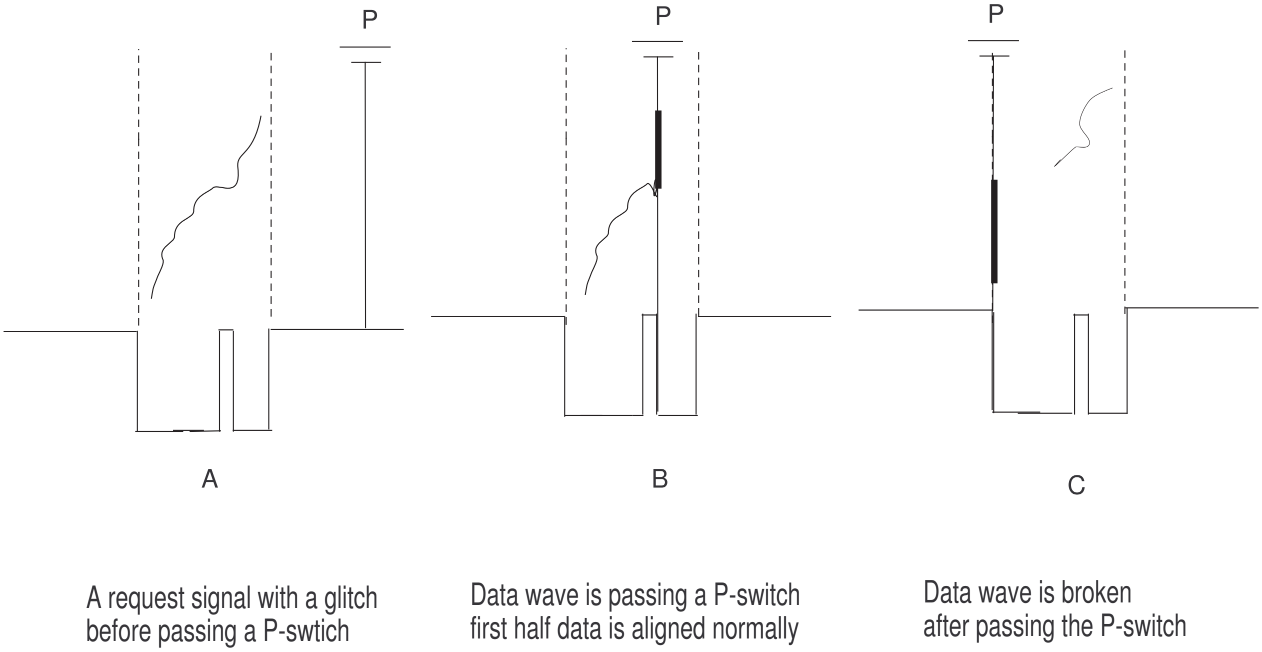

P
P
P
A
B
C
A request signal with a glitch
before passing a P-swtich
Data wave is passing a P-switch
first half data is aligned normally
Data wave is broken
after passing the P-switch

Figure 3.4: Crosstalk noise on a low request signal with P switch

3.1.4 Low Request Signal with Crosstalk Noise to Pass N Switch

The case is similar with the second combination. Crosstalk noise on low request signal before a N switch does not cause fault since temporary latching does not produce datawave's earlier startup and the tiny datawave broken does not cause bits dependent fault unless extremely excessive crosstalk noise density. The figure depicting this situation are omitted since it is same as Figure3.3 except using N switch and low request signal instead of P switch and high request signal.

Up till now, the complete four combinations of request signals and the switches are investigated carefully, two of them, which are high request signal combined with N switch, low request

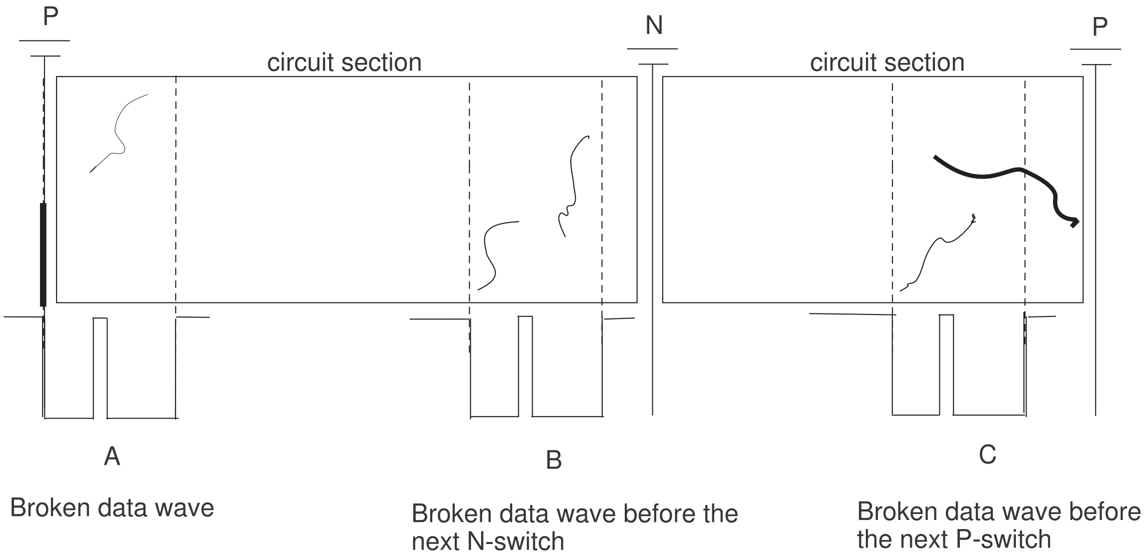

P
N
P
circuit section
circuit section
A
B
C
Broken data wave
Broken data wave before the next N-switch
Broken data wave before the next P-switch

Figure 3.5: Broken data on P switch causing intra-wave fault

Table 3.1: Fault cases

-	N-switch	P-switch
Crosstalk noise at high request signal	faulty	not faulty
Crosstalk noise at low request signal	not faulty	faulty

signal combined with P switch are suspicious to results faults in the present of crosstalk noise. The other two combinations, which are high request signal combined with P switch, low request signal combined with N switch are reluctant to generate any faults even crosstalk noises stand on the request signal. This is summarized in Table 3.1.

3.2 Fault Tolerant Design Approaches

According to the discussion of the interactive behavior of the data waves and switches under the present of faulty crosstalk noise in the previous section, redundant request line are introduced into the two-phase asynchronous wave pipeline architecture to mask the occurred crosstalk noise. The proposed method to ensure a fault-free request signal is Fault-Masking. The fault-

Table 3.2: Truth table

A (regular signal)	B (crosstalk noise signal)	A OR B	A AND B
high	low	high	X
low	high	X	low

masking method is to employ redundant request signal lines and an AND and OR gate at each P and N switch, respectively. An implementation of the fault-masking architecture is shown in Figure 3.6.

An AND gate at a P switch will control the incoming datawave such that any low-glitch hit on the request signal is masked to the normal (i.e., high) value unless every request signal is hit by a low-glitch exactly at the same time (as illustrated in the upper row in Table 3.2; and vice versa in the case of an OR gate at an N switch (as illustrated in the lower row in Table 3.2. Note that the probability for an original request signal and all redundant ones to be hit by the glitches exactly at the same time is practically negligible. Hence, the proposed simple yet powerful fault-tolerant request signal method by using the fault-masking technique is efficient and effective.

Note not all crosstalk noises on the request signal can be hidden by this. The only condition where this fault mask technique fails is that both of the request signal A and B has crosstalk noises at the same time. In that case, another request line with accurate signal is needed to achieve a higher fault tolerance rate and more reliable system. By this means, the number of request line is expended to N so that the modified pipeline's reliability can be described and calculated theoretically in the next section.

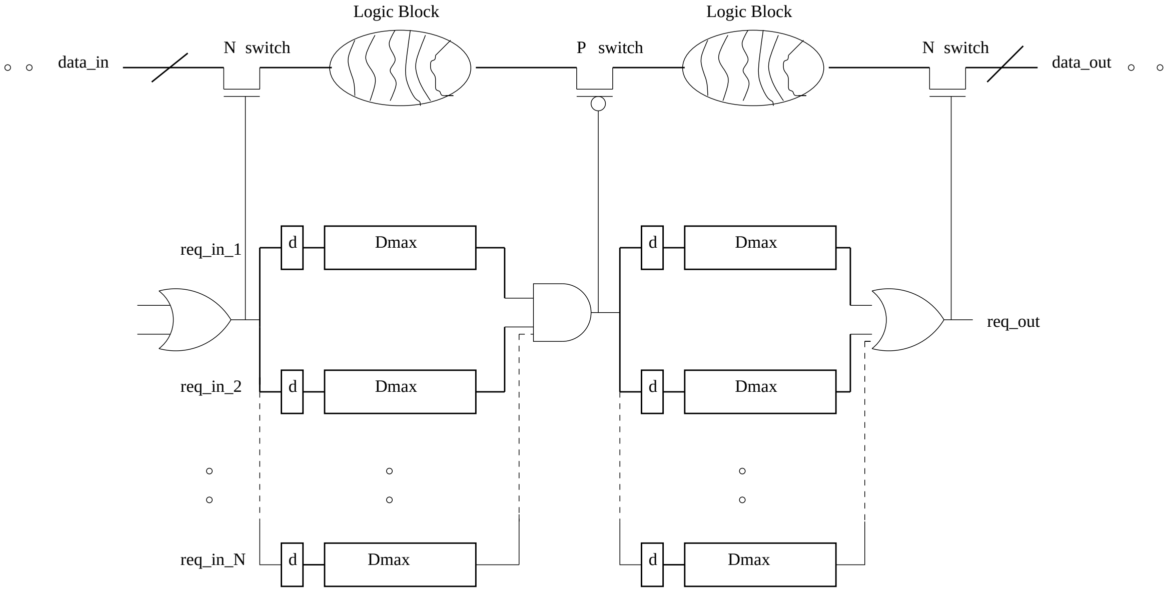


Figure 3.6: Modified two-phase asynchronous wave pipeline with redundant request signal lines

3.3 Reliability Calculation

Assume λ is the rate that the crosstalk noise occurs on any request signal line from the signal entering the circuit to the signal arrives at the primary output of the circuit and assume the primary request signal line and redundant request signal lines share the same crosstalk noise rate.

3.3.1 Effectiveness of Multiple Request Signal Lines on Reliability

After the application of the redundant request signals, the crosstalk noises are masked away except crosstalk noises occurs at every request signal at the same time. so, the final fault rate on the redundant request signal is:

$$\lambda_{total} = (\lambda \times \frac{l}{L})^N \tag{3.1}$$

where l is the minimum valid crosstalk noise width, and L is the length of one request signal section.

Thus, the total reliability of the request signal module is:

$$R(t) = e^{-(\lambda \times \frac{l}{L})^N \times t} \tag{3.2}$$

3.3.2 MTTG, λ, and L

We propose a reliability model to thoroughly verify the effectiveness of the proposed fault-masking method. The model employs the Mean Time to Glitch (MTTG) to take into account and demonstrate the effect of the fault-masking on request signals at different levels of redundancy. Assume λ is the rate that a glitch hits on a request signal line during the period of time from the instant of time when the signal was submitted into the primary input of the circuit to the instant of time when the signal arrives at the primary output of the circuit. Also, assume that the primary request signal line and redundant request signal line(s) have the same glitch rate without loss of generality. Having the redundant request signal line(s) employed, the glitch can be masked off except when the glitches hit on all the request signals respectively at the same time.

Thus, the overall fault rate of the primary and redundant request signal(s) is $\lambda_{effective} = (\lambda \times \frac{l}{L})$, where l is the mean time width of a glitch's effect, and L is the length of the request signal; and the relative width ($\frac{l}{L}$) of a glitch hit on the request signal pulse L, is practically considered. Therefore, the effective reliability of N request signals inclusive of the primary and redundant one(s) is $R(t) = e^{-(\lambda \times \frac{L}{L})^N \times t}$. MTTG is the mean time for a high or low signal to propagate normally through the circuit before the first glitch hits. Hence, the MTTG can be modeled as follows:

$$
\begin{aligned}
MTTG &= \int_0^{\infty} R(t)dt \\
&= \int_0^{kL} R(t)dt \\
&= \int_0^{kL} e^{-\lambda t}dt \\
&= -\frac{1}{\lambda}e^{-\lambda kL} - (-\frac{1}{\lambda}e^{-\lambda(0)}) \\
&= \frac{1}{\lambda}(1 - e^{-kL\lambda}) \qquad (3.3)
\end{aligned}
$$

where k is the *depth of wave-pipeline*, i.e., the number of request signal cycles needed for a data signal to propagate through a partial circuit before being aligned at a switch; thus kL is the total length of a request signal path within the partial circuit. Note that according to Equation (3.3) the glitch rate can impact MTTG significantly, and L can be derived as $L = \frac{\ln(\lambda \times MTTG - 1)}{k\lambda}$. The expression for L can be used as a theoretical bridge between MTTG-induced request signal faults and intra/inter-wave faults; and is used as a primary driver to optimize the intra/inter-wave fault rate with respect to a request signal fault.

Equation 3.3 shown that crosstalk noise rate can influence MTTG in a great extents, the following Figure3.7 is plotted based on equation 3.3 with the assumption of known parameter K, L, and it shows the decreasing trends of MTTG when the crosstalk noise rate increases.

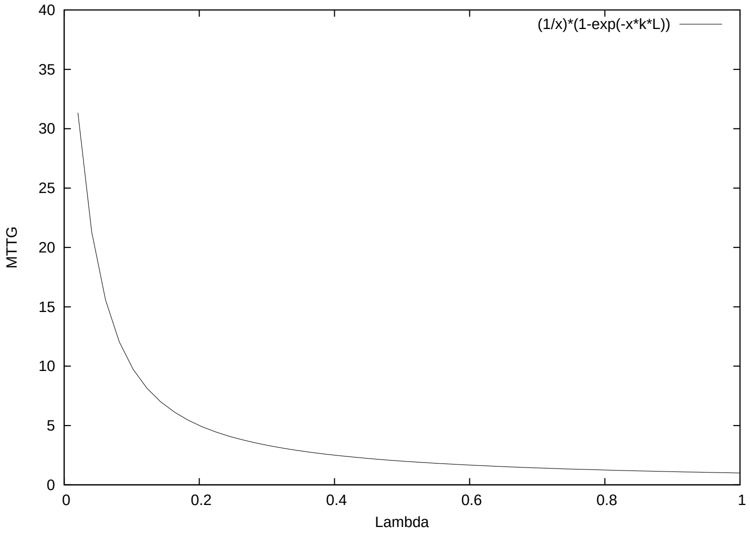


Figure 3.7: MTTG is decreasing when the crosstalk noise rate increases

3.3.3 Reliability of the Redundant Request Signal Part at MTTG

The reliability of a single request signal at the time t is:

$$R(t) = e^{-\lambda t}[31] \tag{3.4}$$

Having MTTG theoretically identified, the reliability of a single request signal at MTTG [31] can be expressed as follows:

$$\begin{aligned} R(MTTG) &= R(\frac{1}{\lambda}(1 - e^{-kL\lambda})) \\ &= e^{-\lambda(\frac{1}{\lambda}(1-e^{-kL\lambda}))} \\ &= e^{(e^{-\lambda kL}-1)} \end{aligned} \tag{3.5}$$

The reliability of the request signal part of the whole circuit with N request signal lines is:

$$R_{total} = \sum_{i=1}^{N-1} \times N \times R^{N-i} \times (1-R)^i \times (1-P) \tag{3.6}$$

where $P = \lambda^N$.

The reliability of the whole request signal part at time point MTTG can be obtained by combining equation 3.5 and 3.6 and displays as follows.

$$\begin{aligned} R(MTTG) &= \sum_{i=1}^{N-1} \times N \times R^{N-i} \times (1-R)^i \times (1-P) \\ &= \sum_{i=1}^{N-1} \times N \times (e^{(e^{-\lambda kL}-1)})^{N-i} \times (1-(e^{(e^{-\lambda kL}-1)}))^i \times (1-\lambda^N) \end{aligned} \tag{3.7}$$

In Figure(3.8) $R(MTTG)_{overall}$ is plotted for 1, 2, 3, and 4 request signal lines, respectively, and each with $\lambda_{effective} = 0.06$. It shows that the overall reliability is enhanced exponentially as the number of the redundant request signal lines increase. In practice, inclusion of an AND or OR gate, and widening that gate to accommodate more request signal lines, would mean a slight increase in the variance of the timing of the request signal, as perceived at a switch. So, this increase in variance would eventually limit how worthwhile the Fault Masking is.

3.4 Implementation Result

The circuit implementation is conducted using Cadence to verify the proposed fault tolerant technique. Two request signal lines with crosstalk noises are filtered by AND gate and OR gate respectively. They are then extracted and simulated using Spectre. The designed circuit is shown in Figure3.9.

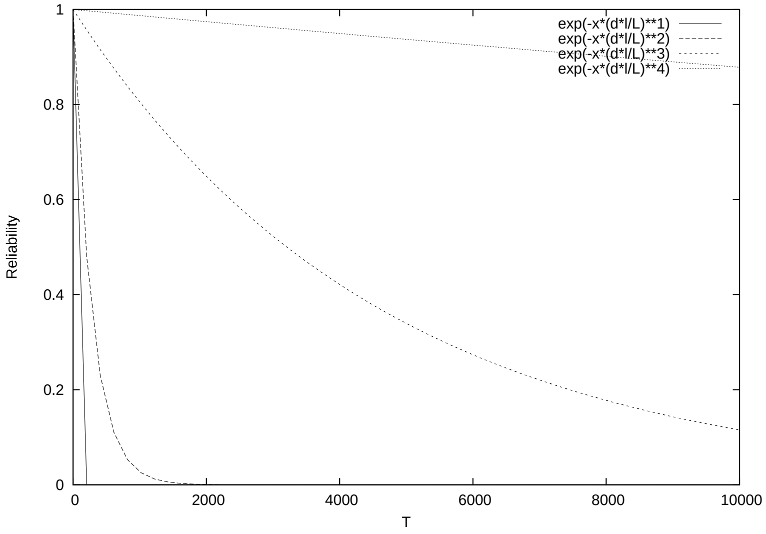


Figure 3.8: System reliability with 1,2,3,4 request signal lines

In Figure3.10, $input1$ and $input2$ are the signals of the two request signal lines and crosstalk noises reside on them. The out signal shows that noises resided on the low request signal are successfully masked away by the AND gate. Although the noise density on the high request signal level double, as discussed in section 3.1, noises on the high request signal level for a P switch is harmless until the crosstalk noise density goes extremely high and in reality, that is hardly occur. Meanwhile the crosstalk noises accumulation can not happen since all the noises resided on high request signal level, including the original noises and the AND gate introduced noises, will be discard by the OR gate in the next successive N switch as shown in Figure3.11. So there is no other faulty risk introduced by this technique.

Figure3.11 demonstrate the OR gate masks the crosstalk noise resided on high request signal. Equivalently, $input1$ and $input2$ are the two request signals with noises resided on and out is the signal filtered by the OR gate, where the noises on the high request signal are hidden away by the OR gate. Noise density on the low request signal also doubles after OR gate, but normally they will be casted away by AND gate in the next successive P switch.

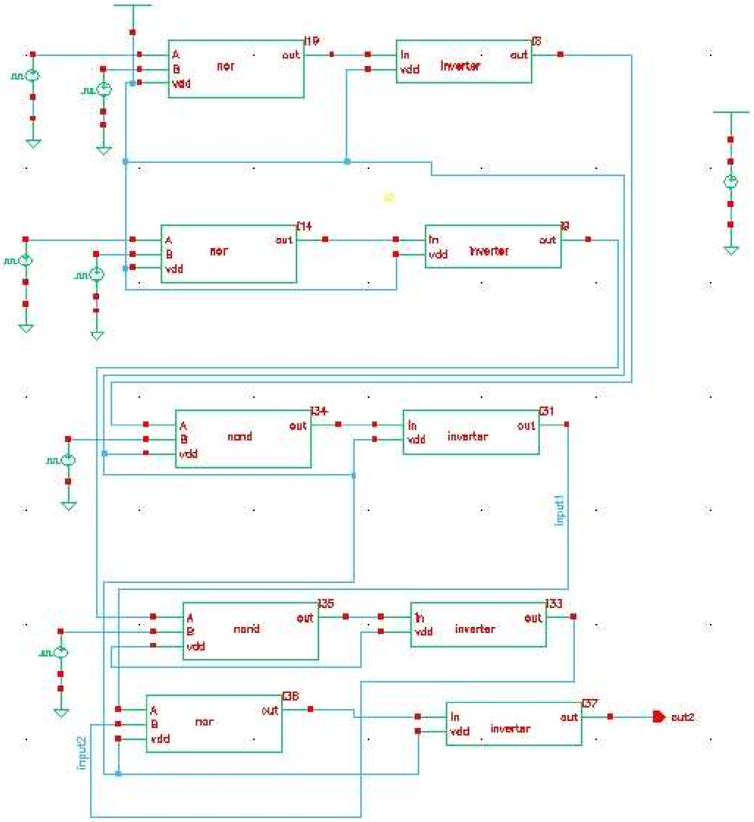


Figure 3.9: Circuit implementation of the fault tolerant technique

3.5 Conclusions

This chapter has presented a totally new approach to fault tolerate crosstalk noises on request signal in the clockless wave pipelined circuit by redundant hardware model, i.e., redundant request signals.

It was illustrated that low level crosstalk noises on the high request signal line can cause data broken and consequently intra-wave fault on N switches, whereas harmless to P switches. On the contrary, high signal crosstalk noises in the low request signal line can result data broken

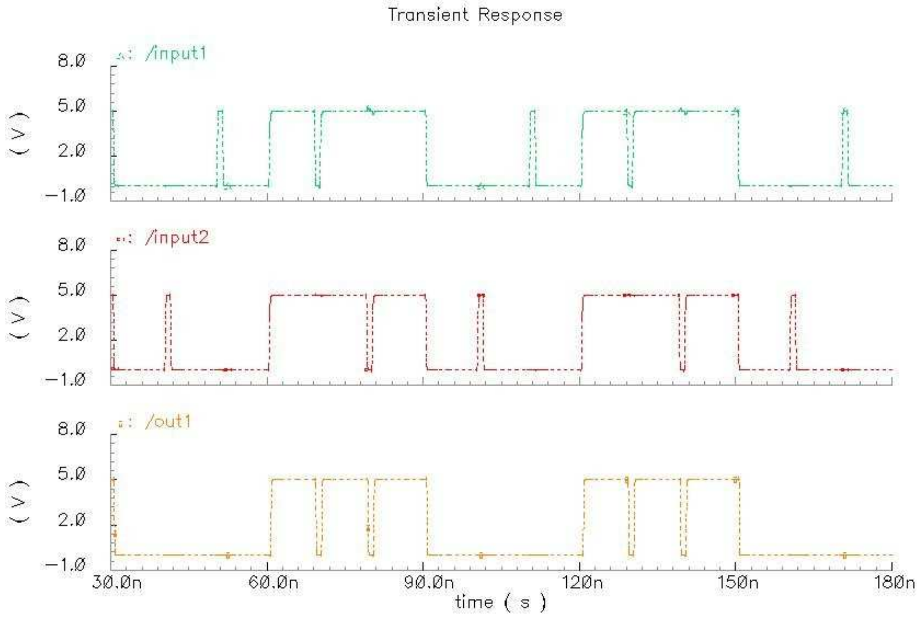


Figure 3.10: Simulation of crosstalk noise on a low request signal masked by AND gate on P switch

and intra-wave fault, but safe to N switches. Based on this cognition, AND gate are used to hide high level crosstalk noises resided on low request signal to the P switch and OR logic gates are used to mask low level crosstalk noises resided on high request signal level to the N switch.

This simple and efficient approach can hide all the crosstalk noises on request signal lines except each redundant request signal line is impaired by crosstalk noise at the same time. Computations are achieved on the effectiveness of redundant request signal lines. It was demonstrated that the reliability of the system increased exponentially as more redundant request signal lines included. Furthermore, consideration of the reliability at MTTG (Mean Time To Crosstalk noise), λ (crosstalk noise rate) and L (request signal level length) were also been provided. Finally, the simulation verified that this approach is a novel yet practical fault tolerance technique to the clockless wave pipelined circuit.

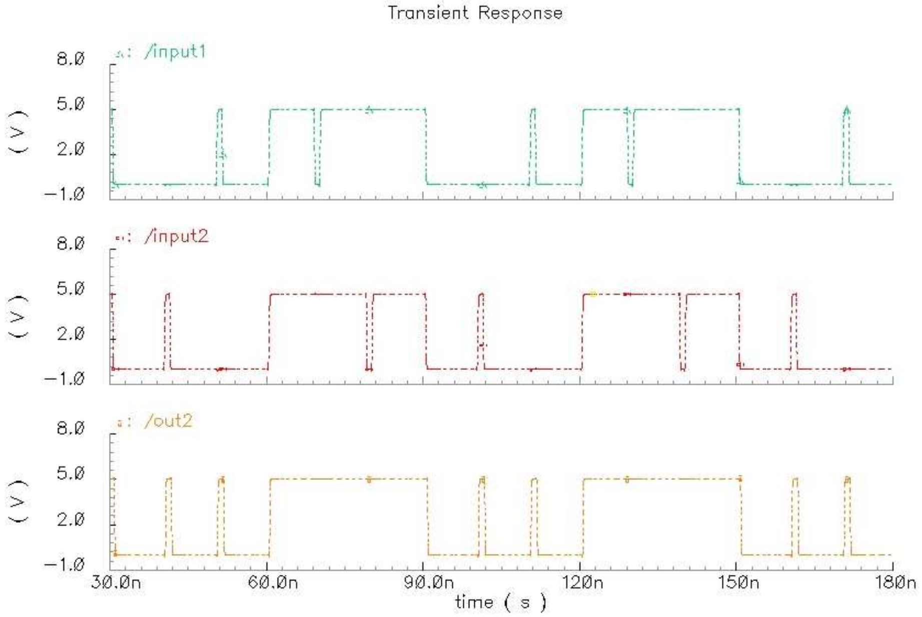


Figure 3.11: Simulation of crosstalk noise on a high request signal masked by OR gate on N switch

CHAPTER 4

FAULT TOLERANT DESIGN FOR DATAWAVE FAULTS

The basic computational components of clockless wave pipeline are datawaves and request signals. In this chapter we present the fault tolerant design for datawaves, namely the bipolar switches in association with affiliate request signals. The proposed technique will reduce or possibly completely tolerate the modeled intra-wave and inter-wave faults on datawaves.

The main objective of the proposed method is to control the shape of the datawaves to maintain their width (i.e., $d_{max} - d_{min}$) within L (i.e., the length of the request signal pulse in association).

4.1 The Basic Framework for Reliability Optimization

The basic framework for reliability optimization employs the proposed novel idea of Bipolar Switches in association with Affiliate Request Signals.

The proposed bipolar switches, as shown in Figure 4.1, consists of two switches connected in series where each of the switches has an opposite polarity. There are two types of bipolar switches, PN-switch (i.e., series connection of P- and N-type switches) and NP-switch (i.e., series connection of N- and P-type switches). The bipolar switches can be deployed such that each type of bipolar switch is alternately placed through the circuit. The alternate placement of bipolar switches enables the proposed Double Alignment Method, i.e., datawaves are given another chance for alignment. Hence, there are two possible arrangements to consider in

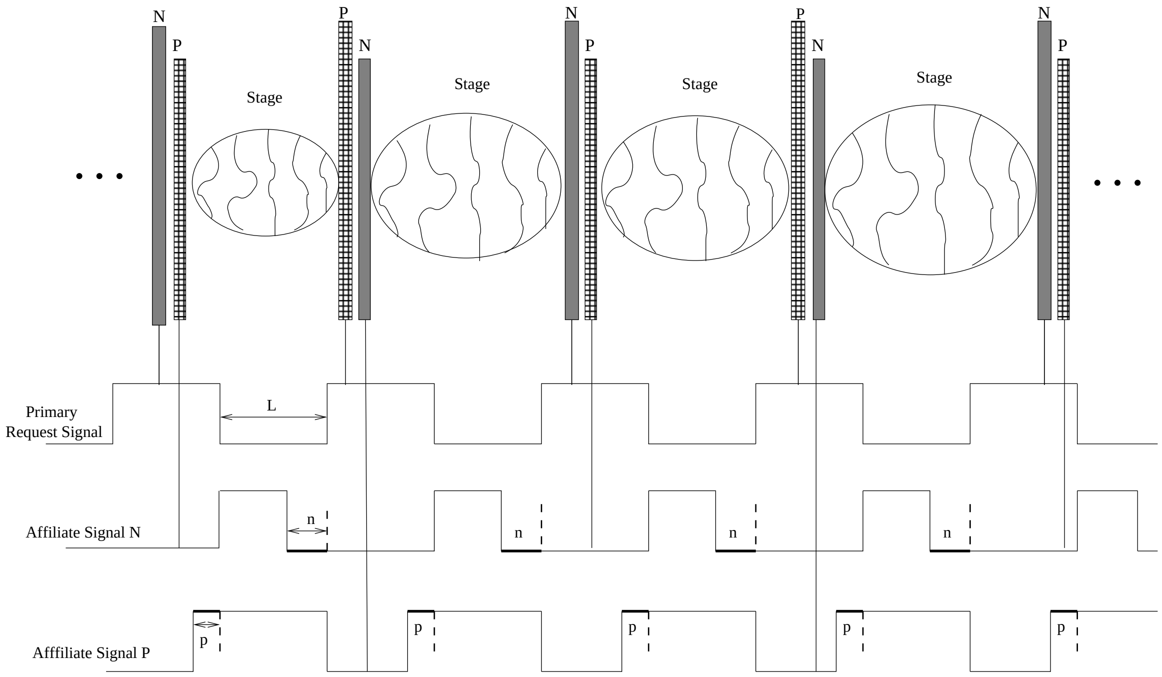


Figure 4.1: Architecture of the reliability enhanced 2-phase clockless wave pipeline

demonstrating the double alignment methods, such as PN-NP-PN and NP-PN-NP. As the two arrangements are symmetric, we will demonstrate only the first arrangement of PN-NP-PN as shown in Figures 4.2, 4.3 and 4.4.

In Figure 4.2, we show a datawave propagating through the stage in between the first PN and NP; the datawave, which was initially supposed to be completely aligned (i.e., the first alignment) at the P-switch in the first PN-switch, is then getting out-of-alignment during the propagation due to variation in path delays; the datawave is associated with its primary request signal with low-level; the affiliate request signal with low-level in length n is provided to control the double alignment. Figure 4.3 demonstrates the double alignment method: at the NP-switch the datawave just passes through the N-switch because the datawave is associated with high-level primary request signal and the N-switch is transparent to high-level request signal; passing through the N-switch, the datawave then gets aligned at the P-switch (i.e., the second-chance alignment) in association with the low-level affiliate request signal for n period of time because P-switch is opaque to high-level request signal; the second-chance alignment period n can be determined by the expected extent of the misalignment (note that in the symmetric case of NP-PN-NP, the high-level affiliate request signal extends for p period of time). In Figure 4.4, it is shown that the datawave has realigned after passing through the double alignment; Thus, the proposed double alignment method will definitely reduce the chance for an intra-wave fault to occur, thereby reducing the chance for an inter-wave fault as well.

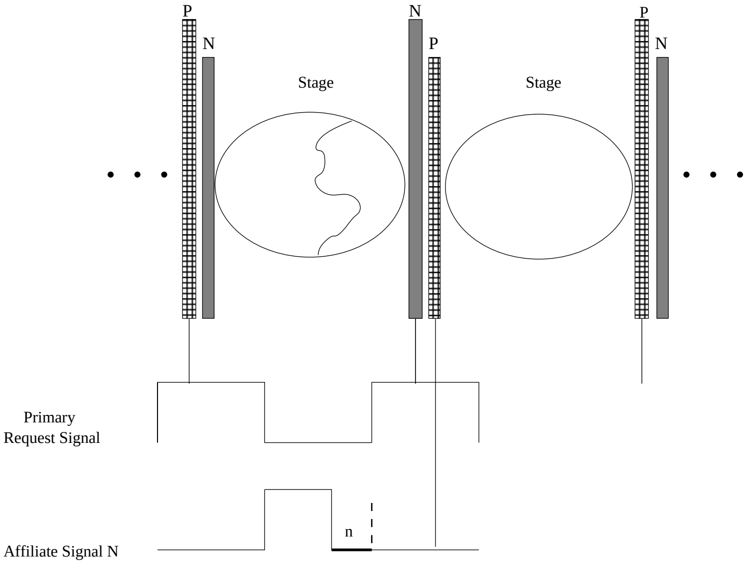


Figure 4.2: Before passing the middle N-switch and P-affiliated switch

4.2 Theoretical Validation of the Proposed Fault Tolerance Method

In order to demonstrate the validity of the proposed fault tolerance method for the datawave faults, we use a modeling and assurance method as follows.

Assume a random variable x with a Gaussian distribution to represent the path delay of a

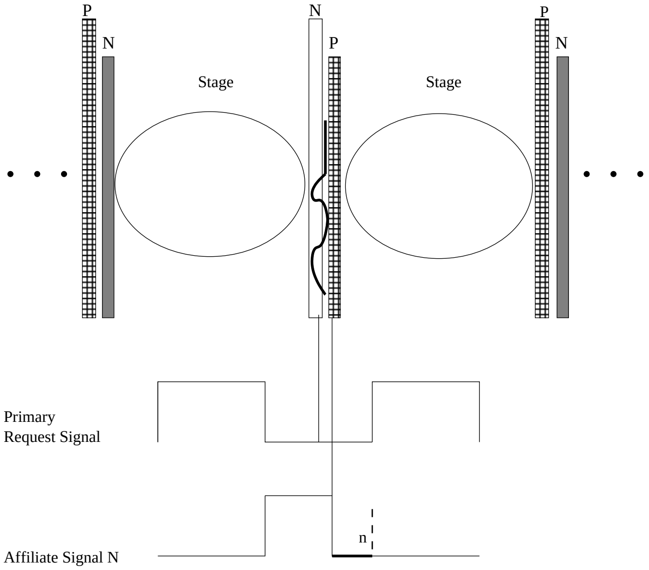


Figure 4.3: Passing the middle N-switch and P-affiliated switch

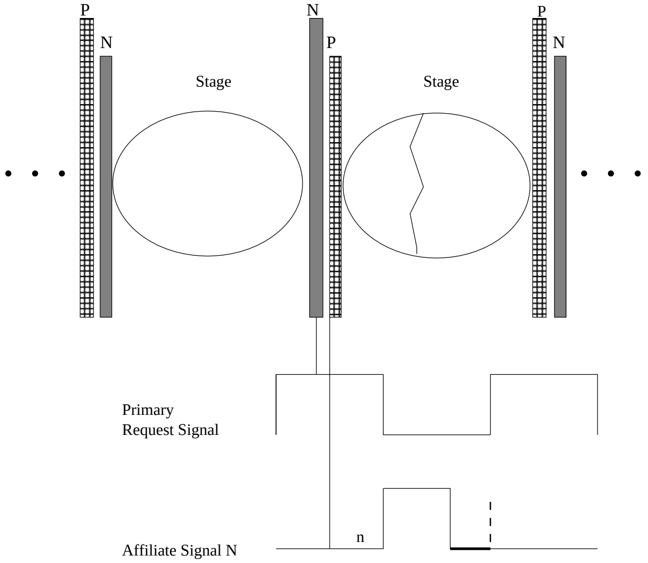


Figure 4.4: After passing the middle N-switch and P-affiliated switch

bit in a datawave without the bipolar switches; and y with the bipolar switches. The p.d.f. of x can be expressed as follows: $f_x(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})}$. In order to manipulate the extra-delay induced by the double-alignment, we introduce a parameter γ such that the value of the effective extra-delay is $n \times \gamma$ (or symmetrically $p \times \gamma$). y is the random variable of concern since that represents the new delay variable with the bipolar switches employed, and is to be formalized in terms of the path delay variable without the bipolar switches (i.e., x). In the analysis, suppose there is a specific path delay value z of x at which the bipolar switch has the effect of increasing y from x (i.e., z) to $x + n\gamma$ (i.e. $z + n\gamma$) as shown in Figure 4.5. In this context, y and x can be expressed as follows.

$$y = \begin{cases} x & \text{if } x \geq z \\ x + n \times \gamma & \text{if } x < z \end{cases} \tag{4.1}$$

The inverse function of Equation 4.1 is as follows, and illustrated in Figure 4.6.

$$x = \begin{cases} y & \text{if } y \geq z \\ y + n \times \gamma & \text{if } y < z \end{cases} \tag{4.2}$$

An expected resulting p.d.f. of y, $f(y)$, is shown on the bottom in Figure (4.7). As shown in Figure 4.7, $f(y)$ can be derived range by range as follows: in the range $(-\infty, z)$, $f(y)$ is shifted downward resulting in the reduced shaded area beyond L to the left; this is an obvious indication of Reduced Intra-Wave Fault Rate; likewise, in the range of $(z, z + n\gamma)$, $f(y)$ is shifted upward

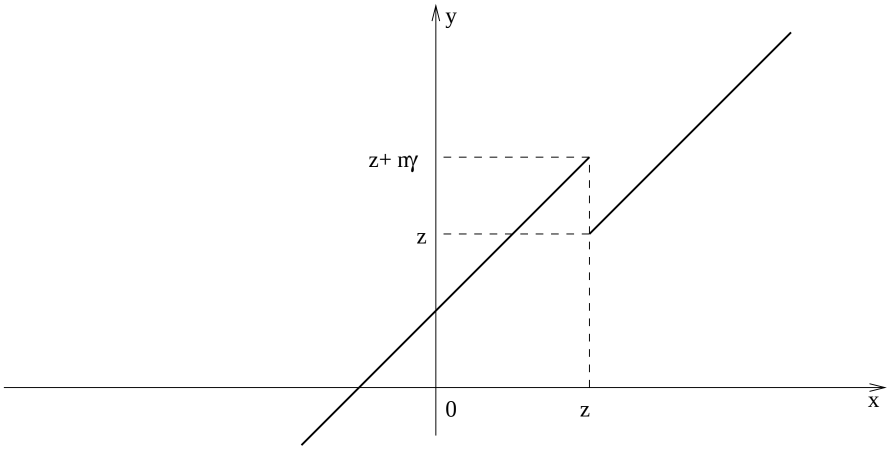


Figure 4.5: Path delay with bipolar switches vs. without

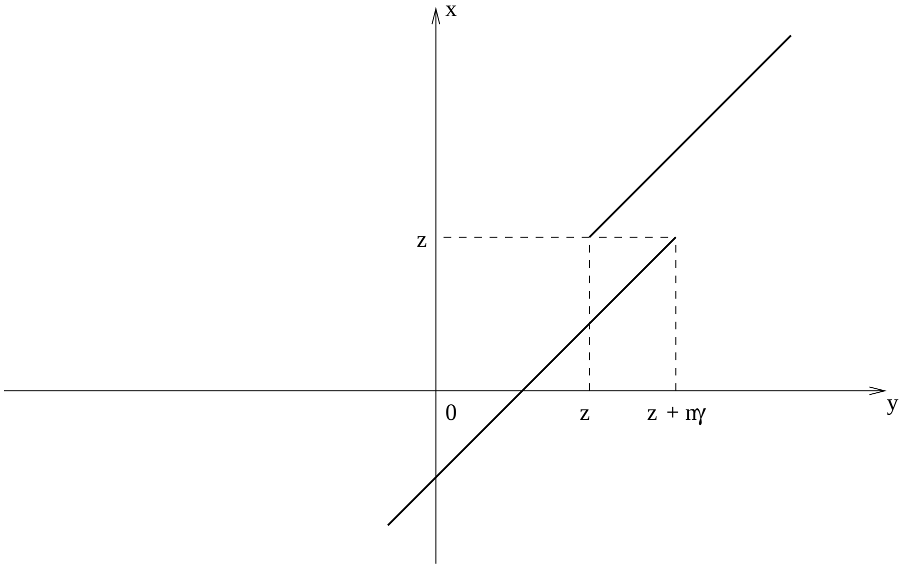


Figure 4.6: Inverse function

resulting in a new p.d.f; and the remaining portion of the p.d.f. remains unchanged.

1. If $y \in [-\infty, z]$,

$$\begin{aligned} f_y(y) &= \frac{d}{d_y}F(y) = \frac{d}{d_y}[\int_{-\infty}^{y} f_y(t)dt] = \frac{d}{d_y}[\int_{-\infty}^{x} f_x(t)dt] = \frac{d}{d_y}[\int_{-\infty}^{y-n\times\gamma} \frac{1}{\sqrt{2\pi}\sigma}e^{(-\frac{(t-\mu)^2}{2\sigma^2})}dt] \\ &= \frac{1}{\sqrt{2\pi}\sigma}e^{(-\frac{(y-n\times\gamma-\mu)^2}{2\sigma^2})} \end{aligned} \quad (4.3)$$

2. If $y \in [z + n \times \gamma, +\infty]$, $x = y$. Thus, $f_y(y) = \frac{1}{\sqrt{2\pi}\sigma}e^{(-\frac{(y-\mu)^2}{2\sigma^2})}$.

3. If $y \in [z, z + n \times \gamma]$,

$$\forall y \in [z, z + n \times \gamma], \quad x = y \quad \& \quad x = y - n \times \gamma$$

$$f_x(y) = \frac{1}{\sqrt{2\pi}\sigma}e^{(-\frac{(y-\mu)^2}{2\sigma^2})}$$

$$f_x(y - n \times \gamma) = \frac{1}{\sqrt{2\pi}\sigma}e^{(-\frac{(y-n\times\gamma-\mu)^2}{2\sigma^2})}$$

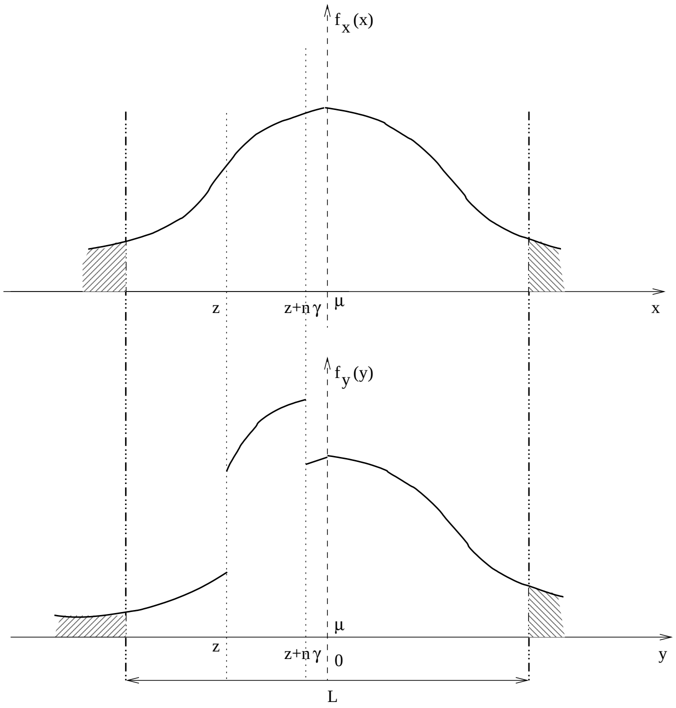


Figure 4.7: Probability density function of the path delay with bipolar switches

and thus

$$\begin{aligned} f_y(y) &= f_x(y) + f_x(y - n \times \gamma) \\ &= \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-\mu)^2}{2\sigma^2})} + \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-n\times\gamma-\mu)^2}{2\sigma^2})} \qquad (4.4) \end{aligned}$$

Putting the $f_y(y)$'s from each range together, the overall $f_y(y)$ is as follows and verifies the correctness of the proposed model.

$$\begin{aligned} f_y(y) \quad &= \int_{-\infty}^{z-n\times\gamma} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-\mu)^2}{2\sigma^2})} dy + \int_{z-n\times\gamma}^{z+n\times\gamma} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-\mu)^2}{2\sigma^2})} dy + \int_{z+n\times\gamma}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-\mu)^2}{2\sigma^2})} dy \\ &= \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-\mu)^2}{2\sigma^2})} dy \\ &= 1 \qquad (4.5) \end{aligned}$$

4.3 The Intra-Wave Fault Rate with the Proposed Architecture

Without loss of generality, we assume functional independence between PN bipolar switches and NP bipolar switches, and then based on the shaded area in the p.d.f. on the bottom of Figure 4.7, the intra-wave fault rate for each type of bipolar switch (i.e., i–th n-type for a PN bipolar switch and j–th p-type for an NP bipolar switch) can be expressed as follows, respectively.

$$P_{i'th-n} = \int_{-\infty}^{d_{min_i}-(L-\alpha_i-\triangle_i)} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-n\times\gamma-\mu)^2}{2\sigma^2})} dy + \int_{d_{max_i}+\alpha_i}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-\mu)^2}{2\sigma^2})} dy \qquad (4.6)$$

$$P_{j'th-p} = \int_{-\infty}^{d_{min_j}-(L-\alpha_i-\triangle_j)} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-p\times\gamma-\mu)^2}{2\sigma^2})} dy + \int_{d_{max_j}+\alpha_j}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(y-\mu)^2}{2\sigma^2})} dy \qquad (4.7)$$

Thus, assuming there are n number of n–type bipolar switches and p number of p–type bipolar switches, the overall intra-wave fault rate ($P_{overall}$) can be expressed as:

$$P_{overall} = 1 - \prod_{i=1}^{n}(1-P_i) \times \prod_{j=1}^{p}(1-P_j) \qquad (4.8)$$

Therefore, the overall yield, Y, is

$$\begin{aligned} Y &= 1 - P_{overall} \\ &= \prod_{i=1}^{n}(1-P_i) \times \prod_{j=1}^{p}(1-P_j) \qquad (4.9) \end{aligned}$$

4.4 Simulations

The simulation and validation of the proposed fault tolerant model and design is conducted against the delay information obtained from ISCAS benchmark circuits $C432$, $C499$ and $C880$ [83]. 83926, 9440 and 8642 paths with delay information are extracted from these three benchmark circuits, respectively. The statistical analysis of the path delays are implemented for with or without the bipolar switches design respectively. The simulation results demonstrate that the proposed fault tolerance technique using the bipolar switches has a great impact on the intra-wave fault rate. Note that the simulations are conducted only with respect to intra-wave fault because inter-wave fault functionally depends on intra-wave fault.

The circuit paths are extracted out from the benchmark circuit netlist specification. Then, the delay information of the wires and gates, which are stored in the file in a stand delay format (sdf), are applied to those paths. Because the delay information specified in this benchmark circuit are all static worst case delay, the upper delay bound of each path can be directly obtained by adding up the gate delay through the longest port for every gate and all the wire delays along the path. The lower delay bound of each path is obtained by adding up the gate delay through the shortest port for every gate and all the wire delays along the path.

Since the benchmark circuits are not in its pipeline design, at this point, the maximum and minimum path delays are the bound for the whole path from the Primary Input to the Primary Output. The benchmark circuits are virtually transformed into clockless wave pipeline circuits and clockless wave pipeline circuits with bipolar latches for the demonstration purpose in this work. The whole paths from PI to PO are ideally equally divided into multiple fractions, and each fraction fits in to one stage. Note that in the real case, the functional dependence has to be considered when designing a pipelined circuit. Thus, the whole path usually is not divided into equal fractions. However, the way how the pipelined circuit is designed does not effect the simulation experiment results, since generally, the bipolar switch-based fault tolerant design works on all clockless wave pipelined circuit.

In the original clockless wave pipeline design, there is only one latch (N or P) in between stages. The delay of the latch is not counted in this work since it is negligible compared with the whole path delay. According to the clockless wave pipeline architecture, the partial paths are calculated as part of the path on any two attached stages. All the partial path delays are considered as sample, and are statistically analyzed. Their results are listed in the tables to be shown later.

In the bipolar switch-based fault tolerant design, the only difference is that there are two latches in between stages instead of one latch. The partial paths are still part of the path on any two attached stages. Since the added latches extended the faster paths, all the partial paths which have longer path delay remains unchanged, while all the partial paths which have shorter path delay are extended into a certain point. The value of this certain point depends on how the parameters (γ and np) are set. All the new generated partial paths are considered as a sample, and are statistically analyzed. Their results are listed in the tables too.

Table 4.1 and Figure 4.8 show the statistical analysis results of the benchmark circuit, $C432$ clockless wave pipeline, without bipolar design; Table 4.2 and Figure 4.9 show the statistical analysis results of the same circuit with bipolar design respectively. They clearly manifest the effectiveness of the bipolar design by increasing the values of D_{min}. For example, the value of D_{min} increased from 41.5053 to 109.2241 when the circuit has 3 stages, while D_{max}

Table 4.1: Statistic of ISCAS C432 path delays in the original design (ps)

Number of Stages	Mean	Variance	S.D.	D_{max}	D_{min}	Δ
3	275.4943	1530.4559	39.121	323.6667	41.5053	282.1613
4	206.6207	860.8814	29.3408	242.75	31.129	211.621
5	165.2966	550.9641	23.4726	194.2	24.9032	169.2968
6	137.7471	382.614	19.5605	161.8333	20.7527	141.0807
7	118.069	281.1041	16.7662	138.7143	17.788	120.9263
8	103.3104	215.2204	14.6704	121.375	15.5645	105.8105
9	91.8314	170.0507	13.0403	107.8889	13.8351	94.0538
10	82.6483	137.741	11.7363	97.1	12.4516	84.6484
11	75.1348	113.8356	10.6694	88.2727	11.3196	76.9531
12	68.8736	95.6535	9.7803	80.9167	10.3763	70.5403
13	63.5756	81.5036	9.0279	74.6923	9.5782	65.1142
14	59.0345	70.276	8.3831	69.3571	8.894	60.4631
15	55.0989	61.2182	7.8242	64.7333	8.3011	56.4323
16	51.6552	53.8051	7.3352	60.6875	7.7822	52.9052

Table 4.2: Statistic of ISCAS C432 path delays in the fault tolerant design with $np = 0.6, \gamma = 0.6$ (ps)

Number of Stages	Mean	Variance	S.D.	D_{max}	D_{min}	Δ
3	275.542	1512.6147	38.8923	323.6667	109.2241	214.4426
4	206.6379	855.7118	29.2526	242.75	69.2208	173.5292
5	165.305	548.8563	23.4277	194.2	49.2819	144.9181
6	137.7523	381.5238	19.5326	161.8333	37.6823	124.151
7	118.0723	280.4876	16.7478	138.7143	30.2261	108.4882
8	103.3126	214.8549	14.6579	121.375	25.0874	96.2876
9	91.833	169.8152	13.0313	107.8889	21.3594	86.5295
10	82.6495	137.5813	11.7295	97.1	18.5463	78.5537
11	75.1358	113.7177	10.6638	88.2727	16.3566	71.9162
12	68.8744	95.5639	9.7757	80.9167	14.6088	66.3079
13	63.5763	81.434	9.0241	74.6923	13.1845	61.5078
14	59.0351	70.2212	8.3798	69.3571	12.0035	57.3536
15	55.0993	61.1741	7.8214	64.7333	11.0098	53.7235
16	51.6556	53.7691	7.3327	60.6875	10.163	50.5245

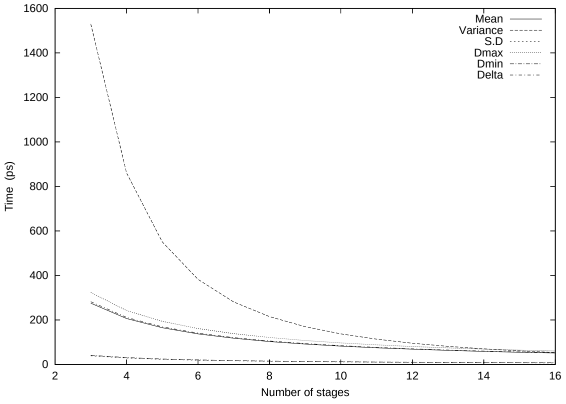


Figure 4.8: Statistic of ISCAS C432 path delays in the original design

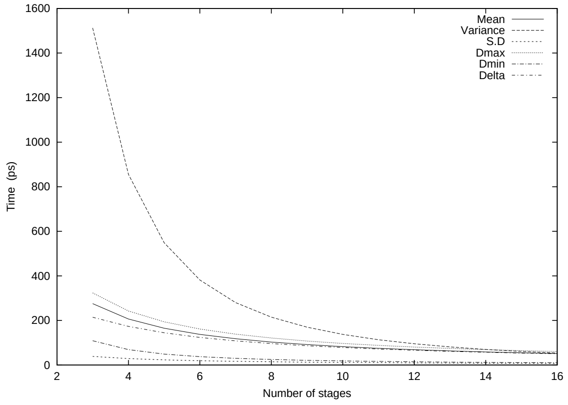


Figure 4.9: Statistic of ISCAS C432 path delays in the fault tolerant design with $np = 0.6, \gamma = 0.6$

Table 4.3: Statistic of ISCAS C499 path delays in the original design (ps)

Number of Stages	Mean	Variance	S.D.	D_{max}	D_{min}	Δ
3	239.355	1908.764	43.689	285.118	29.859	255.259
4	179.516	1073.68	32.767	213.838	22.394	191.444
5	143.613	687.155	26.214	171.071	17.915	153.156
6	119.677	477.191	21.845	142.559	14.929	127.63
7	102.581	350.589	18.724	122.193	12.797	109.397
8	89.758	268.42	16.384	106.919	11.197	95.722
9	79.785	212.085	14.563	95.039	9.953	85.086
10	71.806	171.789	13.107	85.535	8.958	76.578
11	65.279	141.974	11.915	77.759	8.143	69.616
12	59.839	119.298	10.922	71.28	7.465	63.815
13	55.236	101.65	10.082	65.796	6.89	58.906
14	51.29	87.647	9.362	61.097	6.398	54.698
15	47.871	76.351	8.738	57.024	5.972	51.052

Table 4.4: Statistic of ISCAS C499 path delays in the fault tolerant design with $np = 0.6, \gamma = 0.6$ (ps)

Number of Stages	Mean	Variance	S.D.	D_{max}	D_{min}	Δ
3	239.562	1834.537	42.831	285.118	91.121	193.997
4	179.633	1041.045	32.265	213.838	56.854	156.984
5	143.688	670.044	25.885	171.071	39.97	131.101
6	119.729	467.135	21.613	142.559	30.245	112.314
7	102.619	344.189	18.552	122.193	24.049	98.145
8	89.787	264.098	16.251	106.919	19.812	87.107
9	79.808	209.032	14.458	95.039	16.76	78.28
10	71.825	169.553	13.021	85.535	14.471	71.064
11	65.294	140.288	11.844	77.759	12.7	65.059
12	59.852	117.995	10.863	71.28	11.294	59.986
13	55.247	100.623	10.031	65.796	10.153	55.644
14	51.3	86.823	9.318	61.097	9.211	51.885
15	47.879	75.679	8.699	57.024	8.422	48.601

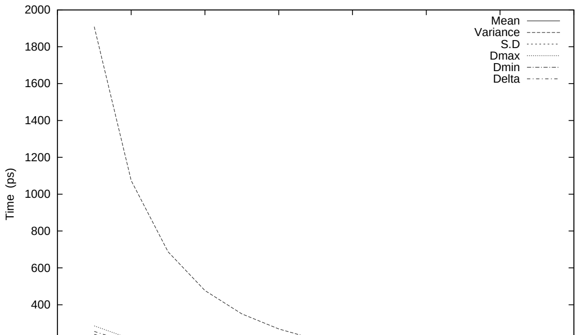


Figure 4.10: Statistic of ISCAS C499 path delays in the original design

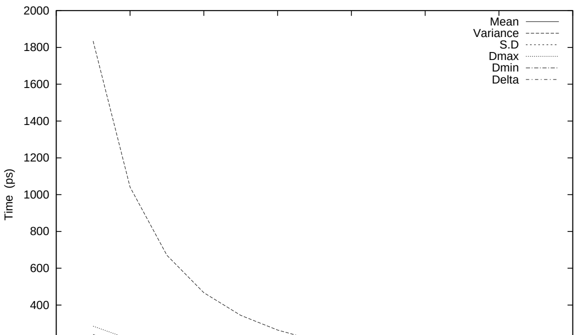


Figure 4.11: Statistic of ISCAS C499 path delays in the fault tolerant design with $np = 0.6, \gamma = 0.6$

Table 4.5: Statistic of ISCAS C880 path delays in the original design (ps)

Number of Stages	Mean	Variance	S.D.	D_{max}	D_{min}	Δ
3	294.938	5993.6	77.418	458.865	19.891	438.973
4	221.203	3371.4	58.064	344.148	14.918	329.23
5	176.963	2157.696	46.451	275.319	11.935	263.384
6	147.469	1498.4	38.709	229.432	9.946	219.487
7	126.402	1100.865	33.179	196.656	8.525	188.131
8	110.602	842.85	29.032	172.074	7.459	164.615
9	98.313	665.956	25.806	152.955	6.63	146.324
10	88.481	539.424	23.226	137.659	5.967	131.692
11	80.438	445.805	21.114	125.145	5.425	119.72
12	73.734	374.6	19.355	114.716	4.973	109.743
13	68.063	319.186	17.866	105.892	4.59	101.302
14	63.201	275.216	16.59	98.328	4.262	94.066
15	58.988	239.744	15.484	91.773	3.978	87.795

Table 4.6: Statistic of ISCAS C880 path delays in the fault tolerant design with $np = 0.6, \gamma = 0.6$ (ps)

Number of Stages	Mean	Variance	S.D.	D_{max}	D_{min}	Δ
3	296.008	5556.114	74.539	458.865	125.245	333.62
4	221.644	3225.019	56.789	344.148	74.18	269.969
5	177.175	2098.225	45.806	275.319	49.862	225.457
6	147.586	1470.12	38.342	229.432	36.284	193.148
7	126.477	1085.042	32.94	196.656	27.876	168.781
8	110.653	833.241	28.866	172.074	22.275	149.8
9	98.35	659.682	25.684	152.955	18.336	134.618
10	88.509	535.103	23.132	137.659	15.449	122.21
11	80.46	442.71	21.041	125.145	13.261	111.884
12	73.752	372.329	19.296	114.716	11.557	103.159
13	68.077	317.457	17.817	105.892	10.201	95.691
14	63.213	273.87	16.549	98.328	9.1	89.228
15	58.998	238.677	15.449	91.773	8.192	83.581

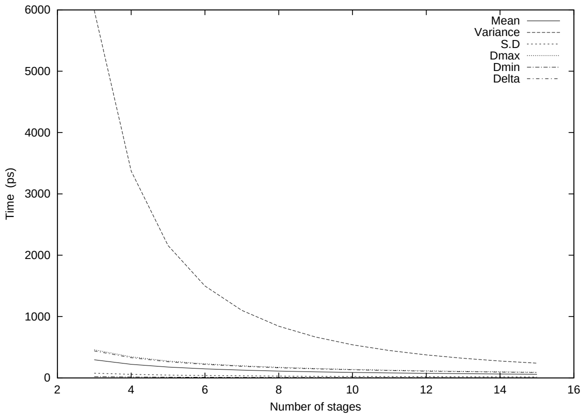


Figure 4.12: Statistic of ISCAS C880 path delays in the original design

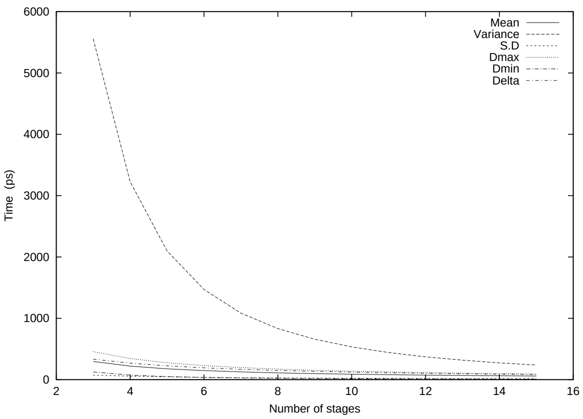


Figure 4.13: Statistic of ISCAS C880 path delays in the fault tolerant design with $np = 0.6, \gamma = 0.6$

remains unchanged. As a result, Δ, i.e., decreased from 282.1613 to 214.4426. The mean and variance all have no significant change. If L and α remained the same, $d_{min_i} - (L - \alpha_i - \delta_i)$ and $d_{min_j} - (L - \alpha_j - \delta_j)$ have become smaller. Therefore, according to Equation 4.6 and Equation 4.7, the intra-wave fault rate of the design with bipolar switches are theoretically reduced.

The statistical analysis results of circuit $C499$ and $C880$ with and without bipolar design are also conducted. Similar with the analysis of circuit $C432$, the statistical analysis results of the circuit, $C499$ clockless wave pipeline, without bipolar design are shown in Table 4.3 and Figure 4.10; the statistical analysis results of the circuit, $C499$ clockless wave pipeline, with bipolar design are shown in Table 4.4 and Figure 4.11; Table 4.5 and Figure 4.12 show the statistical analysis of the circuit $C880$ without bipolar design; and Table 4.6 and Figure 4.13 shown the statistical analysis of the circuit $C880$ with bipolar design. They all support the claim that we have for the circuit $C432$ as discussed above, that the bipolar design can reduce the intra-fault rate significantly by increasing the value of D_{min} of the circuit.

In order to further clearly demonstrate the difference of intra-wave fault rate between the design with and without bipolar switches, the intra-wave fault rates for the benchmark circuit $c432$ are calculated in the original design and the design with bipolar switches when np and γ are set to 0.8. The results are illustrated in Figure 4.14 and Figure 4.15. In both designs, the intra-wave fault rates decrease dramatically as the request signal length (L) increases, and as the circuit is divided into more stages, the intra-wave fault rate increases. The intra-wave fault rate with bipolar switches of the clockless wave-pipelined circuit is significantly lower than the equivalent circuit without bipolar switches. For example, the intra-wave fault rate is 0.0218%

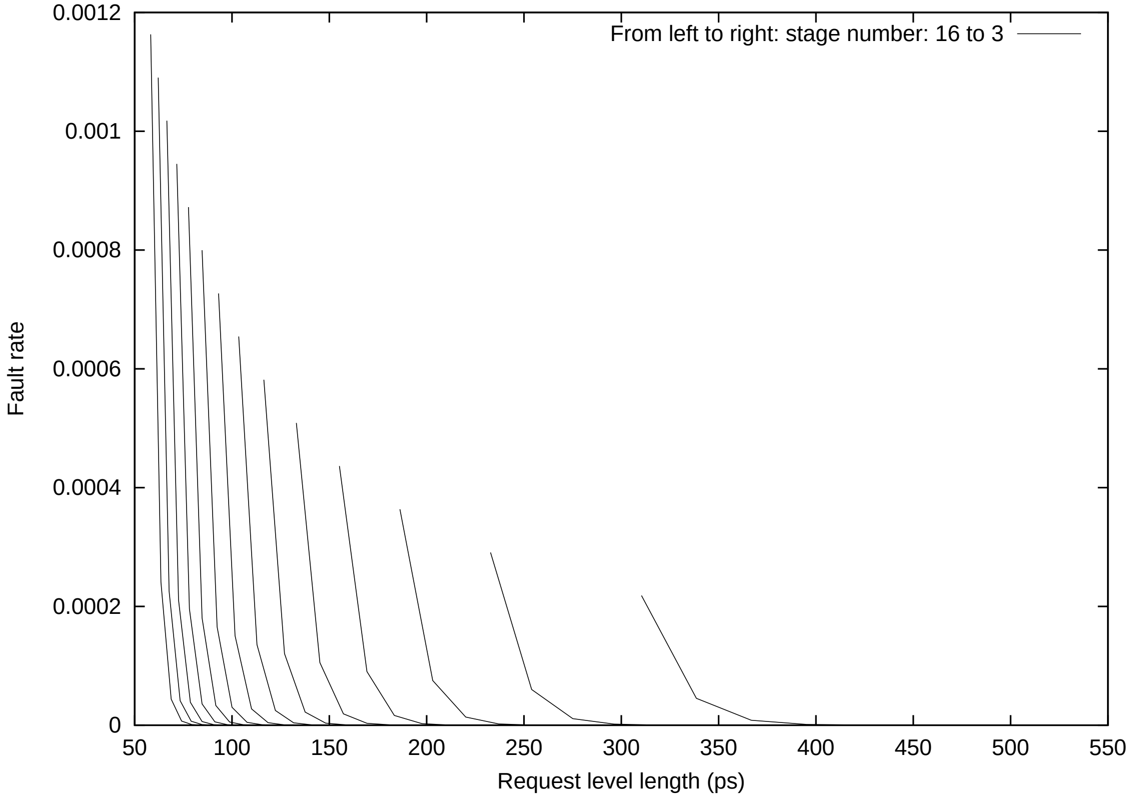


Figure 4.14: Intra-wave fault rate in the original design of $c432$

if the circuit has 3 stages and the request signal level is $310.377ps$ long in the original design, while the intra-wave fault rate of the new design drops to 0.01009% under the same circuit circumstance.

Figure 4.16 demonstrates the effectiveness of the bipolar switch-based design further by depicting the percentage of the intra-wave fault rate decrease in the new design of the benchmark circuit $c432$ compared with the original design. This comparison is conducted in the same circuit with different stage numbers varying from 3 to 16. For simplicity , we set α to be equal to 0.5 and set np and γ to be equal to 0.6. The detailed effectiveness of these parameters is researched and shown in the Figures 4.17 - 4.19. As shown in Figure 4.16, the bipolar switch-based design achieves 16.313% intra-wave fault rate decrease when the circuit has 3 stages, and

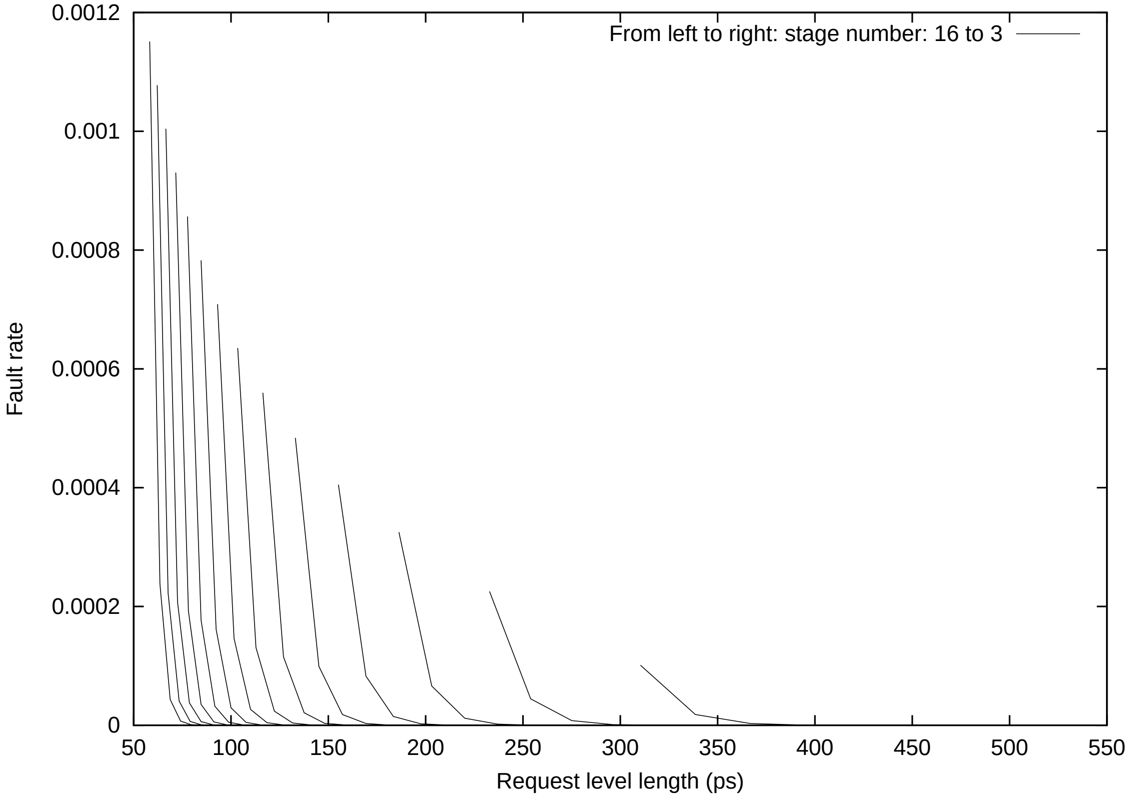


Figure 4.15: Intra-wave fault rate in the fault tolerant design of $c432$ with $np = \gamma = 0.8$

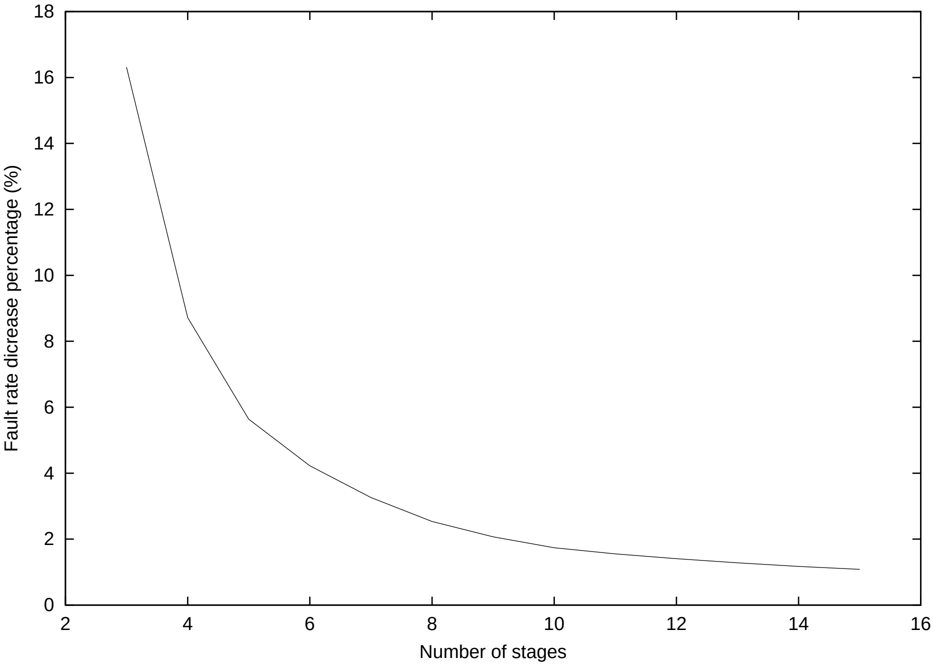


Figure 4.16: Decreased intra-wave fault rate with the fault tolerant design of $c432$, $\alpha = 0.5$, $np = \gamma = 0.6$

achieves 1.083% intra-wave fault rate decrease when the circuit has 15 stages. Thus, it reveals that even though the intra wave fault rate is higher if the same circuit being divided into more stages, the bipolar switch-based design is more effective in the circuit with less stages.

As illustrated in the Equation 4.6 and 4.7 , the parameters α, n, p, γ all play very important roles in the bipolar switch-based design. Thus, an experiment is designed to explore the effectiveness of those parameters on the intra-wave fault rate of the benchmark circuit $c432$. The experiment is carried out with focus on the following three analyses: to analyze the effectiveness of request signal length on intra-wave fault rate; to analyze the effectiveness of $np \times \gamma$ on intra-wave fault rate; to analyze the effectiveness of np and γ on intra-wave fault rate separately. They are illustrated in Figures 4.17 - 4.20, and 4.19.

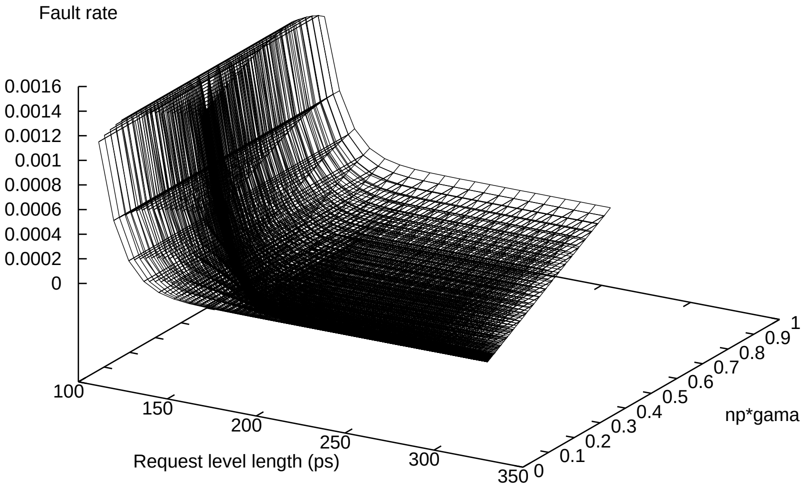


Figure 4.17: Effectiveness of $np \times \gamma$ and α in the fault tolerant design of $c432$ with stage number $= 5$

In Figure 4.17, the effectiveness of parameter α and $np \times \gamma$ on intra-wave fault rate is evaluated. Since $L = \Delta + 2\alpha$ and Δ is static once the number of stages in the circuit is fixed, the value of α actually determines the length of the request signal level. As shown in the figure, the intra-wave fault rates are reduced significantly as the request signal length changes from $100\,ps$ to $350\,ps$ if the circuit has 5 stages. This can be accounted for the effectiveness of α on the intra-wave fault rate indirectly. The parameter $np \times \gamma$ is also very critical to the intra-wave fault rate, while its effectiveness is not very visible as shown in this figure. Thus, in the next figure, they are further investigated.

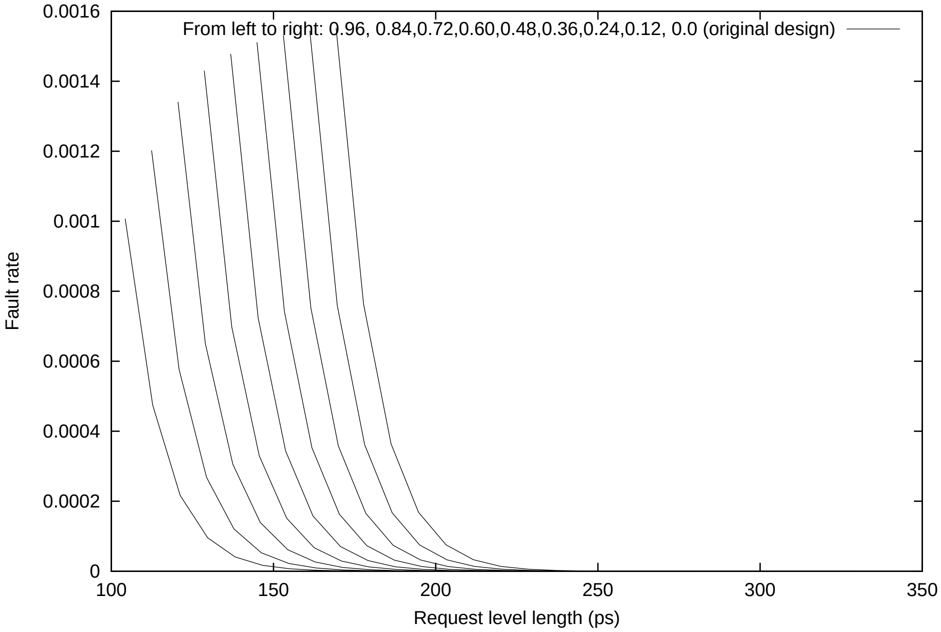


Figure 4.18: Effectiveness of $np \times \gamma$ in the fault tolerant design of $c432$ with stage number $= 5$

Figure 4.18 shows the effectiveness of $np \times \gamma$ as the intra-wave fault rates when stage number is 5 and the request signal length is in $100 - 350ps$. The higher the value of $np \times \gamma$ goes, the lower the intra-wave fault rate is. As shown in the figure, the intra-wave fault rate is 0.0075399% when $np \times \gamma$ equals to 0.96 and the request signal level equals to $203.15616\,ps$, while the intra-wave fault rate is 0.0000229% when $np \times \gamma$ equals to 0.24 at the same request signal level length. The situation of $np \times \gamma = 0$ is actually the case where there is no bipolar switches. As shown in the figure, the intra-wave fault rate when $np \times \gamma = 0$ is the highest. This is another evidence of the effectiveness of the bipolar switch-based fault tolerant design.

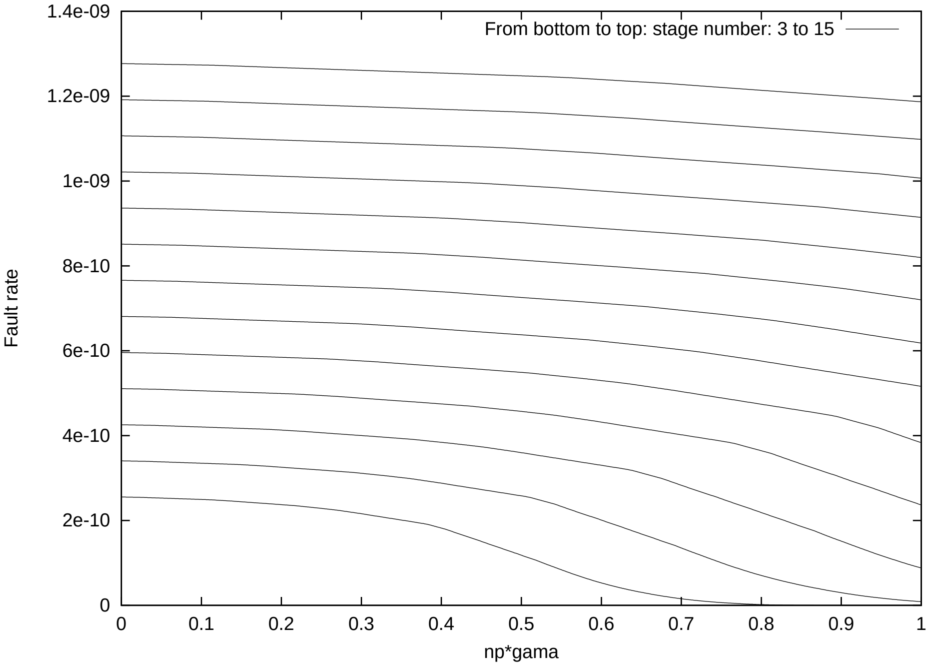


Figure 4.19: Effectiveness of $np \times \gamma$in the fault tolerant design of $c432$ with $\alpha = 0.8\Delta$

Figure 4.19 is another figure showing the effectiveness of parameter $np \times \gamma$ on the intra-wave fault rate with designs with respect to different stage numbers when $\alpha = 0.8$. The intra-wave fault rate increases greatly as the stage number increases, and reduces significantly as $np \times \gamma$ increases.

Figure 4.20 serves the purpose of further illustrating how the parameter of np and γ affect the intra-wave fault rate individually, when the circuit has 3 stages and α is 0.8. As shown in the figure, γ has significant effects on the intra-wave fault rate. The intra-wave fault rate drops greatly as γ increases from 0 to 1. np also affects the intra-wave fault rate considerably such that intra-wave fault rate drops as np increases. The effectiveness of np is particularly addressed and demonstrated in Figure 4.21 as $\gamma = 0.5$.

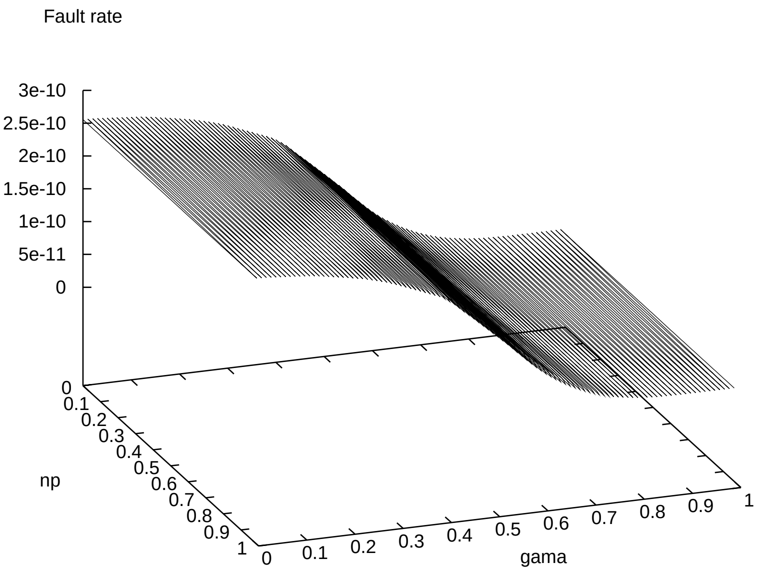


Figure 4.20: Effectiveness of np and γ in the fault tolerant design of $c432$ with stage number $= 3$, $\alpha = 0.8\Delta$

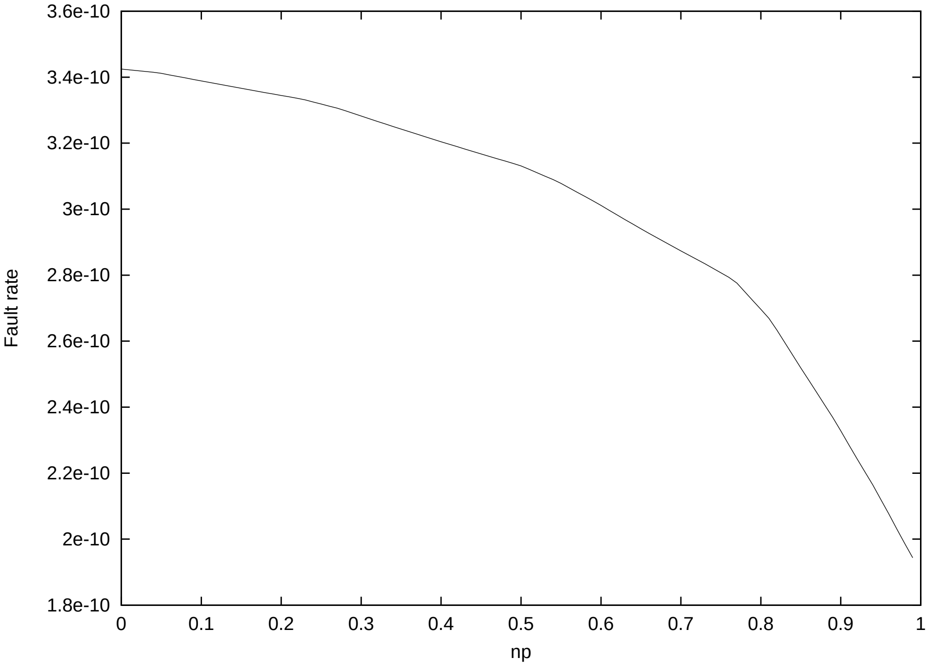


Figure 4.21: Effectiveness of np in the fault tolerant design of $c432$ with stage number $= 3$, $\alpha = 0.8\Delta$, and $\gamma = 0.5$

4.5 Conclusions and Discussion

In this chapter, the bipolar switches in association with affiliate request signals has been proposed to tolerate intra-wave fault on the datawaves of the 2-phase clockless wave pipeline. The basic framework of this reliability optimization technique was: it replaces every single polar switch in the original clockless wave pipeline design by bipolar switches, which consists of two switches with opposite polarity connected in series devised by attaching the opposite polarity switch to the one in the original clockless wave pipeline design; in addition, two extra request signals, signal N and signal P, are affiliated into the basic framework to control the newly attached switches; specifically, we use the affiliate signal N to control the newly attached P switches, and use the affiliate signal P to control the newly attached N switches.

These bipolar switches in association with affiliate request signals enable double alignment of the propagating bits of the datawaves in the 2-phase clockless wave pipeline. It is demonstrated that double alignment can significantly reduce the data skew along the datapaths. As the intra-wave fault is caused by the data skew on different data paths, the bipolar switches in association with the affiliated request signals can effectively slow down the propagation speed of the faster data bits without affecting the propagation of the slower data bits. Therefore, the data skew rate has been reduced and consequently the intra-wave fault rate is effectively reduced as well.

The effectiveness of the proposed fault tolerant technique is statistically verified by comparing the new and old probability density function of the path delays within the circuit and by calculating the integrals of these probability density functions. Moreover, the equations to calculate the intra-wave fault rate of the new 2-phase clockless wave pipeline with bipolar switches and affiliate request signals are provided.

The proposed fault tolerant model and design were simulated and experimented on ISCAS benchmark circuits $c432$, $c499$ and $c880$. The results show that the proposed bipolar switch-based design reduced the intra-wave fault rate significantly. On $c432$, it achieves 16.313% intra-fault rate decrease when the circuit has 3 stages, and achieves 1.083% intra-wave fault rate decrease when the circuit has 15 stages. Furthermore, more experiments were conducted to explore the effectiveness of different parameters, i.e., α, γ, n, p, and $number\ of\ stages$, on intra-wave fault rate of the new proposed bipolar switch-based clockless wave pipeline. The experimental results demonstrated that, specifically, α with $np{\times}\gamma$, $np{\times}\gamma$, np and γ individually, and $number\ of\ stages$ with $np{\times}\gamma$ play critical roles in the bipolar switch-based clockless wave pipeline design, and can effect the intra-wave fault rate substantially.

CHAPTER 5

THE PROPOSED NEW WAVE PIPELINE FOR MAXIMUM CIRCUIT SPEED AND RELIABILITY ASSURANCE

In this Chapter, a new approach to substantially enhance the circuit speed is proposed in the context of conventional wave pipeline. The proposed approach is based on the fact that there is more temporal space in combinational circuits to push the frequency of datawaves by allowing superimposition between adjacent datawaves. Due to the ultra-tight delay constraint by allowing datawave-superimposition, it is highly required to assure that there is no uncertainty region in terms of mutual exclusion between adjacent datawaves. Also, even a slight possible delay variation in the designed-delay of datawaves can be hardly tolerated in order to achieve a reliable data processing at the ultimately maximum circuit speed. Under the above mentioned delay-constraints, the dynamically bounded delay has to be taken into account in order to fully address the practical circuit-delay issues.

Therefore, the dynamically bounded system delay model as a basis is necessary to adequately characterize the datawave processing in the context of conventional wave pipeline. The performance advantage of the proposed technique over conventional wave pipeline will be theoretically demonstrated along with the cost for the higher reliability requirement. Also, a new output register architecture with delay buffers will be designed and presented in order to achieve the resulting performance and reliability of the proposed approach.

This chapter is organized as follows: The delay uncertainty issues in high speed circuits is addressed in the Section 5.1; The dynamically bounded system delay model is introduced and the new delay model in the context of conventional wave pipeline is proposed in the Section 5.2; Then the proposed new technique to maximize the circuit speed is presented in the Section 5.3; The proposed new wave pipeline technique is simulated and verified on ISCAS benchmark circuits in the Section 5.4; Then, this chapter is concluded in the Section 5.5.

5.1 High Speed Circuits and Delay Uncertainty

The number of transistor devices on an IC and the operating frequency have both been increasing dramatically ever since the first commercial integrated microprocessor developed by Intel Corporation [28] in 1971 built by using a 4-bit CPU implemented with 2,300 transistors and with the 8 micro-meter fabrication technology. Since then, the management of circuit delay and reliability assurance have become exigent issues for the design of high performance and speed integrated circuits. Today, as predicted by the Moore's Law, the density of the core processor built by Intel Corporation has reached 151 million transistor count with the 65 nano-meter fabrication technology and its operating circuit frequency has reached 2 GHz [81]. As the circuit operating frequency becomes higher and higher, the circuit is more and more susceptible to the delay uncertainty. The delay uncertainty can easily cause timing constraint violation within a system if not managed tightly and adequately.

In this context, in the state-of-the-art conventional wave pipeline technique-based processor cores, the circuit reliability assurance has been a much more critical issue in order to achieve the low-power and higher performance compared to the conventional pipeline-based circuits [76]. Since the proposed new technique to achieve the maximum possible circuit speed in combinational circuits exploits a further or possibly the furthest temporal or spatial space in order to push the clock frequency to its possible maximum, the assurance of the circuit reliability is the key to the success of the realization of the maximum circuit frequency.

There are various factors [34] [71] [5] reported to cause the circuit delay variation such as the transistor channel length, the dopant atom count in the semiconductor, the oxide thickness, the dielectric thickness, the Vcc value, and the temperature, to mention a few. Those variation factors may occur during function or physical-level design, fabrication, or even circuit-execution time. The delay variation due to fabrication process and environmental parameters are known to be one of the dominant factors that cause the uncertainty of delay of the signals within a circuit [71]. The variations in the power supply voltage [52], temperature [54], and electromagnetic effects [6] also are reported to affect the signal delay, as another major factors causing the delay and timing uncertainty. On transistor level, the variation of geometric and electrical parameters such as the effective channel length and threshold voltage are also known to be the possible causes to introduce delay uncertainty [73]. On wire level, capacitive and inductive coupling among wires may also cause delay uncertainty [66] [67] [29] as the device scale down below deep submicron. Furthermore on wire level, the geometry of the interconnects may cause delay variation due to limitation of the design and manufacturing process for highly dense and complex circuit routing [38], [80], [43].

On a functional level, another major contributor to circuit delay uncertainty is the fact that circuit delay can be dictated by data dependency of complementary metal-oxide-semiconductor (CMOS) circuit. The gate delay actually heavily depends on how many and which inputs are rising or falling, and the arrival time of those rising or falling inputs [65].

Traditional static delay models, which only consider the resistance and capacitance in a gate can not adequately and accurately capture the delay behavior within a circuit. Most commercial circuit simulators don't have the capability to compute practical circuit delay since they generally assume a static delay model without being able to incorporate the timing uncertainties due to the above mentioned causes for delay variations at various production phases.

Therefore, a delay model with consideration of the delay uncertainty and data dependency is necessitated especially for such circuit design methodologies as wave pipelining in which a tight control of delay is required. Gray [20] proposed a delay model which incorporates the data dependency for computing the circuit delays. Then, a bounded delay model was proposed [56], in which a bound $[d_{min}, d_{max}]$ can be computed for each circuit component. Furthermore, a dynamically bounded delay model [25] was proposed, in which the delay bound in [56] can be manipulated with the possible delay variations considered.

5.2 The Proposed Circuit Delay Model

The basic delay model can be viewed on the gate level and the delay information of each type of gate can be based on the data in a technology file for placement and routing purpose. The gate delay can be defined by the time period required for a bit propagation from the time when the signal of the bit crosses the threshold-voltage (either on the rising pulse or falling pulse depending on how the behavior of the gate is defined) until the time when the signal crosses back the threshold-voltage in the opposite direction from the input signal. The gate delay is essential to extend the circuit timing and analysis because it is the unit delay component of any extended delay computation; and the clock cycle must obey the fact that no input signal be injected into the circuit at least before the time interval required by the worst case gate delay. Note that due to the possible delay variation as discussed earlier, even on the gate delay level, a bound can be imposed. For instance, the dependency of delay on the data value of the signal can vary to the extent of the delay bound across a gate; and all other possible sorts of delay variation can necessitate the dynamic delay evaluation before determining the clock frequency of the circuit under design.

The gate delay model [82] considers two delay components: one is a constant component of the delay as referred to as the parasitic delay, the other is the dynamic component of the delay which is proportional to the load on the output signal as referred to as the effort delay. Also, the gate delay model [73] can manage both the cases of the output change to be observed at a rising edge and vice versa. Note that measurement technique of either the gate delay and path delay is beyond the scope of the work.

Delay path is a route of an input signal on a path as a chain of devices from a given input to a particular output. Hence, a path delay is the sum of all the gate delays and the wire delays on the path. Critical path is the path with the longest path delay, which generally determine the speed of the circuit. Likewise, the shortest path delay can be determined. Therefore, the path delay model [73] can be computed and evaluated cumulatively based on the gate delays on the path reaching at an observation gate. Thus, the bound on the delay on a path also carries on cumulating the delay bounds on the gates on the path, referred to as the bounded path delay.

When there exist more than one path in the circuit to reach at a common gate or device for signal observation, then the critical path delay among those multiple path delays determines the upper bound of the bounded path delay, and likewise the lower bound is determined by the shortest path delay. In fact, in the path delay with a bound, a further consideration of delay variation is to be made in order to address the possible delay faults that may cause an invalidation of the data signals unless otherwise tolerated.

5.2.1 The Proposed Bounded System Delay Model

The bounded delay model views the delay as a bounded interval instead of a static single value. The dynamic bounded delay model has extended the interval to a dynamically bounded interval in order to capture the delay variations due to different operating conditions. All the delays from primary input to primary output, or to any gate within the circuit, is the cumulative sum of the delays of all the gates on each path. The path delay is also bounded since gate delay on the path is bounded. However, the path delay bounds are not simply the sum of the bounds of all the nodes because the gate delay is affected by various operating conditions dynamically such that during a path delay computation, a possible delay variation due on the next gate delay may either be added to or subtracted from the overall cumulative path delay bound depending on the gate's positive or negative contribution.

For any arbitrary path, the actual path delay from the primary input to any node within a combinational circuit is falling into a range between d_{min} and d_{max}, where d_{min} represents the minimum propagation delay on the path, i.e, the shortest path delay, and d_{max} represents the maximum path delay on the path, i.e., the longest path delay. The snapshot of a bit propagation and its delay bound on a path within a combinational circuit can be illustrated as shown in Figure 5.1. The line in between d_{min} and d_{max} represents the bound, i.e., the interval of the propagation time, from the primary input to the node.

A datawave can be viewed as a vector of data which consists of multiple bits; hence there

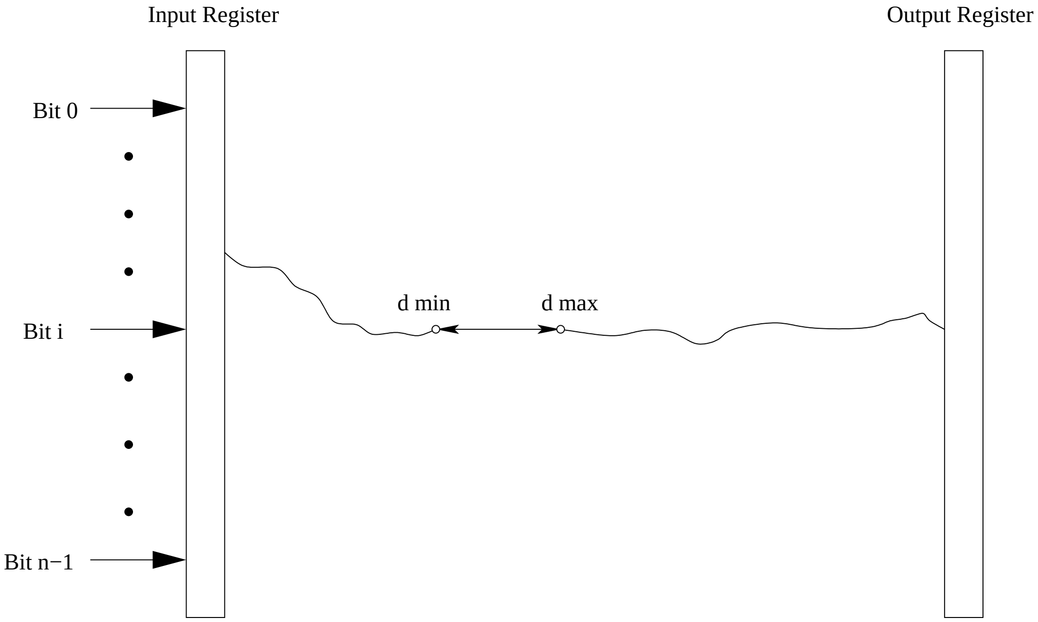


Figure 5.1: Bounded delay for a single bit

may exist multiple delay bounds, each of which exhibited by each bit possibly on a different path. Those multiple bounds can be represented by a polygon formed by the upper bound datawave and the lower bound datawave. Thus, at any instant of time during a datawave propagation, a datawave can be located within a polygon which is the possible range of span of the datawave and all the bits in the datawave can be covered by this polygon inclusively. The polygon may continue to mutate its shape during the datawave propagation since the cumulation of bounded path delay by each bit in the polygon continues to increase possibly at different rate determined by the path delay and the gate delays it takes. Note that there exists only one unique polygon for a datawave at any instant of time of propagation. In this context, the polygon in which the lower bound of the last front-end bit hits the primary output forms a unique polygon, and this is the polygon that enables us to compute the maximum clock frequency in this work.

It is defined as the polygon in which the lower bound of the slowest bit (since we use bounded delay model, each bit will hit the output within a delay interval with an upper and lower bound) hit the output. Technically, all the other bits are in the output delay buffers at that time. Yet this wouldn't affect our purpose, since, even though the bits are physically in the output delay buffers, their bounded path delay intervals can still be recorded and calculated, and these intervals are the all essential information necessary to summarize this polygon.

Polygon representations of datawaves are dynamically changed during the propagation process, thus there must exist one polygon that it is just big enough to cover every other polygon within the circuit. This polygon is called Global Maximum Polygon (GMP). It is guaranteed that there is no collision between datawaves if those GMPs are not overlapping, because every datawave is covered by a GMP inclusively.

There are two important intrinsic properties of the delay variation that decide which polygon is the GMP. First is the cumulative property of delay variation. Since the path delay is the sum of of all the gate delays and wire delays on the path, its variation is also the cumulation of gate delay variation and wire delay variation. Thus, the delay variance will always reach its maximum value at the end of the path. The second property is the similar property of delay variation. The main factors cause delay uncertainty, such as the power supply voltage , temperature, electromagnetic effects, i.e., all have the similar effect on each gate. Thus, two paths having similar gates on it also have the similar delay variance. These two properties are the prominent key points leading to the GMP, and can be confidently set up to the last instance of polygon in the circuit, i.e., the front end polygon.

The bounded delays of a datawave viewed in a polygon is used to define the proposed delay model referred to as Bounded System Delay Model in this work. Figure 5.2 illustrates a polygon of a datawave propagation in a $4 - bit$ combinational circuit.

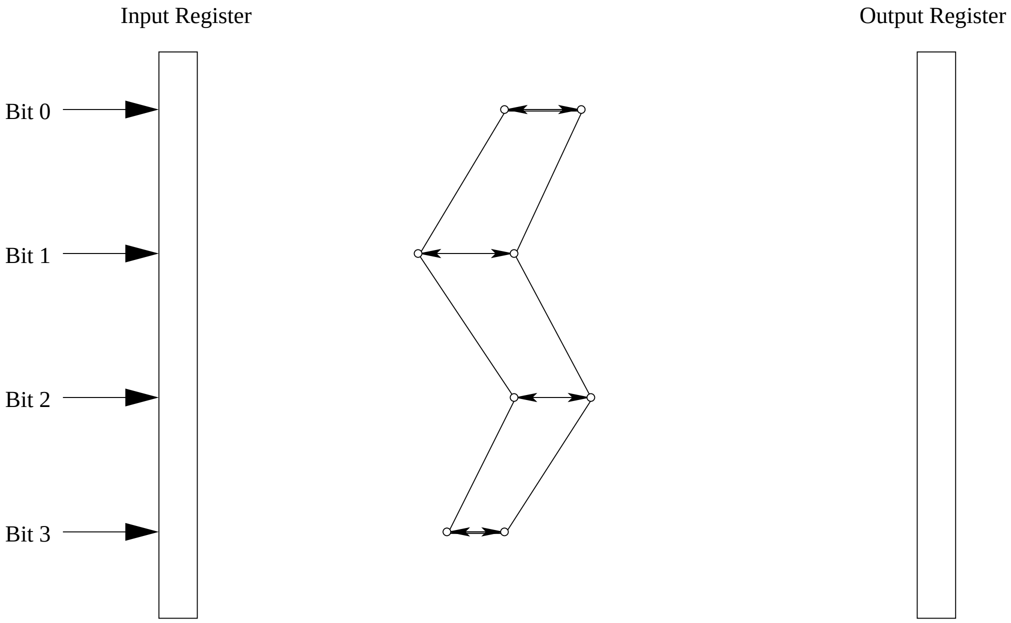


Figure 5.2: Polygon representation of a datawave

The polygon representation views delays at the system level, and the proposed bounded system delay model can view and manipulate delays at a system level temporally and spatially. It actually extends the concept of circuit delay from currently prevalent delay path-based level to system level. Instead of only paying attention to the critical path, or at most includes the shortest path as practiced in the delay path-based model, the bounded system delay model considers every path, and assigns a bounded interval to every path at every node along the path. Thus the system delay model is able to view the datawave propagation integratively, other than bit by bit.

The polygon can be interpreted temporally as the bounded intervals of the delay time from the primary input to a certain position, also it can be interpreted spatially as the bounded intervals of the datawave propagation physical locations after a certain period of propagation time. Even though temporal bounded system delay model and spatial bounded system delay model are conceptually different, both of them describe the same physical phenomena. The spatial model is good at illustrating and understanding the propagation situation of datawaves and the temporal model is more suitable at delay calculations.

The temporal/spatial bounded system delay model dynamically visualizes the datawave propagation situation. By locating the bounded system delay information into a coordinate system, the datawave propagation situation can be easily viewed at the system level.

Figure (5.3) shows a sample combinational circuit with 10 AND gates at the gate level, and each AND gate is assumed to have a bounded gate delay of $4 \sim 5$ ns. Figure 5.4 shows the temporal bounded system delay model extracted from the circuit module. The alphabets or numbers represent the input or output of the gates. They are used to represent paths within the circuit in this work. There are 23 paths totally in this circuit, namely hm8, hmz7, gm8, gmz7, gsy7, gs6, gsw5, f6, fw5, fx5, f4, epx5, ep4, e3, dpx5, dp4, d3, cv2, c1,biv2, bi1, aiv2, and ai1. The x-axis represents the temporal path delay measurement. The y-axis characterizes the arrangement of paths within the system. Whereas, the precise locations of the paths on the y-axis do not make differences to the visualized temporal/spatial bounded system delay model according to our purpose of timing analysis. So they are just arbitrarily distributed on the y-axis without being sorted as shown in Figure 5.4. The dashed line represents a propagating datawave with each bit properly located in between a pair of dots. The 23 pairs of dots simulate the 23 bounded intervals for the delay on the 23 paths. However, it doesn't mean that there are 23 physical bits propagating simultaneously from the input to the output. As seen in Figure 5.3, there are only 8 input bits and 8 output bits in the circuit, but the exclusive numeration of the 23 paths along with the 23 dots of the circuit are essential for our bounded system delay model, but the 23 bits can only be regarded as virtual bits.

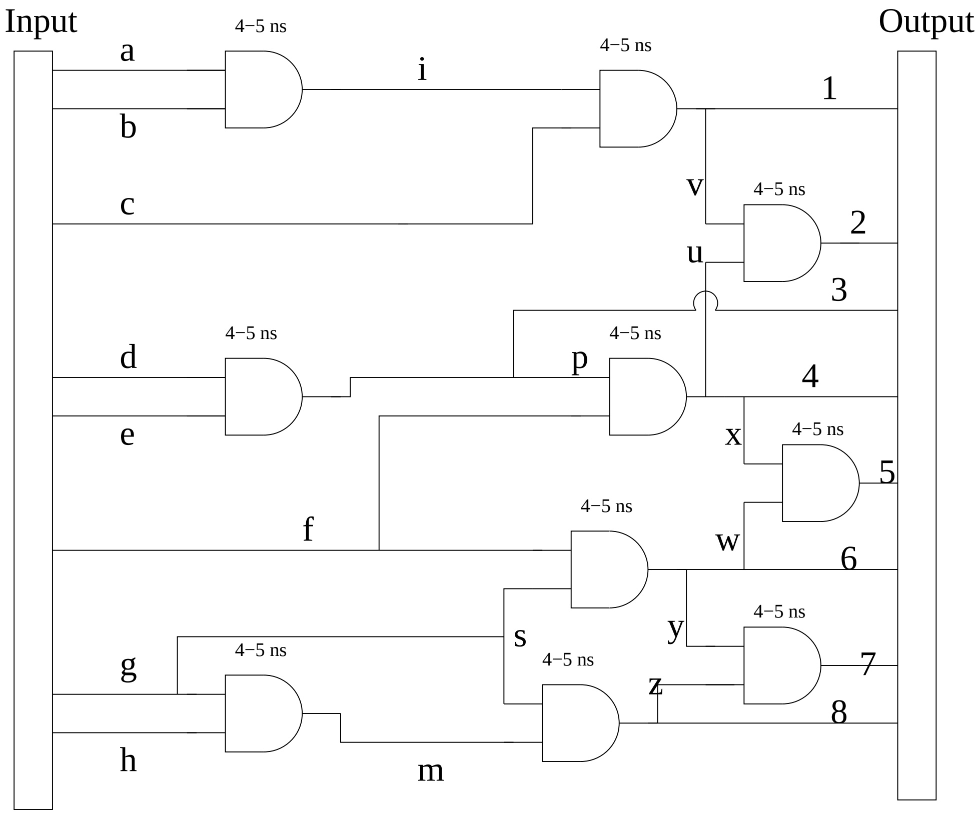


Figure 5.3: Example of a combinational circuit at gate level

If the x-axis indicates the spatial physical distance measurement (i.e., propagation depth) on each path other than the amount of temporal delay, it is a spatial bounded system delay model. The spatial bounded system delay model of the combinational circuit shown in Figure 5.3 is shown in Figure 5.5. The polygon represents a propagating datawave in the middle of the circuit. It shows the physical range after a certain period time of propagation, and the datawave is guaranteed to locate itself into it, with each bit being spatially bounded by the $length_{max}$ and

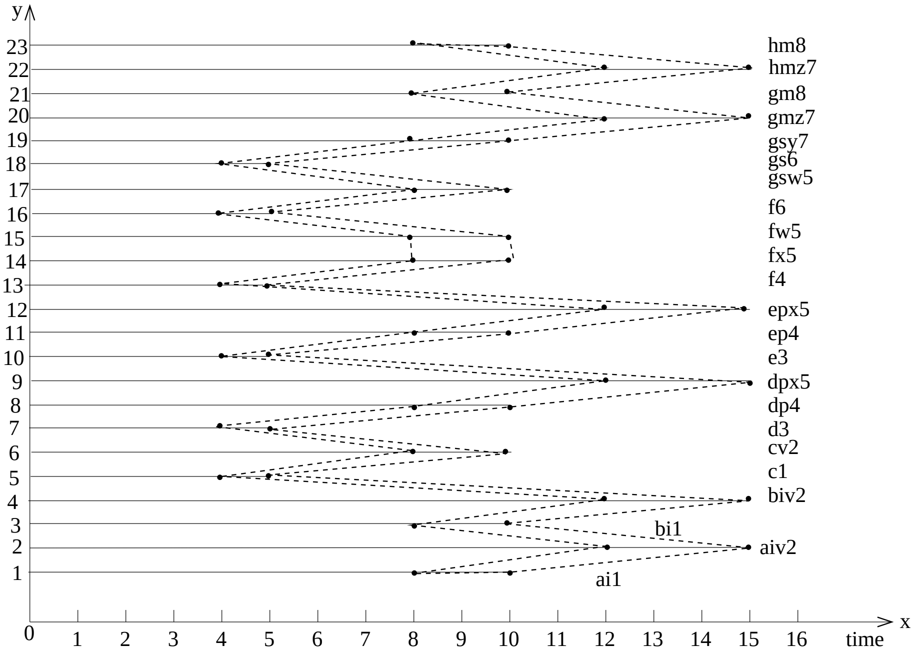


Figure 5.4: Extracted temporal bounded system delay model

$length_{min}$, which are represented by the two dots in the figure, where $length_{max}$ is the upper bound of the length interval, and $length_{min}$ is the lower bound.

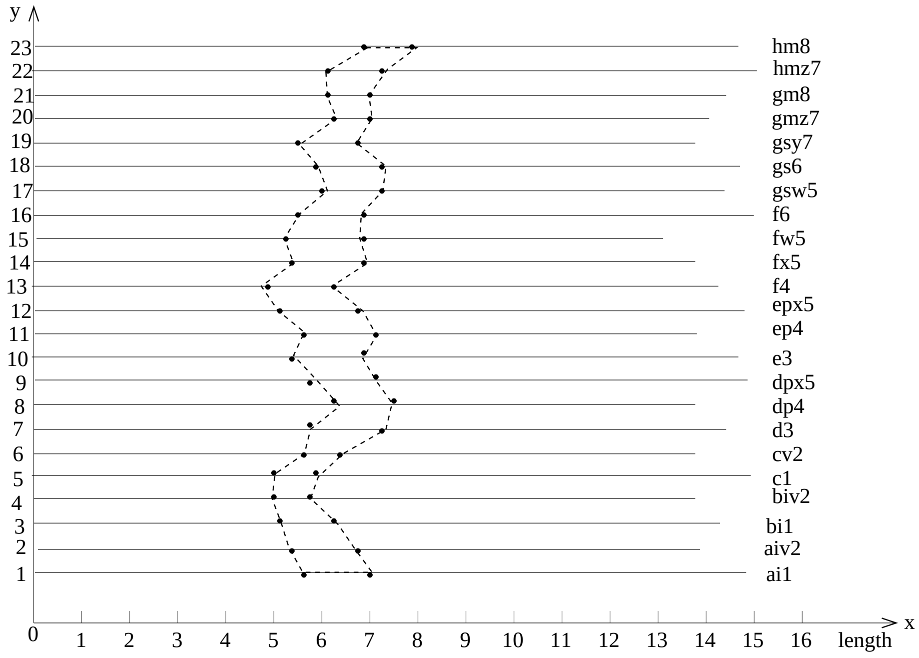


Figure 5.5: Extracted spatial bounded system delay model

The temporal/spatial bounded system delay model is applied to the conventional pipeline and conventional wave pipeline next. The conceptional difference of them is demonstrated clearly within the model.

5.2.2 Conventional Pipeline

In conventional pipeline, the circuit is divided into multiple stages by registers or latches. Each stage can have one and only one datawave propagating through at any instant of time. The clock cycle of conventional pipeline is determined by the longest path delay (D_{max}) of all the stages, where $D_{max} = MAX(d_{max_i}), i = 1, 2, ..., m$, m is the number of stages, d_{max_i} is the longest path delay in stage i. The datawave propagation situation of a conventional pipelined circuit is applied to the spatial bounded system delay model as shown in Figure 5.6.

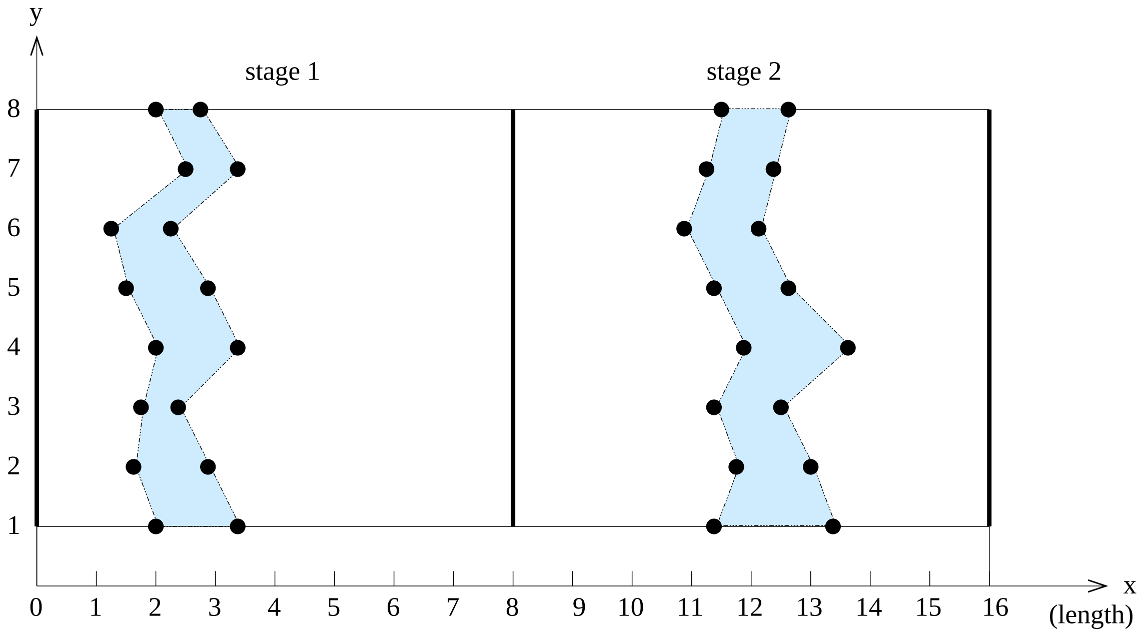


Figure 5.6: Conventional pipeline in the spatial bounded system delay model

There are 8 paths and two stages in the circuit. Each stage has one datawave propagating through. Thus, there are two polygons as shown in Figure 5.6. It is the snapshot of the propagation situation of these two datawaves. The polygon area is the guaranteed spatial region for the datawave to possibly stay at that instant of time. Theoretically, it can be thought that only the part of circuit which represented by the area of polygon is possibly used by the datawave.

In the spatial/temporal bounded system delay model, the actual circuit can be symbolized and represented by the geometric area in the coordinate system. The circuit utilization can be visualized as the precise polygon area of the datawave divided by the area that datawave officially occupied. Consequently, the circuit utilization of the conventional pipeline is that the area of the polygon divided by the whole stage area as shown in Figure 5.6 . Theoretically, conventional pipelined circuit has a relatively low circuit utilization comparing with conventional wave pipeline and the proposed new wave pipeline. Since the datawave officially holds the areas of the whole stage all the time from one register/latch to the follower register/latch, i.e., 0 to D_{max}, even though it can only utilize the area of its polygon takes.

5.2.3 Conventional Wave Pipeline

Conventional wave pipeline removed the internal register/latch from conventional pipeline. Thus there is no stage exist in it. Multiple datawaves can propagate simultaneously without being separated by register/latch. Instead of setting the clock cycle to longest critical path of all the stages, the clock cycle of conventional wave pipeline is determined by the difference of the longest path delay and the shortest path delay, i.e, $D = D_{max} - D_{min}$, where D_{max} and D_{min} are the longest path (critical path) delay and the shortest path delay for all the paths in the circuit. In our temporal/spatial bounded system delay model, the clock cycle can also be expressed as $MAX(d_{max_i} - d_{min_i}), i = 1, 2, ..., m$, where d_{max_i} and d_{min_i} are the maximum and minimum path delay of the polygon for datawave i, and m is the total number of datawaves in the conventional wave pipelined circuit. By this way, more datawaves can be accommodated by the same circuit. Therefore conventional wave pipeline achieves a faster clock cycle than conventional pipeline, also it has a higher circuit utilization than conventional pipeline as shown in Figure 5.7.

In the example as shown in Figure 5.7, there are still 8 paths and no internal registers/latches except the primary input and primary output. The front end polygon, i.e., the polygon for datawave 4, is the GMP. Since $D = MAX(d_{max_4} - d_{min_4})$, d_{max_4} is on the path 3, which is 16, and d_{min_4} is on the path 6, which is 12, the clock cycle for this conventional wave pipeline is $D = 16 - 12 = 4$. Recall that the conventional pipeline has a clock cycle of 8, thus this conventional wave pipelined circuit is two times faster than the conventional pipelined circuit. Accordingly, the circuit utilization is better for conventional wave pipelined circuit, because datawave in conventional wave pipelined circuit releases the circuit area once it passes through. Each datawave only occupies the amount of circuit area of $MAX(d_{max_i} - d_{min_i})$. In conventional pipeline, it occupies the area from 0 to D_{max}.

5.3 The Proposed Maximum Circuit Speed Based on the New Wave Pipeline

Based on the proposed temporal/spatial bounded system delay model, and the observation of the datawave propagation in conventional pipeline and conventional wave pipeline, the proposed new wave pipeline architecture for maximum circuit speed is proposed in this section.

5.3.1 The Proposed New Wave Pipeline

The new wave pipeline-based architecture is proposed practically and theoretically in this section. It is innovated based on the wave pipeline architecture and the temporal/spatial bounded

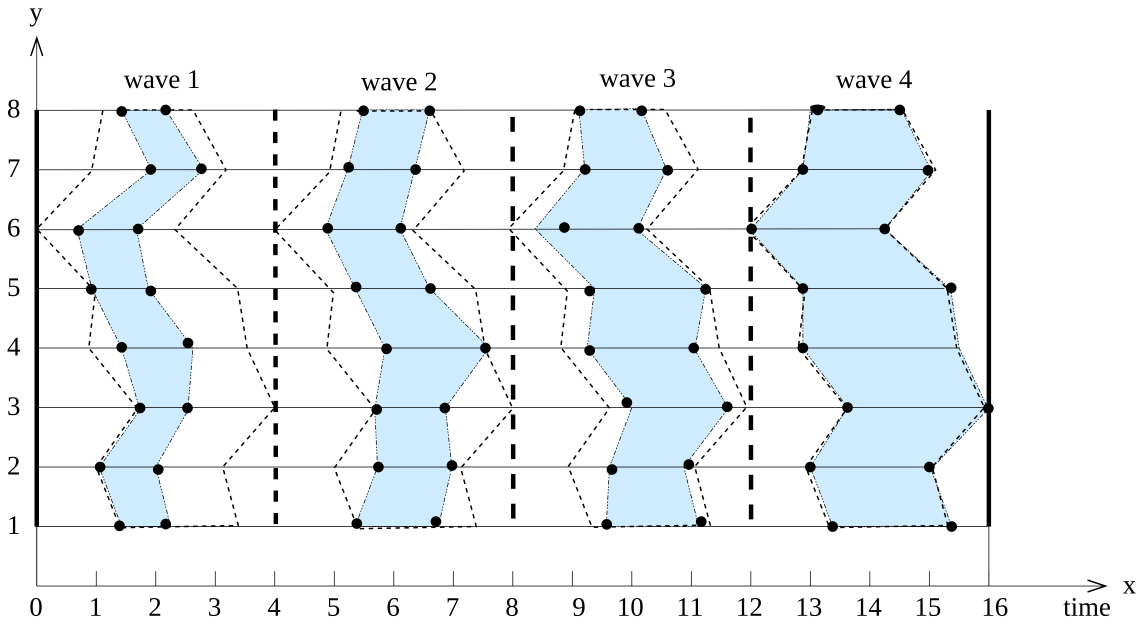


Figure 5.7: Conventional wave pipeline in the temporal bounded system delay model

system delay model. Generally, the proposed wave pipeline not only allows multiple datawaves propagating simultaneously like a conventional wave pipeline, but also breaks the concept that adjacent datawaves have to be temporally separated completely and clearly. The proposed new wave pipeline allows the bits belonging to adjacent datawaves are relatively mixed together. The bits belonging to the following datawave can temporally precede the bits belonging to the previous datawave. But the bits belonging to different datawaves propagating on the same path still have to be in order and still it is guaranteed that there is no bit collision between them.

The basic idea of the proposed new wave pipeline-based architecture is shown in Figure 5.8. On the left hand side, it depicts datawaves propagation situation in conventional wave pipeline. On the right hand side, it is the propagation situation of datawaves in the proposed new wave pipeline-based architecture. In the new architecture, the datawave on the left side is pushed closer to its preceding datawave, and it crosses the middle line which was there to temporally separate these two adjacent datawaves in conventional wave pipeline.

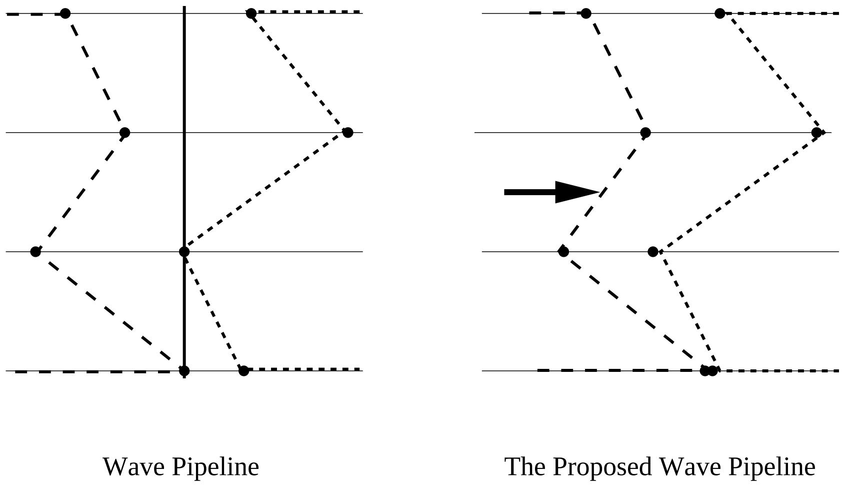


Figure 5.8: Concept comparison between conventional wave pipeline and the proposed new wave pipeline

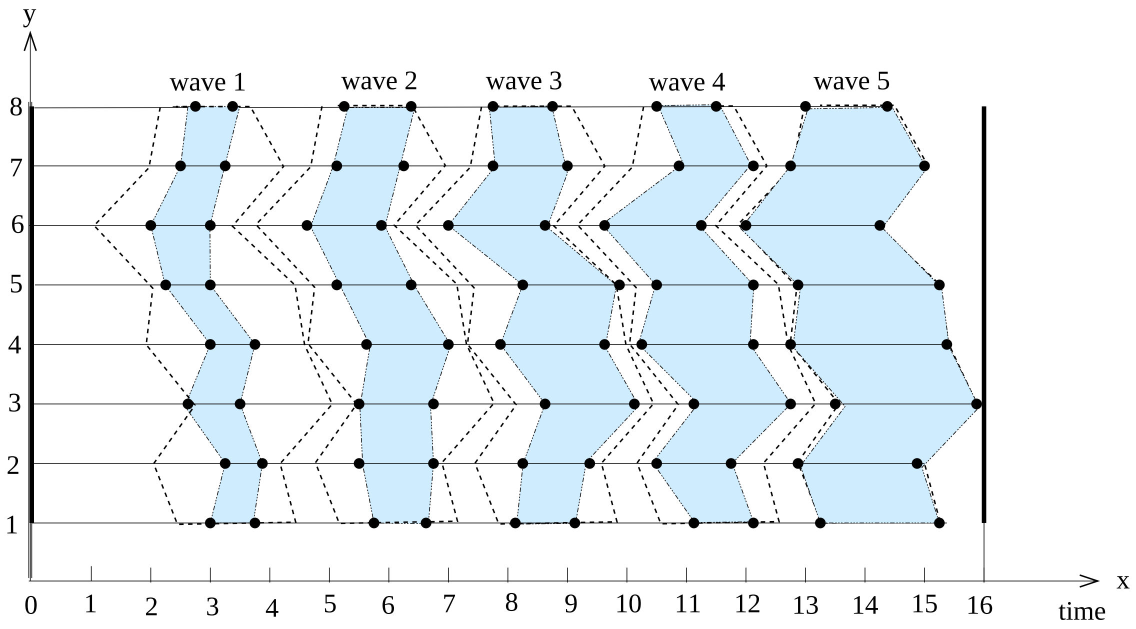


Figure 5.9: Proposed new wave pipeline in the temporal bounded system delay model

The datawave flow situation in our proposed new wave pipeline-based architecture is shown in Figure 5.9. The shaded polygons represent the propagating datawaves, and the dotted polygons represent the GMP. The arrangement of datawave in the proposed wave pipeline is denser than in conventional wave pipeline since the datawaves have been further pushed closer to each other.

The clock cycle of the proposed wave pipelined-based circuit is shorter than the conventional pipelined circuit because of the further pushing. The amount of time the proposed wave pipeline can further push is denoted by δ_{push} and the clock cycle of the the proposed wave pipeline is then $Clock_{wave} - \delta_{push}$.

The δ_{push} makes the proposed wave pipeline-based circuit has a shorter clock cycle and a better circuit utilization. As shown in Figure 5.9, the clock cycle of the the proposed wave pipelined circuit is reduced to 3 while it was 4 in conventional wave pipeline. The datawave in the proposed wave pipeline-based circuit only occupies the amount of area of the GMP, which is smaller than the area between $MAX(d_{max_i} - d_{min_i})$ as in conventional wave pipeline.

5.3.2 The Clock Cycle for the Proposed Maximum Speed Circuit

In order to find out the clock cycle of the proposed new wave pipeline-based architecture, the exact time the proposed wave pipeline can further push, which is δ_{push}, has to be found out. This can be calculated through the shape of the GMP. The two side borders (left and right borders) of the GMP actually can be viewed as two piecewise linear functions. So the question of calculating δ_{push} is changing to the question of finding the minimum distance between those two piecewise linear functions.

$Step$ 1: GMP's left side border and right side border:

An example of the two borders of a GMP extracted from a 4-path circuit is shown in Figure 5.10. $d_{j_{left}}$ and $d_{j_{right}}$ represent the left and right bound of the path delay on path j of the GMP.

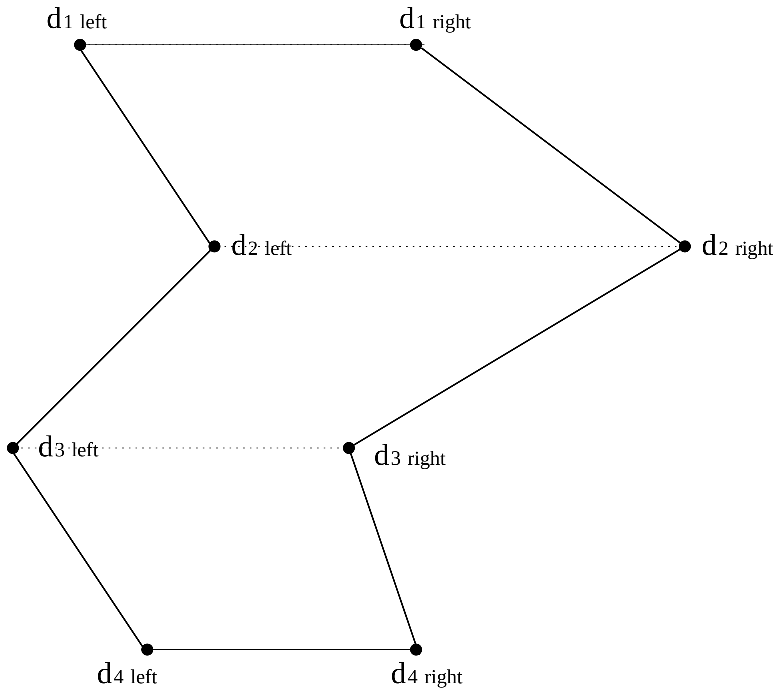


Figure 5.10: Two-sided borders of the global maximum polygon

$Step$ 2: Two functions:

The two borders are turned 90 degree counter clockwise and are located into a coordinate system with the y-axis represents the path delays and x-axis represents the order of the paths (how the paths being ordered will be discussed later). All the points are labeled as (a, b) and (a', b') as shown in Figure 5.11. (a, b) means that the left border of the GMP on path a has a path delay b, and (a', b') means that the right border of the GMP on path a' has a path delay b'. These two lines in Figure 5.11 actually can be represented by two piecewise linear functions: $y(x)$ and $y'(x)$, where $y(x)$ represents the left border and $y'(x)$ represents the right border respectively as shown in Equation 5.1 and 5.2.

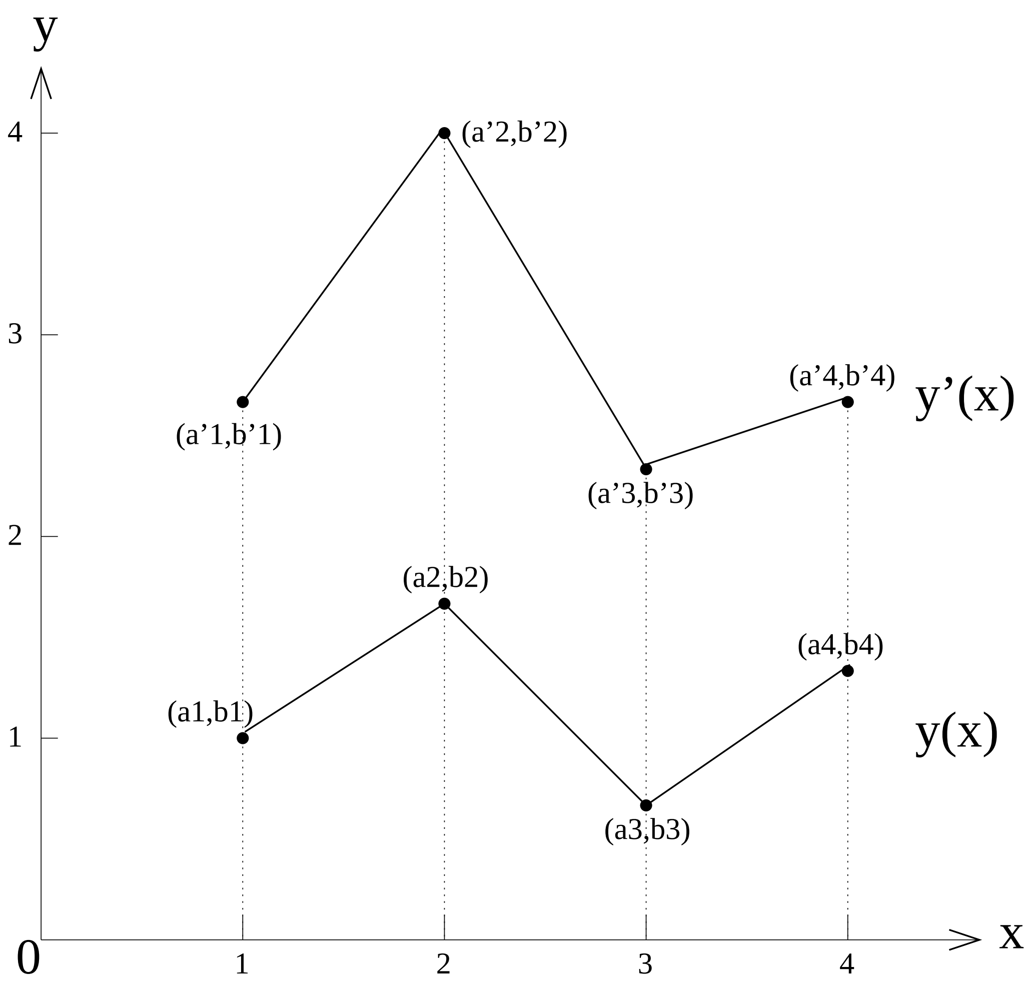


Figure 5.11: Two piecewise linear functions

$$y(x) = \frac{(b_{j+1} - b_j)}{(a_{j+1} - a_j)}(x - a_j) + b_j, for\ a_j \leq x \leq a_{j+1}\ and\ 1 \leq j < n \tag{5.1}$$

$$y'(x) = \frac{(b'_{j+1} - b'_j)}{(a'_{j+1} - a'_j)}(x - a'_j) + b'_j, for\ a'_j \leq x \leq a'_{j+1}\ and\ 1 \leq j < n \tag{5.2}$$

where $y(x)$ and $y'(x)$ are the two piecewise linear functions representing the left and right borders of the GMP, where y is the path delay and x is the order of the path, a_j is the order of the j_{th} path, b_j is the path delay on the left border of GMP on path j, a'_j is also the order of the j_{th}, b'_j is the path delay on the right border of GMP on path j, and n is the total number of paths.

since a_j=a'_j for $1 \leq j \leq n$, Equation 5.2 can be rewritten as:

$$y'(x) = \frac{(b'_{j+1} - b'_j)}{(a_{j+1} - a_j)}(x - a_j) + b'_j, for\ a_j \leq x \leq a_{j+1}\ and\ 1 \leq j < n \tag{5.3}$$

$Step\ 3$: Repositing $y(x)$.

Assuming datawaves propagate from left to right, datawave collisions have to take place between adjacent datawaves, and thus the distance between the left side border of the previous GMP and the right side border of the following GMP need to be studied. In order to figure out how much space can be further pushed by the new proposed wave pipeline-based architecture, the two functions $y(x)$ and $y'(x)$ need to be repositioned by switching $y(x)$ to be on top of $y'(x)$. It can be done either by moving $y(x)$ up or moving $y'(x)$ down. Here, we choose moving $y(x)$ up.

The destination of moving $y(x)$ needs to be figured out before the moving. Because δ_{push} is the final aim of all these calculation and it is assumed that the push start from the datawaves arrangement in a conventional wave pipeline. According to this, the $y(x)$ needs to reposition itself to conform the conventional wave pipeline datawaves arrangement. Conventional wave pipeline arranges its datawaves based on $d_{max_i} - d_{min_i}$, for $i = 1, 2, ..., m$, where d_{max_i} and d_{min_i} represent the path delay of the slowest and fastest bits in datawave i, m is the number of datawaves within the wave pipelined circuit. That means, theoretically, in conventional wave pipeline, the slowest bit of the previous datawave can at most has the same propagation delay as the fastest bit of the following datawave. Namely, in the coordinate system, the destination of the reposition of $y(x)$ should be the position where the lowest vertex of the new $y(x)$, namely $u(x)$ has the same height as the highest vertex of $y'(x)$. This situation is shown in Figure 5.12. It is actually the start point before pushing further to achieve the proposed new wave pipeline design.

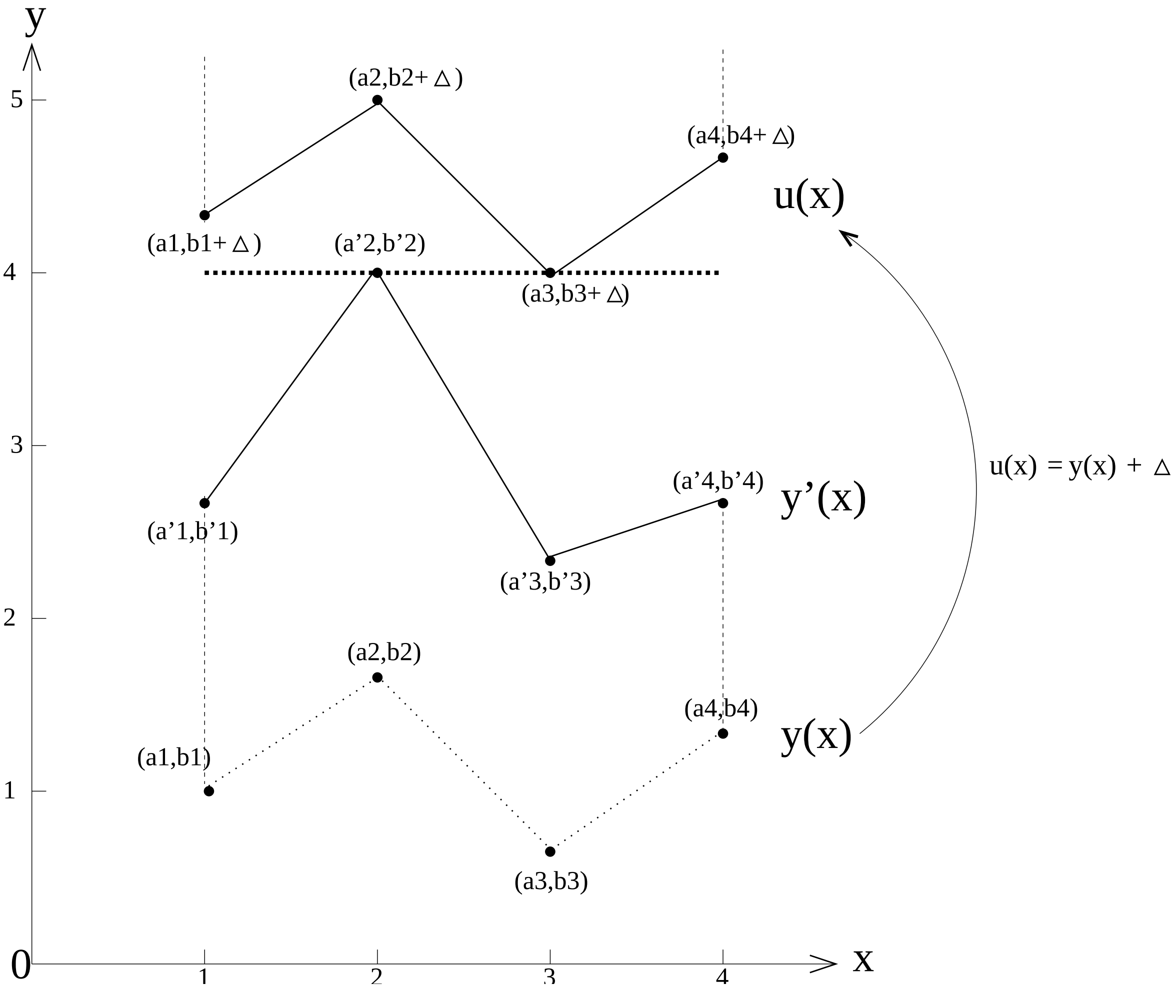


Figure 5.12: Repositing $y(x)$ to the starting point of pushing

Let Δ be the amount of distance $y(x)$ need to move up.

$$\Delta = max(y'_j) - min(y_j), for\ j = 1, 2, ..., n-1 \tag{5.4}$$

Function $y(x)$ changes to $u(x)$ after the relocation.

$$\begin{aligned} u(x) &= y(x) + \Delta \\ &= \frac{(b_{j+1} - b_j)}{(a_{j+1} - a_j)}(x - a_j) + b_j + \Delta, for\ a_j \leq x \leq a_{j+1}\ \ and\ \ 1 \leq j < n \end{aligned} \tag{5.5}$$

$Step\ 4$: Calculating $\delta(x)$

The question of how further can function $u(x)$ been pushed towards $y'(x)$ changes to the question of calculating the minimum value of the distance function $\delta(x)$, where $\delta(x) = |u(x) - y'(x)|$.

$$\begin{aligned} \delta(x) &= |u(x) - y'(x)| \\ &= |\frac{(b_{j+1} - b_j)}{(a_{j+1} - a_j)}(x - a_j) + b_j + \Delta - (\frac{(b'_{j+1} - b'_j)}{(a_{j+1} - a_j)}(x - a_j) + b'_j| \\ &= |\frac{(b_{j+1} - b_j) - (b'_{j+1} - b'_j)}{(a_{j+1} - a_j)}(x - a_j) + (b_j - b'_j + \Delta)| \\ & \quad for\ a_j \leq x \leq a_{j+1}\ \ and\ \ 1 \leq j < n \end{aligned} \tag{5.6}$$

So, the amount of clock cycle the proposed wave pipeline-based circuit can further save compared with conventional wave pipeline design is:

$$\delta_{push} = min[\delta(x)] \tag{5.7}$$

Figure 5.13 shown the result of pushing $u(x)$ towards $y'(x)$, and function $d(x)$ is the result of pushing function $u(x)$.

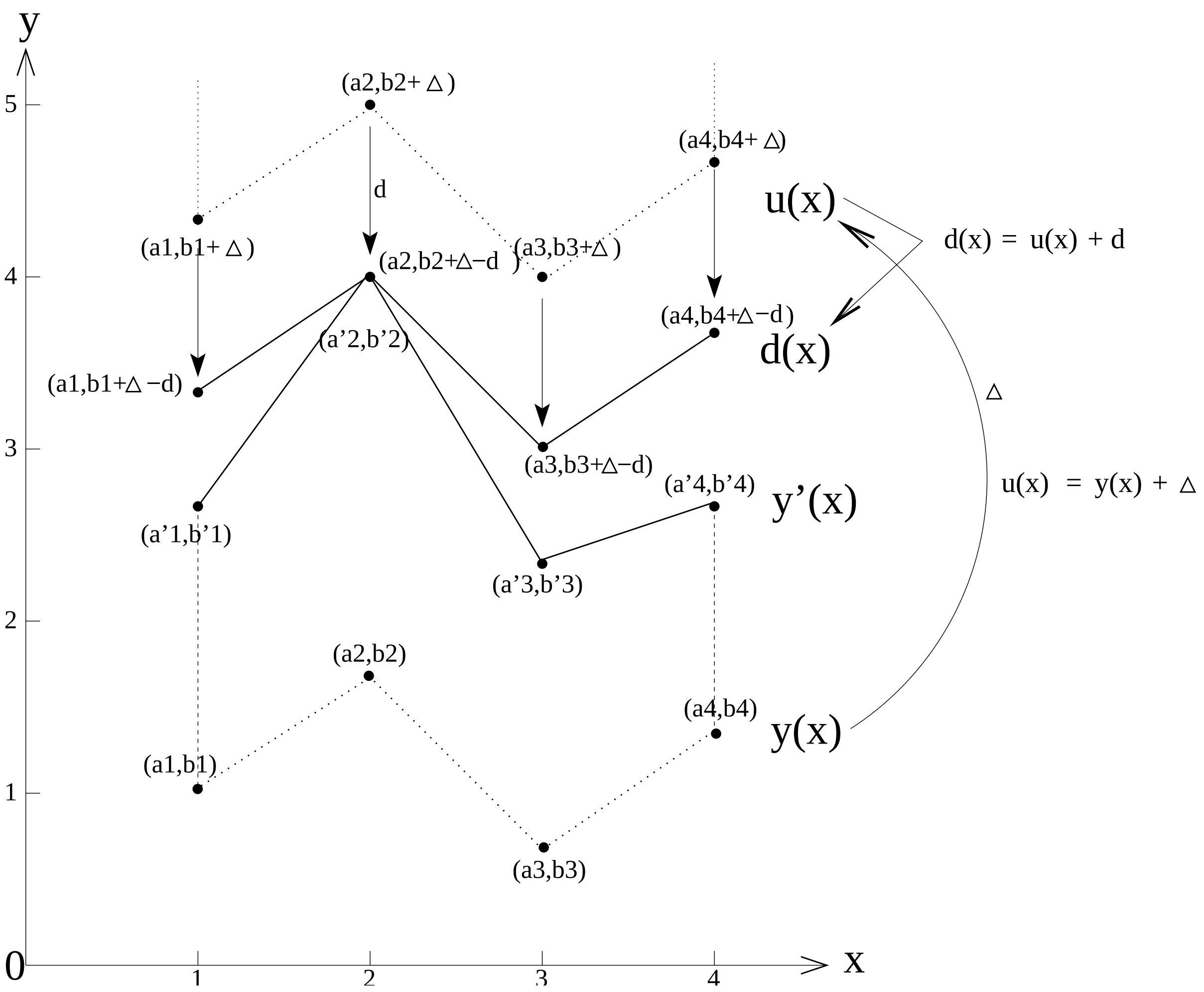


Figure 5.13: The final step of the functions reposition

Finally, the clock cycle in the proposed wave pipeline is:

$$ClockCycle_{max} = D_{max} - D_{min} - \delta_{push} \tag{5.8}$$

$$= D_{max} - D_{min} - min[\delta(x)] \tag{5.9}$$

where $min[\delta(x)]$ is the minimum value of the distance function $\delta(x)$.

The valid clock cycle of the proposed new wave pipeline is chosen based on the situation where $u(x)$ and $y'(x)$ has the minimum distance with each other. On the other hand, it is the same situation where $y'(x)$ and $y(x)$ has the maximum distance with each other as proved in Equation 5.10 as follow.

Let $\varphi(x)$ be the distance function between the two piecewise functions $y(x)$ and $y'(x)$, i.e., $\varphi(x) = |y'(x) - y(x)|$. It actually represents the path delay bounded interval on the path x. then

$$\begin{aligned}
\varphi(x) &= |y'(x) - y(x)| \\
&= |y'(x) - (u(x) - \Delta)| \\
&= |y'(x) - (u(x) + \Delta| \\
&= | - \delta(x) + \Delta|
\end{aligned} \tag{5.10}$$

Thus, $\varphi(x)$ has its maximum value when $\delta(x)$ has its minimum value. Considering in the context of GMP shape, after pushing the amount of δ_{push}, two adjacent GMP have the attaching point with each other on the path which has the maximum path delay bounded interval. It means that during the process of propagation, two GMPs are actually temporally separated by the amount of time equal to the maximum path delay bounded interval within the GMP, i.e., $max[(\varphi(x)]$. Thus, the clock cycle set for the proposed new wave pipeline-based circuit is equal to the maximum path delay bounded interval within the GMP, i.e., $D_{max} - D_{min} - min[\delta(x)] = max[\varphi(x)]$.

Hence, the order of paths in the bounded system delay model can not affect the valid clock cycle of the proposed new wave pipeline, even though it can greatly rule the shape of the polygon. Therefore, the paths are just arbitrarily distributed on the y-axis without considering their order in the proposed bounded system delay model.

The value of the proposed new wave pipeline-based architecture is highly dependent on the shape of GMP, while the shape can be easily adjusted by many different ways just like the practice of delay balancing commonly used in wave pipelining. The scenarios with best case and worst case GMP shape are analyzed as follow.

Worst Case clock cycle Analysis:

Apparently the worst case happens when there is no potential space to push from the conventional wave pipeline, i.e., $\delta_{push} = 0$. That is the slowest bit of the previous datawave and the fastest bit of the following datawave is on the same path, i.e., the GMP has the biggest and smallest delay bound on the same path in the context of GMP shape. In this case, the proposed wave pipeline is just as same as conventional wave pipeline. It is shown in the Figure 5.14.

Mathematically, the worst case can be defined as follow, referring to Figure 5.13, Equation 5.3 and Equation 5.1.

1 Let $y(x_1) = max[y(x)]$ and $y'(x_2) = min[y'(x)]$

2 It is the worst case if and only if $x_1 = x_2$

Best Case clock cycle Analysis:

The best case can be achieved when all the available potential spaces can be pushed. That means $u(x)$ can be pushed all the way to completely superimpose $y'(x)$. It then reaches the maximum gain on the clock cycle and circuit utilization. It happens if and only if the two piecewise linear functions $u(x)$ (or $y(x)$) and $y'(x)$ are parallel to each other on every piece, in addition, the distance of each pair pieces are all same, i.e., the GMP's left and right border are

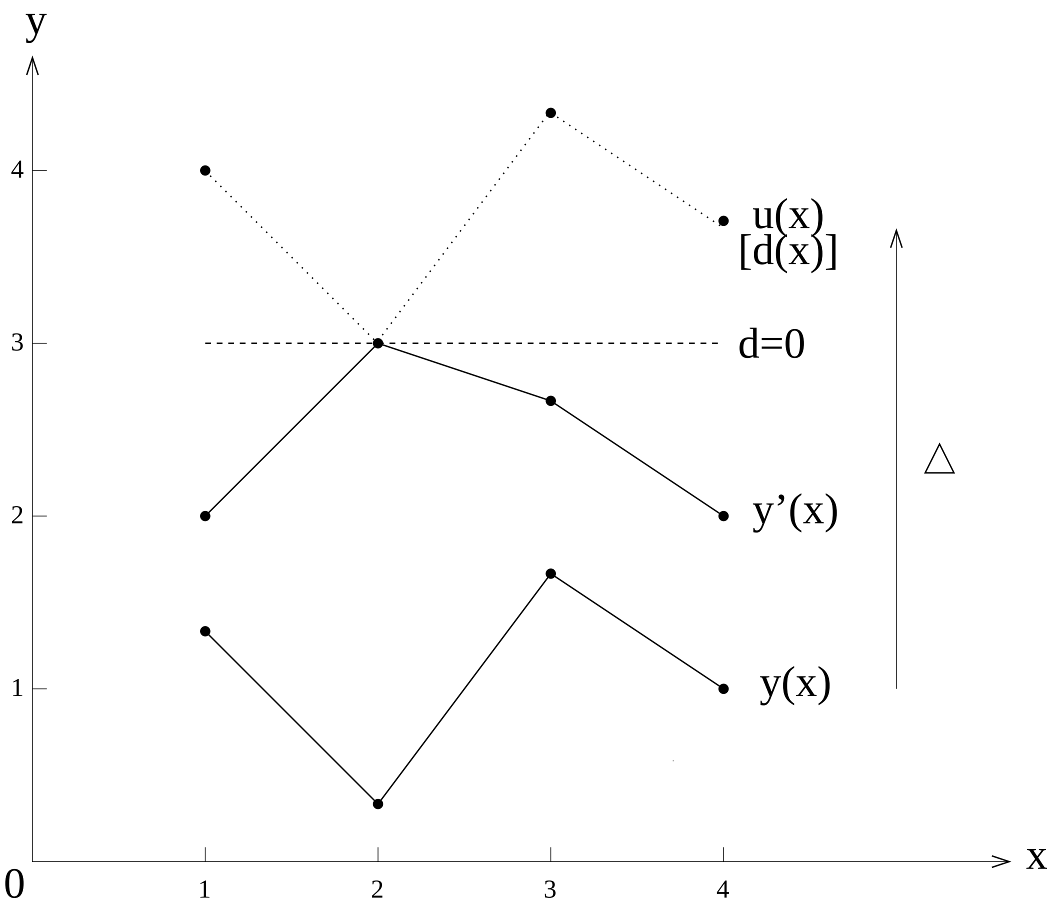


Figure 5.14: Worst case of the proposed wave pipelining

parallel to each other for all the pairs of pieces of lines in the context of GMP shape, besides, the distance between all the pairs of pieces of lines are same.The best case scenario is shown in Figure 5.14. Not that there still exists the minimal delay constraint that no gate can be accessed by more than one datawave at an instant of time of propagation, and this implies the worst case gate delay between adjacent datawaves determines the minimal delay requirement even in case of perfectly parallel datawaves.

Mathematically, the best case can be achieved if the follow equations hold, referring to Equation 5.3 and Equation 5.1.

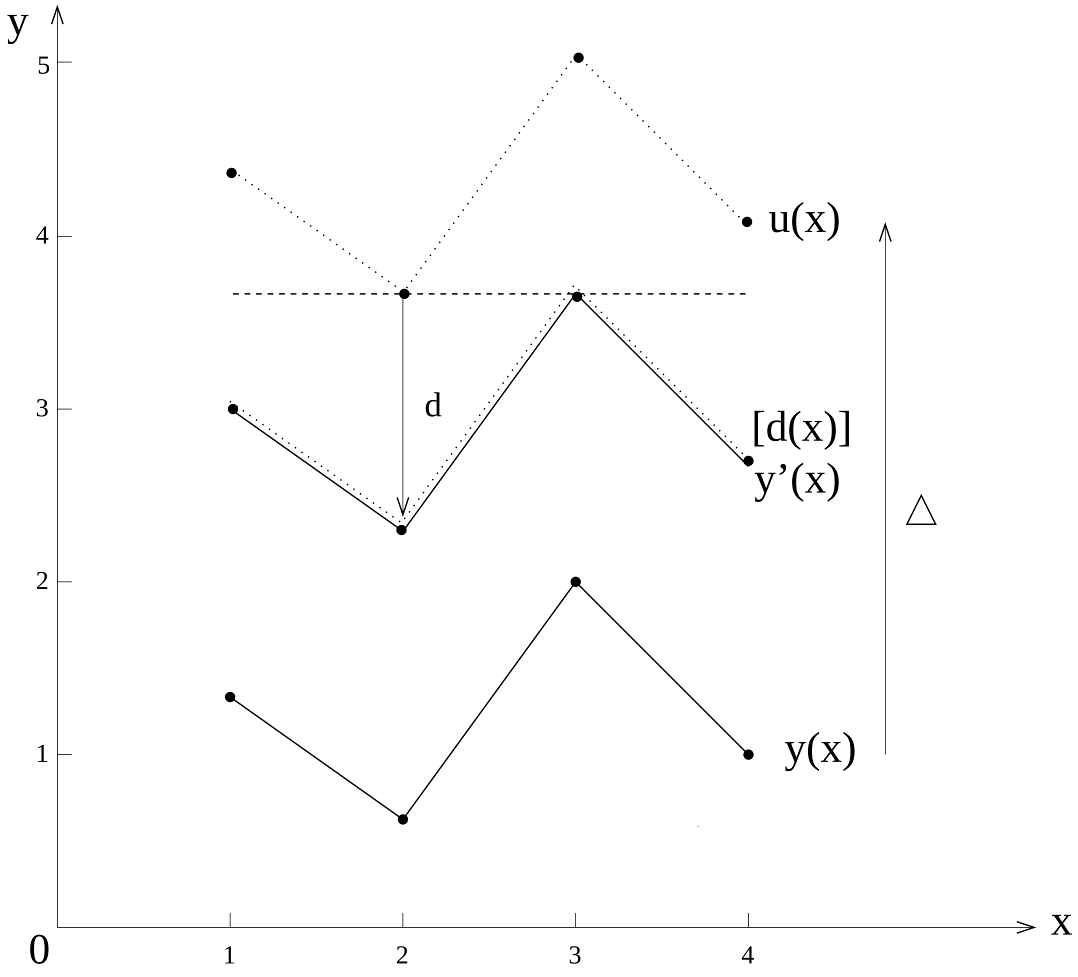


Figure 5.15: Best case of the proposed wave pipelining

1 $\frac{b_{j+1}-b_j}{a_{j+1}-a_j} = \frac{b'_{j+1}-b'_j}{a'_{j+1}-a'_j}$

2 $b_j = b'_j + c, where\ c\ is\ a\ constant, for\ j = 1, 2, ..., n-1$

5.3.3 Output Delay Buffers

The proposed new wave pipeline successfully achieves relative mix of datawaves during the propagating process. Bits belonging to different logic datawaves (has to be adjacent datawaves) can propagate simultaneously within the same circuit without being arbitrarily temporally separated. By this way, a high clock frequency can be accomplished.

However, the mixed bits belonging to different logic datawaves have to be sampled out at the output area according to their original logic datawaves. The conventional output registers/latches can not fulfill this task obviously. Therefore, a new technique is proposed in this section. It attaches arrays of delay buffers to the end of the combinational circuit to adjust the path delays. The objective of this is to change the GMP to a relative simpler polygon. Consequently, the mixed bits can be rearranged back to its original logic datawave, and datawaves can be clearly temporally separately again. Then, the logic datawaves are able to be sampled by the output registers.

The algorithm of how to attach the output delay buffers to the end of the paths are described as follow:

1. Suppose there are n paths within the front end polygon, which is actually the GMP.

2. The GMP has the left and right bound on each path j, i.e., $d_{j_{left}}$ and $d_{j_{right}}$, for $1 \leq j \leq n$.

3. The maximum left bound for all paths is $d_{left_{max}}$ on path max.

4. For each path j,attach the amount of delay buffers: $d_{left_{max}} - d_{j_{left}}$ at the end of the path.

5. For path max, do not attach any delay buffers.

Thus, the GMP changed to a polygon in which all the path delay intervals have the same left bound. By this way, the relative mixed bits are separated and can be observed by the conventional output register/latch as shown in Figure 5.16.

The attached delay buffers increased the path delay for every path except the one that has the maximum left bound. The left sides of all the path delay intervals in the polygon are lined up by the attached delay buffers. Thus the shape of the polygon is changed from the GMP to

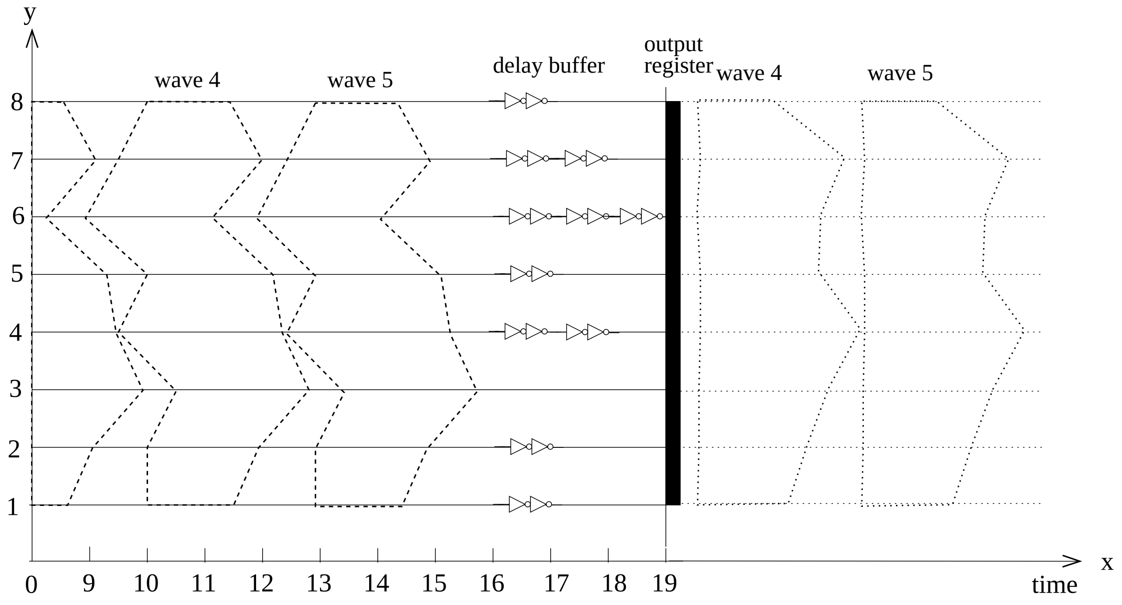


Figure 5.16: Proposed new wave pipeline-based architecture output delay buffers

a polygon with left side lined up. However, the clock cycle for the whole circuit is intact. The logic datawaves can propagate normally at the output delay buffers at the original clock cycle set for the the previous part of the new proposed wave pipelined circuit.

Since the GMP becomes a polygon with its left side lined up to a straight line, apparently the clock cycle for the part of output delay buffers and the output registers is equal to the longest path delay bounded interval within the polygon, i.e., $max(d_{j_{right}} - d_{j_{left}})$. Recall that the clock cycle set for the proposed new wave pipeline circuit is $D_{max} - D_{min} - min[\delta(x)]$, where D_{max} and D_{min} are the longest path delay and minimum path delay for all paths, i.e., $D_{max} = d_{right_{max}}$ and $D_{min} = d_{left_{min}}$, $min[\delta(x)]$ is the minimum value of the distance function $\delta(x)$ as defined before.

Two adjacent GMP always have the attaching point with each other on the path has the maximum path delay bounded interval. It means that during the process of propagation, two

GMPs are actually temporally separated by the amount of time equal to the maximum path delay bounded interval within the GMP. Thus, the clock cycle defined for the proposed new wave pipeline-based circuit is equal to the maximum path delay bounded interval within the GMP, i.e., $D_{max} - D_{min} - min[\delta(x)] = max(d_{j_{right}} - d_{j_{left}})$. Therefore, it is evident that the maximum clock frequency achieved in the proposed new wave pipeline circuit is not modified by the attached output delay buffers.

If the output delay buffers are applied to conventional wave pipelined circuit, the shape of the polygons are changed to be less skewed as they are in the proposed new wave pipeline. Thus, two adjacent polygons are further temporally separated after they pass through the output delay buffers as shown in Figure 5.17. However, the clock cycle time is still $D_{max} - D_{min}$ as it is in the conventional wave pipeline without the attached output delay buffers. Because the attached output delay buffers changed the shapes of the polygons, but not the relative distance between datawaves as the buffers add the same amount of delays to every path. Since the logic datawaves are not mixed with each other originally in conventional wave pipeline, the proposed output buffers are not necessary for conventional wave pipelined circuit.

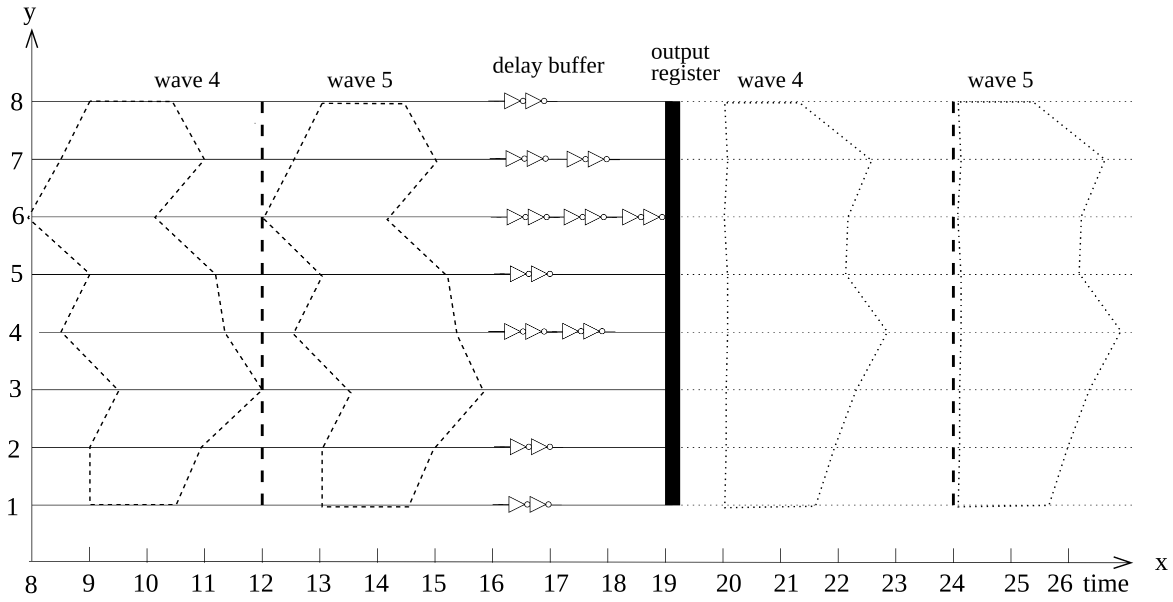


Figure 5.17: Applying the output buffer to the proposed new wave pipeline

5.4 The Simulation and Verification of the proposed new wave pipeline

The proposed bounded system delay model and the new wave pipeline-based architecture are tested on the ISCAS benchmark circuits [83]. In these tests, the circuit paths are extracted out from the benchmark circuit netlist specification. Then, the delay information of the wires and gates, which are stored in the file in a standard delay format (sdf), are applied to those paths. As the delay information specified in this benchmark circuit are all worst case delay, the upper bound of the bounded path delay can be directly obtained by adding up the gate delays through the longest port for every gate (gates may have different input/output ports, and the gate delay through different ports are different) and all the wire delays along the path. The lower bound of bounded path delay is obtained by adding up the gate delays through the shortest port for every gate and all the wire delays along the path. After this step, the GMP can be readily computed by using the proposed bounded system delay model.

Because standard CMOS has a strong data dependency on input patterns, rising and falling delays for the same CMOS gate are generally significantly different. In fact, the delay of a gate can easily vary by a factor of two due to the different input patterns [82]. While for some other techniques, like pseudo-NMOS, where the parallel pullup devices are replaced by a single device whose gate is connected to a bias voltage [76], the effect of data dependency on gate delay in pseudo-NMOS can be negligible. Thus, our experiment also includes the cases which consider the rising and falling delay separately. In these cases, the upper bound of the bounded path delay can be still obtained by adding up the rising or falling gate delays respectively through the longest port for every gate and all the wire delays along the path; while the lower bound can be obtained through the same method except the gate delay on the shortest port is used, while the gate delay on the shortest port is associated with an uncertainty parameter, and in this work, the parameter is assumed to be 10%.

5.4.1 Simulation Algorithm

The algorithm of the simulation program implementation is described as follow:

1. Input: A delay file in the standard delay format.

2. Output: All the paths extracted from the circuit netlist specification with their maximum and minimum path delays. Precise value for the clock cycle for conventional pipeline design, conventional wave pipeline design and the proposed new wave pipeline design.

3. Initialize: Read in the wire interconnection information, and the wire delay information. Read in the gate delay information for gate i; find out the maximum and minimum gate delay (max/min(i)), regardless of rising or falling, and regardless of which port; find out the maximum and minimum rising delay, and the maximum and minimum falling gate delay (max/min riseing/falling (i)), regardless of which port.

4. Bounded system delay model setup:

   (a) Paths extraction: for all the input node j, find all its next level nodes w, add node delay of j to the path delay; if w is the output node, add the node delay of w and the interconnect delay of $j \to w$ to the path delay, return the path $j \to w$. Otherwise, recursively call the extraction algorithm for all the nodes w with the current path delay.

(b) Calculate the bounded path delay: for all the nodes and interconnects along the path, add up their delays in three different cases:

   - Case 1: max/min(i) are the upper bound and lower bound of the gate delay for gate i.
   - Case 2: max(i) rising is the upper bound and (min(i) rising *(1-uncertainty parameter) is the lower bound of the gate delay for gate i.
   - Case 3: max(i) falling is the upper bound and (min(i) falling *(1-uncertainty parameter) is the lower bound of the gate delay for gate i.

5. Clock cycle time calculation for these three different pipeline design:

   (a) For conventional pipeline: $Clock\ Cycle = D_{max}$, where D_{max} is the maximum upper bound of all the paths.

   (b) For conventional wave pipeline: $Clock\ Cycle = D_{max} - D_{min}$, where D_{max} is the maximum upper bound of all paths, and D_{min} is the minimum lower bound of all the paths.

   (c) For the proposed new wave pipeline: $Clock\ Cycle = max(d_{i_{max}} - d_{i_{min}})$, where $d_{i_{max}} - d_{i_{min}}$ is the bounded delay interval for path i.

5.4.2 Experimental Results

The experiments are implemented against 8 ISCAS benchmark circuits, namely $c432$, $c499$, $c880$, $c1355$, $c1980$, $c2670$, $c3540$, and $c5315$. The general statistics of each benchmark circuit are calculated by the simulation program and listed in the Table 5.1 as follow: the sizes of these circuits vary significantly with each other; for example, the circuit $c432$ has only 36 inputs, 7 outputs, 343 interconnects, 160 nodes and 83926 paths, while the circuit $c5315$ has 178 inputs, 123 outputs, 2307 nodes and 1341305 paths; nevertheless, the proposed new wave pipeline works very well on all these benchmark circuits no matter its circuit size is small or big.

Table 5.1: Basic information of benchmark circuits

Circuit Name	# of Input	# of Output	# of Interconnect	# of Node	# of Path
$c432$	36	7	343	160	83926
$c499$	41	32	440	202	9440
$c880$	60	26	755	383	8642
$c1355$	41	32	1096	546	4173216
$c1980$	33	25	1523	880	729057
$c2670$	233	140	2292	1269	679960
$c3540$	50	22	2961	1669	28676671
$c5315$	178	123	4509	2307	1341305

All the paths of each circuit are extracted from the circuit, and are associated with the maximum and minimum path delay value to form the GMP. Three different scenarios are employed by using the maximum and minimum path delay value when only rising delays are considered , using the maximum and minimum path delay when only falling delays are considered, and using the maximum and minimum path delay when both rising and falling delays are considered. The GMPs are shown partially here since the number of paths is too big. Figure 5.18 shows the GMP of the first 100 paths for the circuit $c432$ where both the rising and falling delay are considered in the bounded delay system. Figure 5.19 and Figure 5.20 show part of the GMP for the first 100 paths of the circuit $c432$ if only the rising delay or falling delay is considered respectively. Likewise, Figure 5.21 shown the partial GMP for circuit $c2670$ with considering both rising and falling delay in the bounded delay system. Figure 5.22 and 5.23 shown the partial GMP for circuit $c2670$ if only include rising or falling in the bounded delay system respectively. As shown in all these partial GMPs, no matter in the case of including both rising and falling delay, or rising or falling delay only, for each path, if the maximum path delay is relatively bigger than other maximum path delays, its minimum path delay is also relatively bigger than other minimum path delays. The real GMPs situations displayed here are always lie in between the best case, where all pairs of pieces of lines in the GMP are parallel to each other, besides, the distance between all the pairs of pieces of lines are same, and the worst case, where the GMP has the biggest and smallest delay bound on the same path. Thus, the bounded delay interval for each path, even the maximum one, is always significantly less than the interval between the maximum upper bound of all paths and the minimum lower bound of all the paths. Essentially, this is why the clock cycle time in the proposed new wave pipeline is theoretically better than the clock cycle time in the conventional wave pipeline.

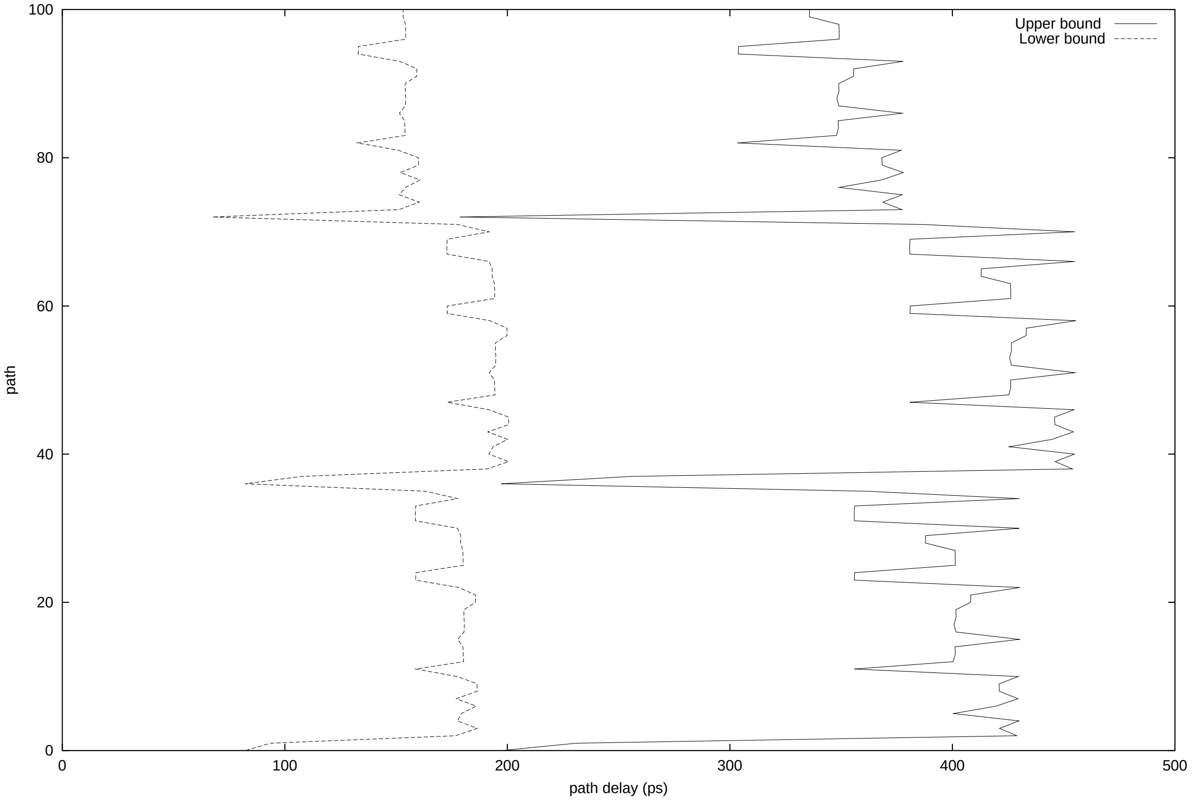


Figure 5.18: GMP for the first 100 paths of ISCAS c432 circuit

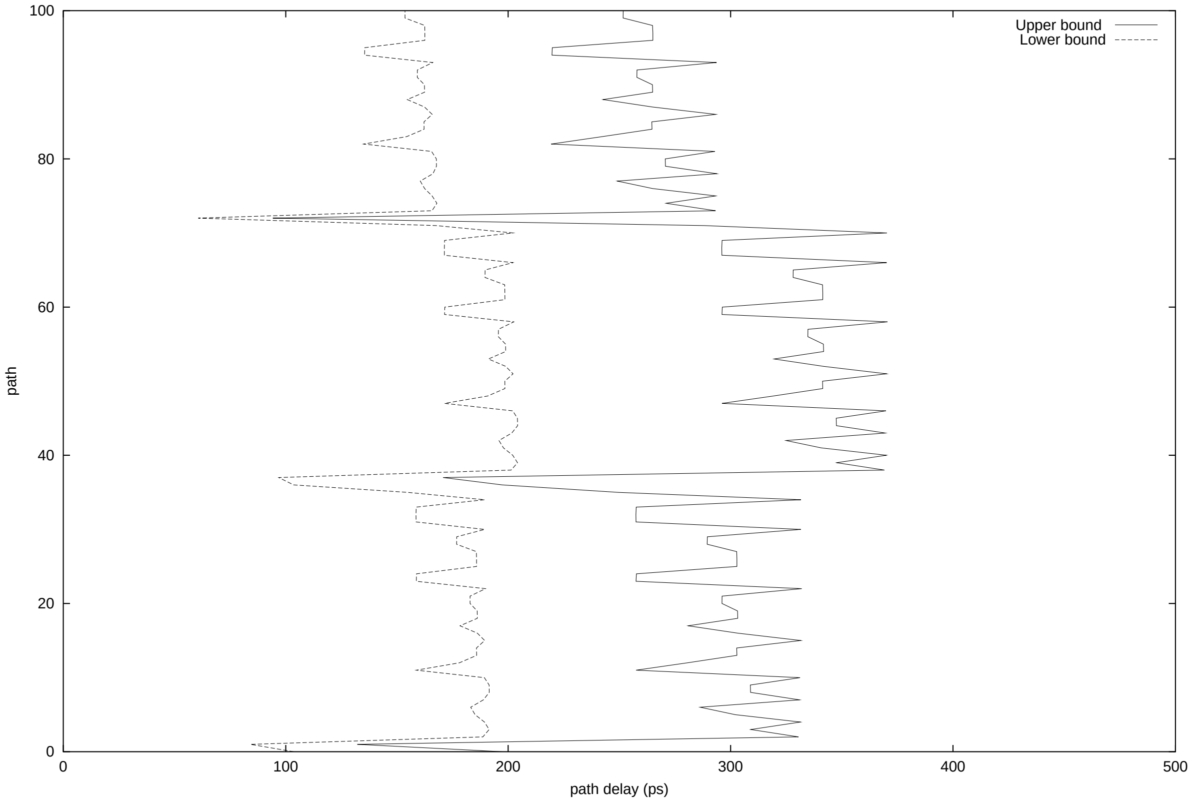


Figure 5.19: GMP for the first 100 paths of ISCAS c432 circuit considering only rising delay with an assumption of 10% delay uncertainty

The valid clock cycle times (CCT) for these ISCAS benchmark circuits are calculated based on conventional pipeline design, conventional wave pipeline design and the proposed new wave pipeline design respectively. Three path delay computing techniques used in this experiments are taking both falling and rising delay into account, considering falling delay only, and considering rising delay only. These three are all implemented for the three pipeline designs.

Table 5.2 and Figure 5.24 show the result of CCT in these three different pipeline designs where both rising and falling delay are included. In Table 5.2, $CCT1$ is the CCT for conventional pipeline design, $CCT2$ is the CCT for conventional wave pipeline design, and $CCT3$ is the CCT for the proposed new wave pipeline design. The experiment results show that the

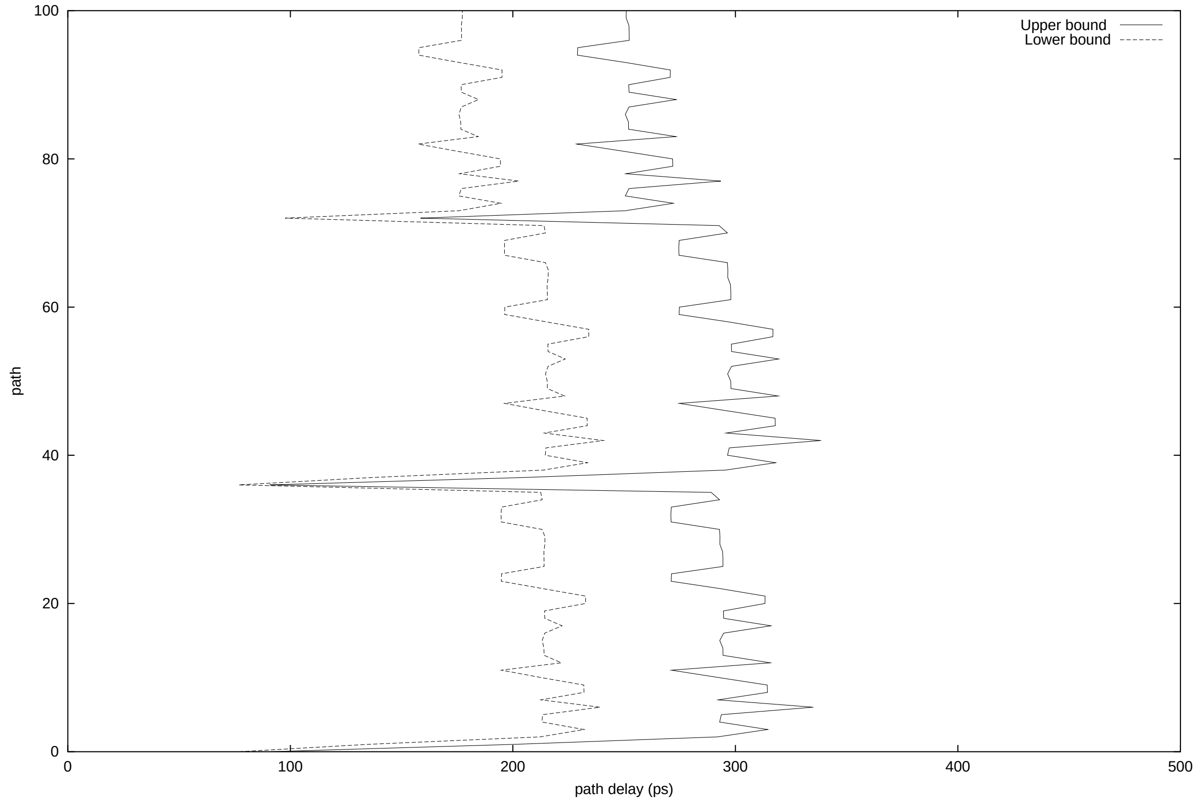


Figure 5.20: GMP for the first 100 paths of ISCAS c432 circuit considering only falling delay with an assumption of 10% delay uncertainty

proposed new wave pipeline can achieve a faster CCT than both the conventional pipeline and the conventional wave pipeline in the same circuit and delay situation. It is 48.63% faster than the CCT in wave pipeline design for circuit $c432$, and 120% faster than the CCT in conventional wave pipeline for circuit $c5315$. Notably as shown in the experiments, the clock cycles in conventional wave pipeline are not significantly shorter than the clock cycles in conventional pipeline design for all these benchmark circuits. It is because that the merit of conventional wave pipeline heavily relies on path delay tuning, which is a step to adjust the path delay before turning the circuit into a conventional wave pipeline circuit. Path delay tuning is basically inserting delay buffers into the path which has shorter path delay. Thus, D_{min} (minimum lower

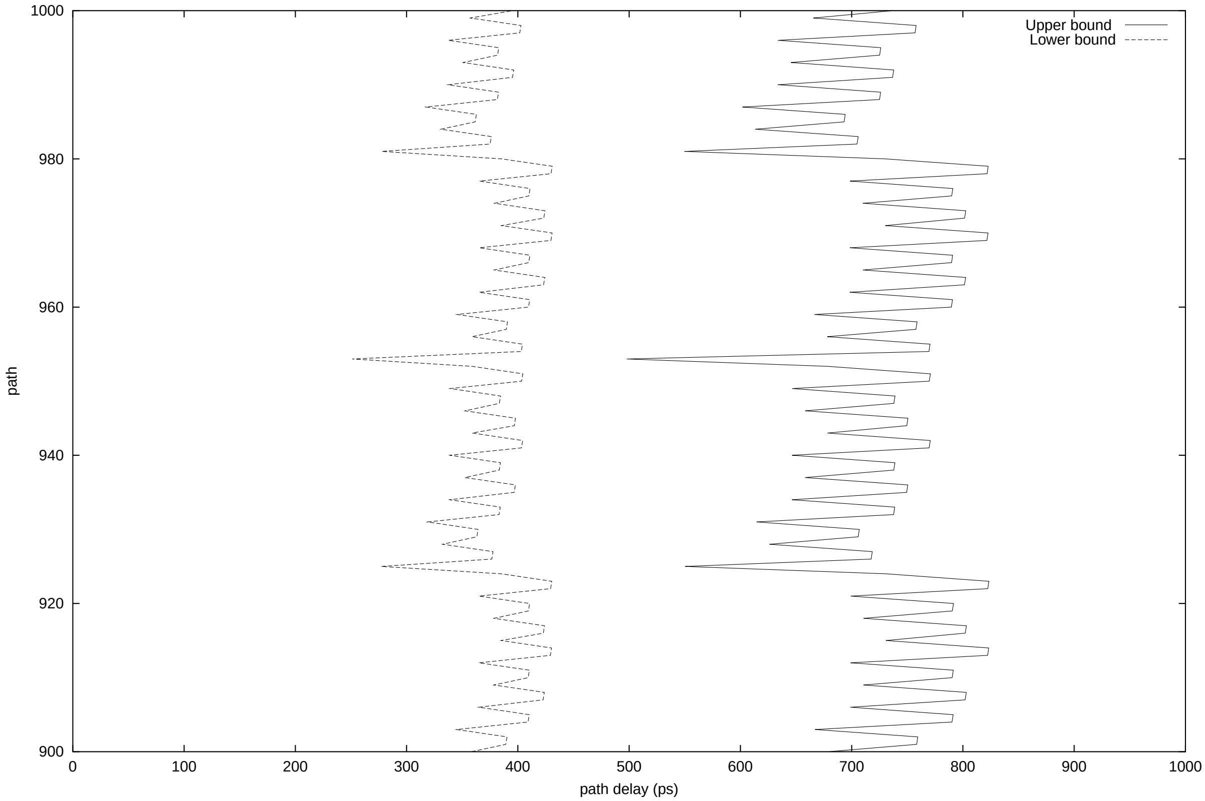


Figure 5.21: GMP for the first 100 paths of ISCAS c2670 circuit

bound of all the paths) can be increased, and the CCT of the conventional wave pipeline circuit, which is $D_{max} - D_{min}$, can be minimized. However, path delay tuning could easily become an extremely big overhead because the number of paths in a circuit is usually huge. In fact, that is one of the major factors hindering the prevalence of wave pipeline. This leads to the discovery of one of the beauties of the proposed new wave pipeline design: there is no need for path delay tuning. The circuit can remain unchanged to implement the proposed new wave pipeline. It can achieve a better CCT without a big overhead with focus on the maximum and minimum delay for each path instead of the maximum and minimum delay for the entire circuit.

Table 5.3 and Figure 5.25 show the result of CCT in these three different pipeline designs where only rising delay is included. The proposed new wave pipeline design achieves a faster

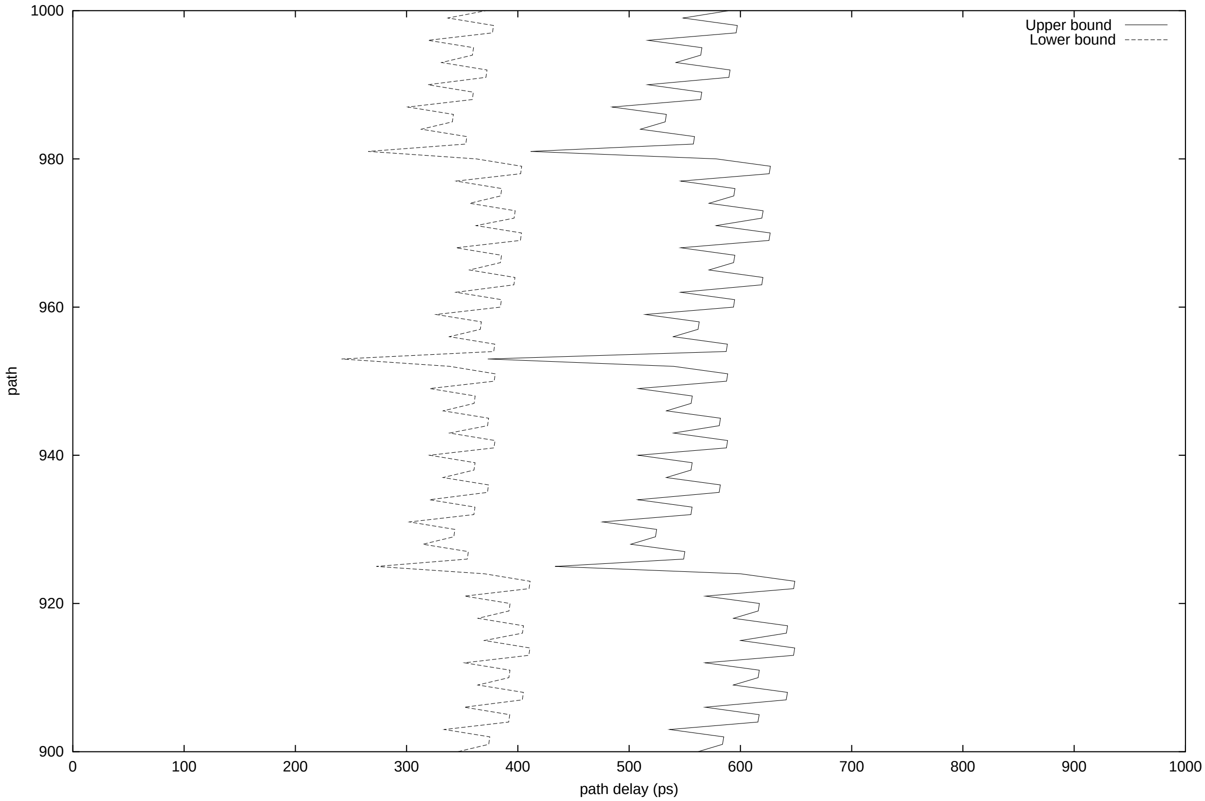


Figure 5.22: GMP for the first 100 paths of ISCAS c2670 circuit considering only rising delay with an assumption of 10% delay uncertainty

Table 5.2: Comparison of clock cycle times of the benchmark circuits

Circuit Name	$CCT1(ps)$	$CCT2(ps)$	$CCT3(ps)$	Increment
$c432$	753	709	477	48%
$c499$	480	460	263	74%
$c880$	791	764	376	103%
$c1355$	826	779	341	128%
$c1980$	1201	1160	600	93%
$c2670$	1047	1041	517	101%
$c3540$	1412	1385	693	99%
$c5315$	1443	1437	653	120%

CCT than the conventional pipeline and the wave pipeline in the same circuit and delay situation. Also, as shown in the Table 5.3 and Figure 5.25, all three pipeline designs have better CCT compared with the situation where both rising and falling delay are included. Besides, the performance increments by the proposed new wave pipeline design are higher in this situation.

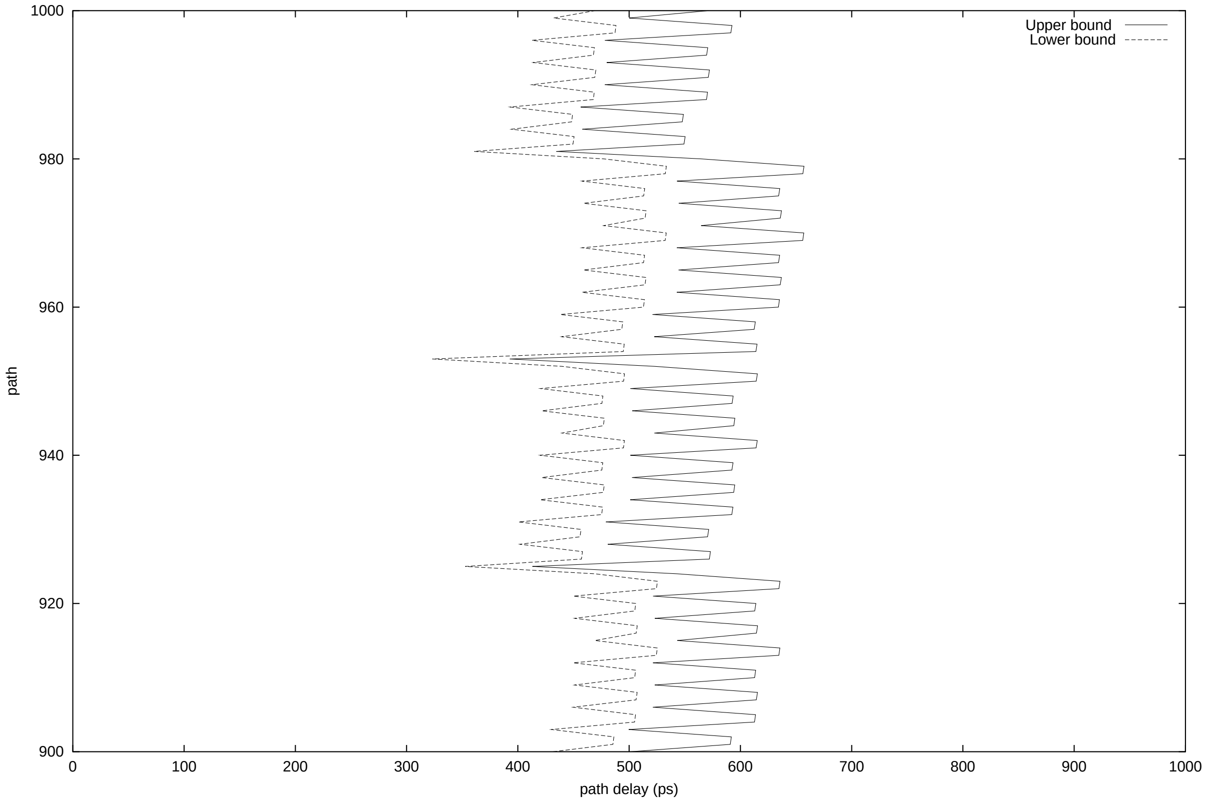


Figure 5.23: GMP for the first 100 paths of ISCAS c2670 circuit considering only falling delay with an assumption of 10% delay uncertainty

Specifically, for all these benchmark circuits, the performance increments vary from 89% to 188%, which is higher than 48% to 128% in the previous situation where both rising and falling delay are taken into account.

Table 5.3: Comparison of clock cycle times of the benchmark circuits if only considering rising delay

Circuit Name	$CCT1(ps)$	$CCT2(ps)$	$CCT3(ps)$	Increment
$c432$	485	442	207	113%
$c499$	427	409	216	89%
$c880$	688	664	276	140%
$c1355$	759	717	307	133%
$c1980$	908	867	345	151%
$c2670$	753	747	259	188%
$c3540$	1093	1068	400	167%
$c5315$	1185	1179	466	153%

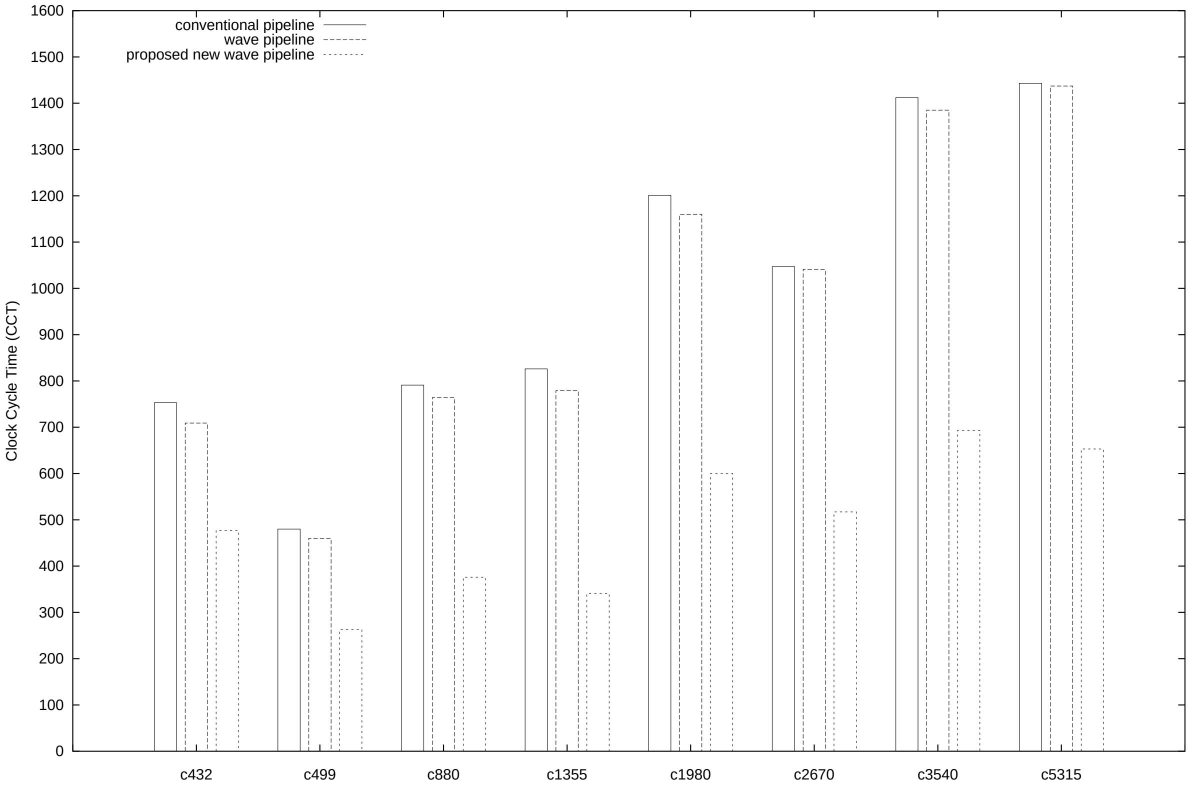


Figure 5.24: Valid clock cycle time if considering both falling and rising delay

Table 5.4 and Figure 5.26 show the result of CCT in three different pipeline designs where only falling delays are considered. In this situation, the proposed new wave pipeline design achieves a much faster CCT than the conventional pipeline and the conventional wave pipeline in the same circuit and delay situation, and a much faster CCT compared with the CCT achieved in the situation considering both rising and falling delay or considering rising delay only. This is because that generally the falling delay of the gate is relatively shorter than rising delay of the gate. Moreover, the falling bounded delay intervals for the gates are usually also shorter than the rising bounded delay intervals or falling and rising bounded delay intervals for the gates. For all the benchmark circuits, the performance increments in this situation are extremely high and vary from 164% to 580%.

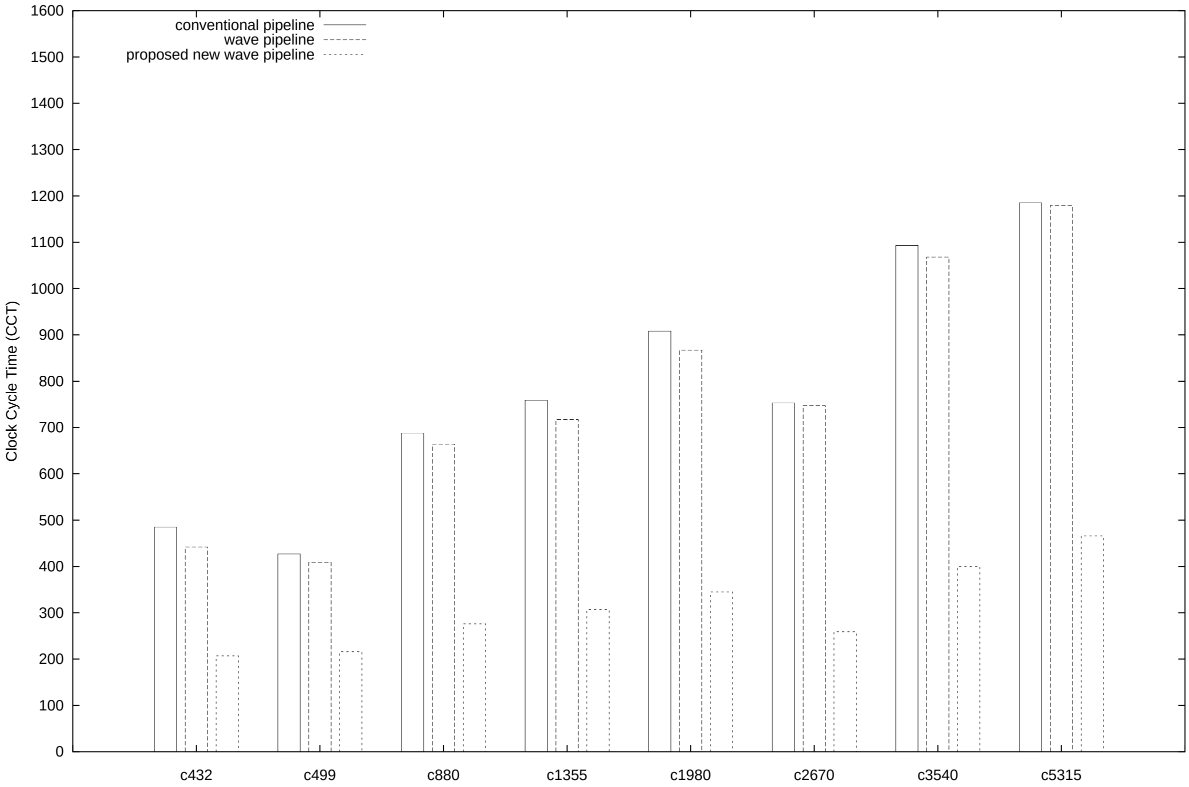


Figure 5.25: Valid clock cycle time if considering rising delay only

Table 5.4: Comparison of clock cycle times of the benchmark circuits if only considering falling delay

Circuit Name	$CCT1(ps)$	$CCT2(ps)$	$CCT3(ps)$	Increment
$c432$	614	575	217	164%
$c499$	296	277	72	284%
$c880$	654	616	147	319%
$c1355$	589	532	94	465%
$c1980$	935	896	141	535%
$c2670$	883	865	204	324%
$c3540$	1166	1128	248	354%
$c5315$	1128	1110	163	580%

5.5 Conclusion

This chapter has proposed the new wave pipeline design for combinational circuits. It is designed based on the the maximum and minimum path delay for each single path in the circuit. Whereas conventional wave pipeline is only focusing on the maximum and minimum path de-

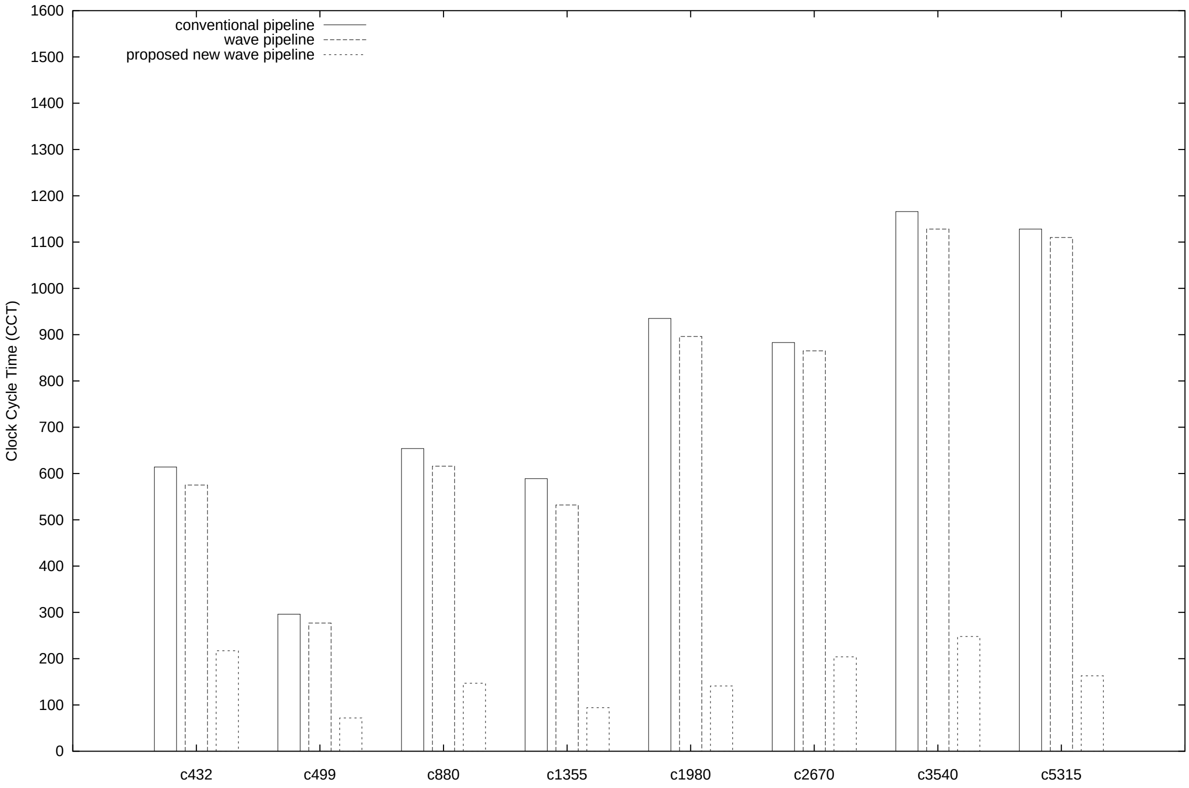


Figure 5.26: Valid clock cycle time if considering falling delay only

lay for all the paths in the circuit. By this intrinsic advantage over conventional wave pipeline, the proposed new wave pipeline can achieve a better CCT than conventional wave pipeline.

Today’s high speed circuit and advanced fabrication technology dictates delay uncertainty an extremely important issue in circuit design. The bounded delay model has been proposed to accommodate the delay uncertainty. The bounded delay model is extended into temporal/spatial bounded system delay model and polygon representation for a more precise manipulation of datawave propagation control. The proposed new wave pipeline is proposed within the bounded system delay model by further locating datawave closer to each other to the limit without violation of datawave mutual exclusion. The CCT calculation for the new proposed wave pipeline circuit is developed and demonstrated, and meanwhile the advantage of the proposed new wave pipeline is also theoretically proved. The worst case and best case of the CCT analysis are illustrated and analyzed. In order to implement the proposed new wave pipeline, a new technique called output delay buffers is proposed to change the GMP back to its original datawave shape, and clearly temporally separate the datawaves again to enable the sampling at the output area. The algorithm of how to attach the output delay buffers to the end of the paths are also discussed.

The proposed new wave pipeline is simulated and verified against a few ISCAS benchmark circuits. Both the simulation algorithm and the experimental results are presented. The results show that the new proposed wave pipeline can achieve at least a 48% performance enhancement on clock cycle time compared with the conventional wave pipeline.

CHAPTER 6

MODELING AND ANALYSIS ON MTTF AND RELIABILITY

The clock cycle time, MTTF (mean time to faulure) and reliability function of the proposed new wave pipeline is analyzed in this Chapter. They are also compared with the clock cycle time, MTTF and the reliability function of conventional pipeline and conventional wave pipeline. Also, the MTTF and reliability functions for these three pipeline designs are simulated and tested on ISCAS benchmark circuits.

Suppose given a combinational circuit, the clock cycle time is determined heavily on how deeply the circuit is pipelined if it is designed as conventional pipeline circuit. The more stages incorporated, the less gates will be included in each pipeline stage ,and therefore, the smaller the clock cycle will be. Nevertheless, the circuit can not be divided into extremely small stages and the number of stages can not be infinite large because the extra cost for the inserted registers or latches could be fairly expensive. Suppose the given combinational circuit is designed into the conventional wave pipeline architecture, the clock cycle time is $D_{max} - D_{min}$, while for the same combinational circuit in the proposed new wave pipeline architecture, the clock cycle time is $D_{max} - D_{min} - \delta_{push}$. These three pipeline design techniques and their valid clock cycle times are demonstrated in the Figure 6.1 as follow.

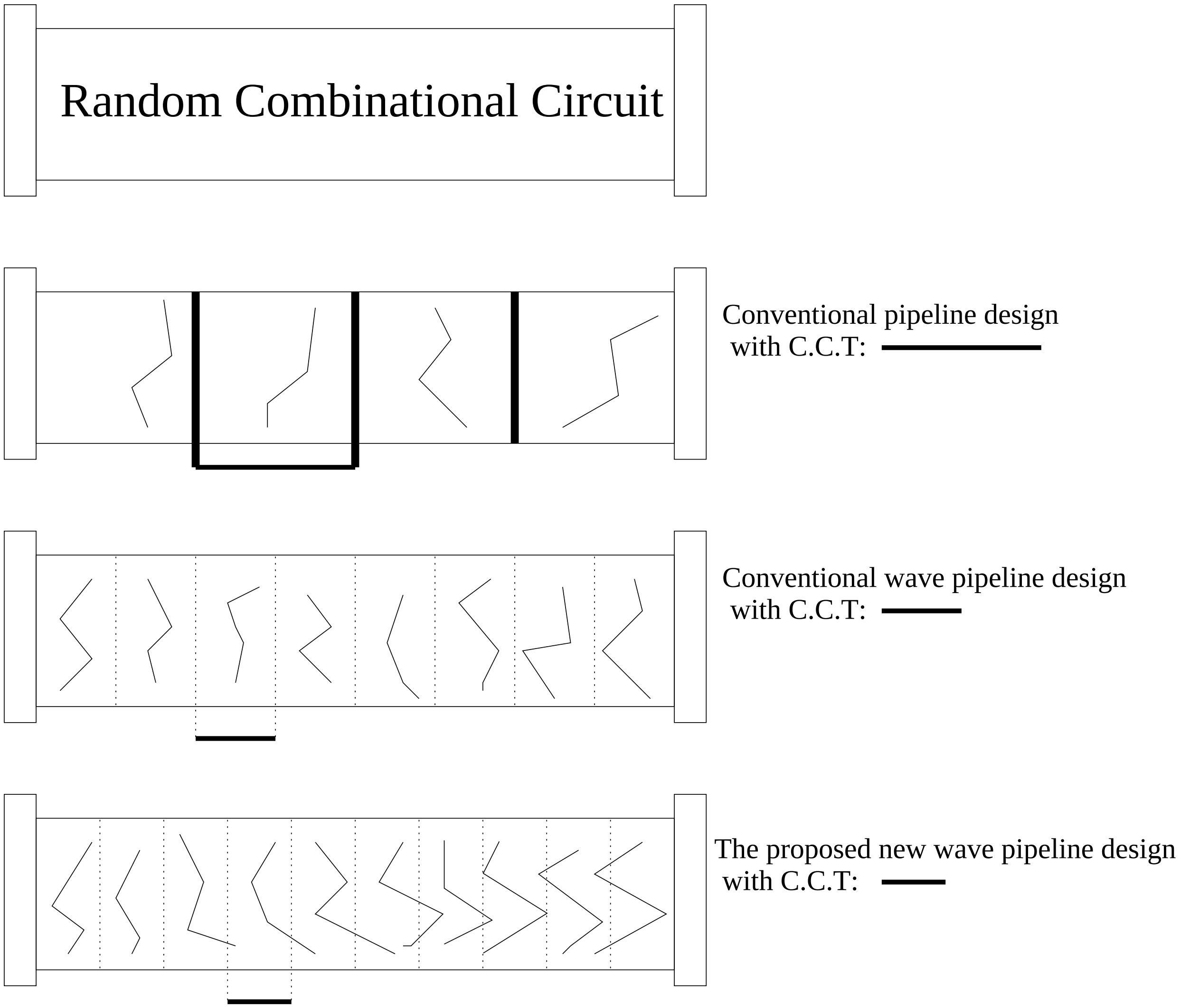

Random Combinational Circuit
Conventional pipeline design
with C.C.T:
Conventional wave pipeline design
with C.C.T:
The proposed new wave pipeline design
with C.C.T:

Figure 6.1: Comparison of clock cycle time for different pipeline designs

6.1 Reliability Model Analysis

The reliability analysis technique introduced in Section 2.1 is employed in this Chapter to model the reliability of the proposed wave pipeline. For a random combination circuit, assume there are n paths $(i.e. x_1, ..., x_n)$, also assume x_i follows Gaussian Distribution with the sample mean $\bar{x}$ and sample standard derivation s. The population mean is μ and the standard variation is σ. Thus, it can be expressed as $x \sim N(\mu\sigma^2)$. Since $\bar{x}$ and s are unbiased estimators of μ and σ, the probability density function the path delays followed is $f_x(x) = \frac{1}{\sqrt{2\pi}s}e^{-\frac{(x-\bar{x})^2}{2s^2}}$ as shown in

Figure 6.2.

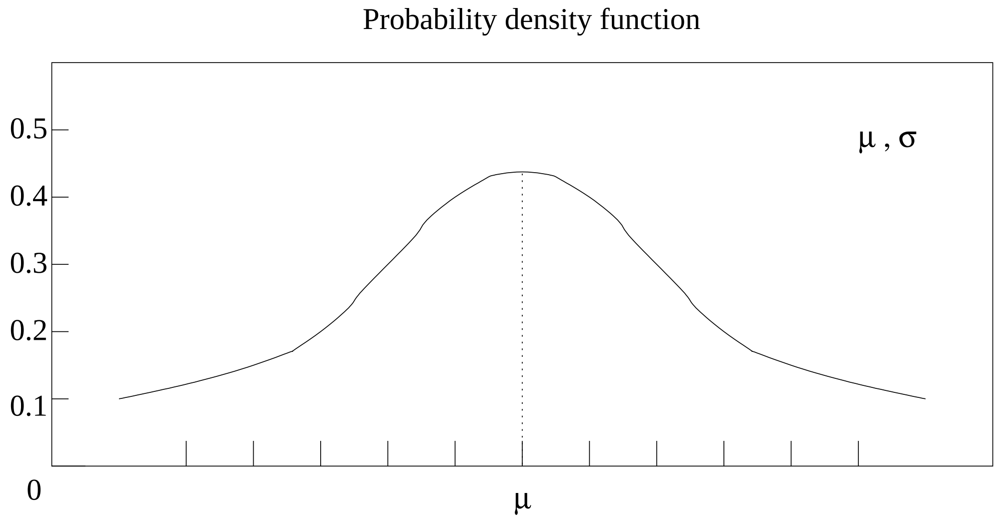


Figure 6.2: Statistical analysis of path delays

The reliability of bits propagation can be modeled as the probability that the bit combines with the valid time interval it associates with. The reliability of bits propagating within a conventional pipeline, wave pipeline, and the proposed new wave pipeline are compared and shown in the Figure 6.3 as follow.

6.2 Mean Time to Failure

The Mean Time to Failure (MTTF) [31] is calculated by finding the expected value of the time of failure. From probability theory, the expected value of a random variable X is

$$E[X] = \int_{-\infty}^{\infty} x f(x) dx \tag{6.1}$$

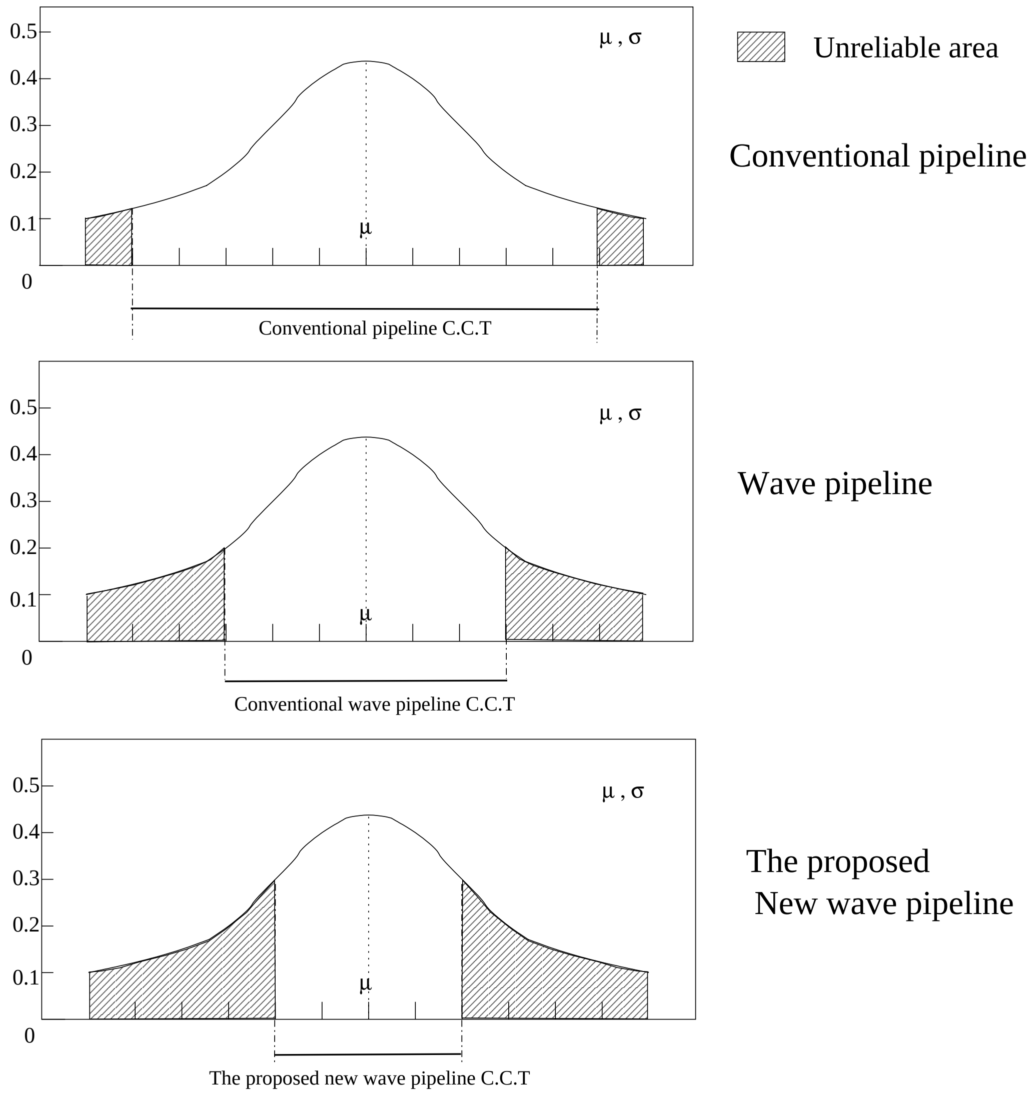


Figure 6.3: Reliability comparison

where $f(x)$ is the probability density function. The random variable is the expected value of the time of failure as follows.

$$MTTF = \int_0^\infty t f(t) dt \tag{6.2}$$

where $f(t)$ is the failure density function. As $f(t)$ is undefined for $t < 0$, thus the integral runs from 0 to ∞. Assuring the path delays follow Gaussian distribution, and also according to the failure density function introduced in 2.1, the failure density function for a circuit with the average path delay μ and clock cycle time $C.C.T$ is:

$$f(t) = 1 - \int_{\mu-\frac{1}{2}(C.C.T)}^{\mu+\frac{1}{2}(C.C.T)} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt \tag{6.3}$$

Thus, given random circuit, the MTTF for conventional pipeline is:

$$MTTF_{pipe} = \int_0^\infty t(1 - \int_{\mu-\frac{1}{2}(D_{max})}^{\mu+\frac{1}{2}(D_{max})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt) dt \tag{6.4}$$

where D_{max} is the longest path delay in all stages.

The MTTF for conventional wave pipeline is:

$$MTTF_{wave} = \int_0^\infty t(1 - \int_{\mu-\frac{1}{2}(D_{max}-D_{min})}^{\mu+\frac{1}{2}(D_{max}-D_{min})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt) dt \tag{6.5}$$

where D_{max} and D_{min} are the longest path delay and the shortest path delay for all the paths in the circuit.

The MTTF for the proposed new wave pipeline is:

$$MTTF_{new} = \int_0^\infty t(1 - \int_{\mu-\frac{1}{2}(D_{max}-D_{min}-\delta_{push})}^{\mu+\frac{1}{2}(D_{max}-D_{min}-\delta_{push})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt) dt \tag{6.6}$$

where where D_{max} and D_{min} are the longest path delay and the shortest path delay, and the δ_{push} is the amount of the exact CCT time the proposed wave pipeline can further push beyond conventional wave pipeline design.

6.3 Reliability Functions

The failure distribution function $F(T)$ is the probability of a bit failing in the time interval $0 \leq t \leq T$.

$$F(T) = \int_0^T f(t)dt \tag{6.7}$$

Where $f(t)$ is the failure density function as defined in the Equation 6.3. Then, the failure distribution function $F(T)$ is:

$$\begin{aligned} F(T) &= \int_0^T f(t)dt \\ &= \int_0^T (1 - \int_{\mu-\frac{1}{2}(C.C.T)}^{\mu+\frac{1}{2}(C.C.T)} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt \end{aligned} \tag{6.8}$$

The reliability function is the probability of a bit which does not fail in the time interval $0 \leq t \leq T$.

$$\begin{aligned} R(T) &= 1 - F(T) \\ &= 1 - \int_0^T (1 - \int_{\mu-\frac{1}{2}(C.C.T)}^{\mu+\frac{1}{2}(C.C.T)} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt \end{aligned} \tag{6.9}$$

Thus, for a given random circuit, the reliability function for conventional pipeline is:

$$R(T)_{pipe} = 1 - \int_0^T (1 - \int_{\mu-\frac{1}{2}(D_{max})}^{\mu+\frac{1}{2}(D_{max})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt \tag{6.10}$$

where D_{max} is the longest path delay.

The reliability function for conventional wave pipeline is:

$$R(T)_{wave} = 1 - \int_0^T (1 - \int_{\mu-\frac{1}{2}(D_{max}-D_{min})}^{\mu+\frac{1}{2}(D_{max}-D_{min})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt \tag{6.11}$$

where D_{max} and D_{min} are the longest path delay and the shortest path delay.

The reliability function for the proposed new wave pipeline is:

$$R(T)_{new} = 1 - \int_0^T (1 - \int_{\mu-\frac{1}{2}(D_{max}-D_{min}-\delta_{push})}^{\mu+\frac{1}{2}(D_{max}-D_{min}-\delta_{push})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt \tag{6.12}$$

where D_{max} and D_{min} are the longest and the shortest path delay, and the δ_{push} is the amount of the exact CCT time the proposed wave pipeline can further push beyond conventional wave pipeline design.

6.4 Experiment and Simulation

The MTTF and reliability functions for these three different pipeline designs are experimented against ISCAS benchmark circuits $c432$, $c499$, $c880$, $c1980$, $c2670$. Their path delays statistical analysis and the circuit reliability calculation based on the reliability model developed in this chapter are conducted and shown in this section. Also, the MTTF and reliability functions specifically for each benchmark circuit are addressed and illustrated.

Table 6.1: Statistical analysis of path delays and reliability of benchmark circuits

Circuit Name	# of paths	Mean	Variance	Area 1	Area 2	Area 3
$c432$	83926	445	4420	$1.47E-8$	$9.42E-8$	$1.74E-4$
$c499$	9440	294	2641	$2.90E-6$	$7.43E-6$	$7.66E-4$
$c880$	8642	415	10591	$1.21E-4$	$2.03E-4$	$3.47E-3$
$c1980$	729057	683	9602	$8.64E-10$	$3.19E-9$	$2.69E-4$
$c2670$	6769960	606	6448	$6.89E-11$	$8.98E-11$	$2.10E-5$

Table 6.1 shows the number of paths, the mean of all the path delays and the variance of all the path delays. Area 1, Area 2, and Area 3 represent the unreliable area for conventional pipeline design, conventional wave pipeline design, and proposed new wave pipeline design respectively, according to Figure 6.3. The result shown in this table matches Figure 6.3 very well. For all benchmark circuits, conventional pipeline and conventional wave pipeline have a relative same level of unreliable area, while still conventional pipeline always has the smaller unreliable area. Both the conventional pipeline and conventional wave pipeline have extremely small unreliable area compared with the new proposed wave pipeline. This is due to the fact that the new proposed wave pipeline achieves a higher CCT by further relaxing the distance limit between datawaves, while at the same time, it losts part of its reliability. Table 6.1 and the CCT comparison conducted among conventional pipeline, conventional wave pipeline and the new proposed wave pipeline clearly manifest the trade-off between speed and reliability.

The MTTF for each benchmark circuit is solved according to Equation 6.4, 6.5, and 6.6. The results for the benchmark circuit $c432$ are shown as follow.

The MTTF of the benchmark circuit $c432$ for conventional pipeline design is:

$$\begin{aligned} MTTF_{c432\ pipe} &= \int_0^\infty t(1 - \int_{\mu-\frac{1}{2}(D_{max})}^{\mu+\frac{1}{2}(D_{max})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt \\ &= \int_0^\infty t \times (1.47E-8)dt \\ &= (1.47E-8) \times \frac{t^2}{2} \end{aligned} \tag{6.13}$$

The MTTF of the benchmark circuit $c432$ for conventional wave pipeline design is:

$$MTTF_{c432\ wave} = \int_0^\infty t(1 - \int_{\mu-\frac{1}{2}(D_{max})}^{\mu+\frac{1}{2}(D_{max})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt$$

$$\begin{aligned} &= \int_0^{\infty} t \times (9.42E-8)dt \\ &= (9.42E-8) \times \frac{t^2}{2} \qquad (6.14) \end{aligned}$$

The MTTF of the benchmark circuit $c432$ for the new proposed wave pipeline is:

$$\begin{aligned} MTTF_{c432\ new} &= \int_0^{\infty} t(1 - \int_{\mu-\frac{1}{2}(D_{max})}^{\mu+\frac{1}{2}(D_{max})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt \\ &= \int_0^{\infty} t \times (1.74E-4)dt \\ &= (1.74E-4) \times \frac{t^2}{2} \qquad (6.15) \end{aligned}$$

The reliability functions for each benchmark circuit is also solved according to conventional pipeline design, conventional wave pipeline design, and the proposed new wave pipeline design based on the Equation 6.10, Equation 6.11, and Equation 6.12. Similarly, the reliability function for the benchmark circuits $c432$ are shown as follow as an example.

The reliability function of the circuit c432 for conventional pipeline is:

$$\begin{aligned} R(T)_{c432\ pipe} &= 1 - \int_0^{T} (1 - \int_{\mu-\frac{1}{2}(D_{max})}^{\mu+\frac{1}{2}(D_{max})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt \\ &= 1 - \int_0^{T} (1.47E-8)dt \\ &= 1 - (1.47E-8) \times T \qquad (6.16) \end{aligned}$$

The reliability function of the circuit c432 for conventional wave pipeline is:

$$\begin{aligned} R(T)_{c432\ wave} &= 1 - \int_0^{T} (1 - \int_{\mu-\frac{1}{2}(D_{max}-D_{min})}^{\mu+\frac{1}{2}(D_{max}-D_{min})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt)dt \\ &= 1 - \int_0^{T} (9.42E-8)dt \end{aligned}$$

$$= \quad 1 - (9.42E - 8) \times T \qquad (6.17)$$

The reliability function of the circuit c432 for the new proposed wave pipeline is:

$$\begin{aligned} R(T)_{c432\ new} &= 1 - \int_0^T (1 - \int_{\mu - \frac{1}{2}(D_{max} - D_{min} - \delta_{push})}^{\mu + \frac{1}{2}(D_{max} - D_{min} - \delta_{push})} \frac{1}{\sqrt{2\pi}\sigma} e^{(-\frac{(t-\mu)^2}{2\sigma^2})} dt) dt \\ &= 1 - \int_0^T (1.74E - 4) dt \\ &= 1 - (1.74E - 4) \times T \qquad (6.18) \end{aligned}$$

The reliability functions for the benchmark circuit $c432$, $c499$, $c880$, $c1980$, and $c2670$ are plotted and shown in the Figure 6.4, Figure 6.5, Figure 6.6, Figure 6.7 and Figure 6.8 as follow. For each benchmark circuit, the reliability of the conventional pipeline is better than the reliability of the conventional wave pipeline, and the reliability of the conventional wave pipeline is better than the reliability of the proposed new wave pipeline as shown in the figures.

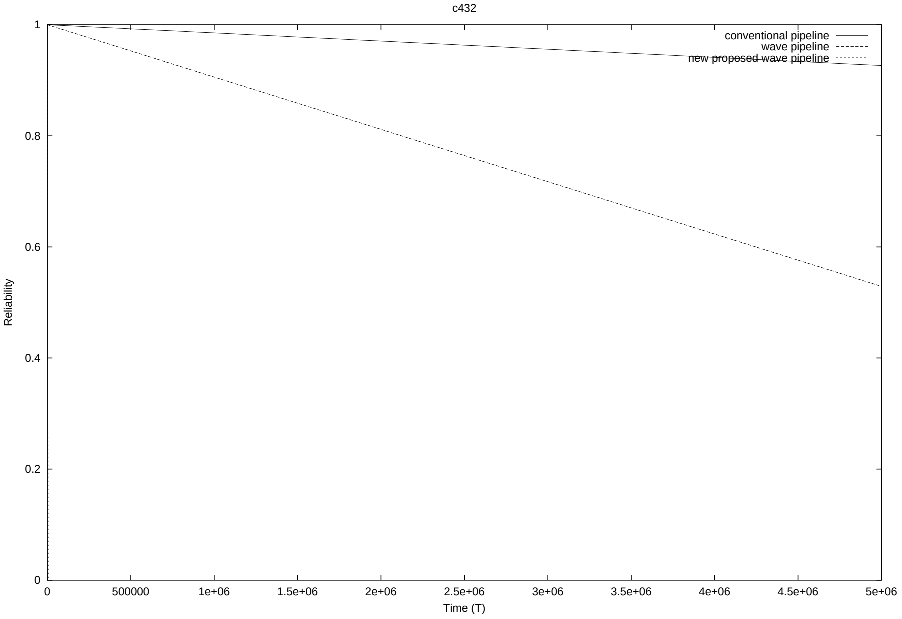


Figure 6.4: Reliability function of the circuit c432

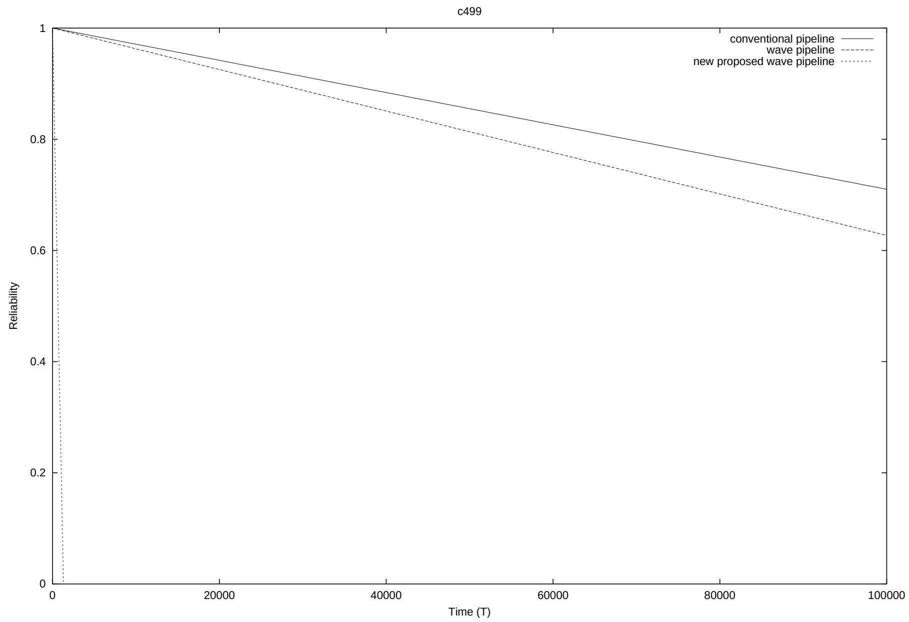


Figure 6.5: Reliability function of the circuit c499

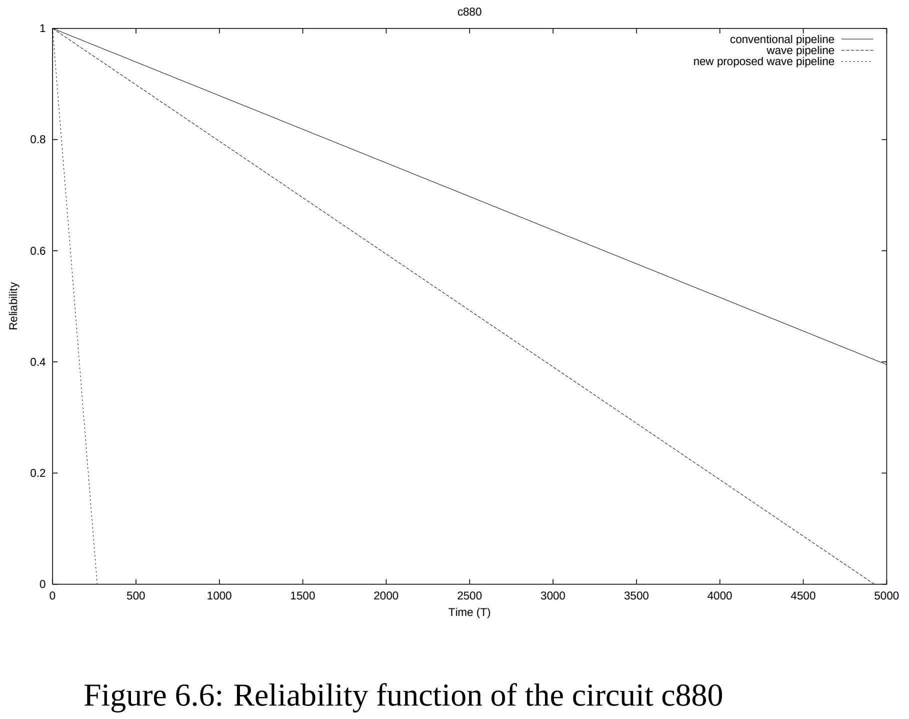


Figure 6.6: Reliability function of the circuit c880

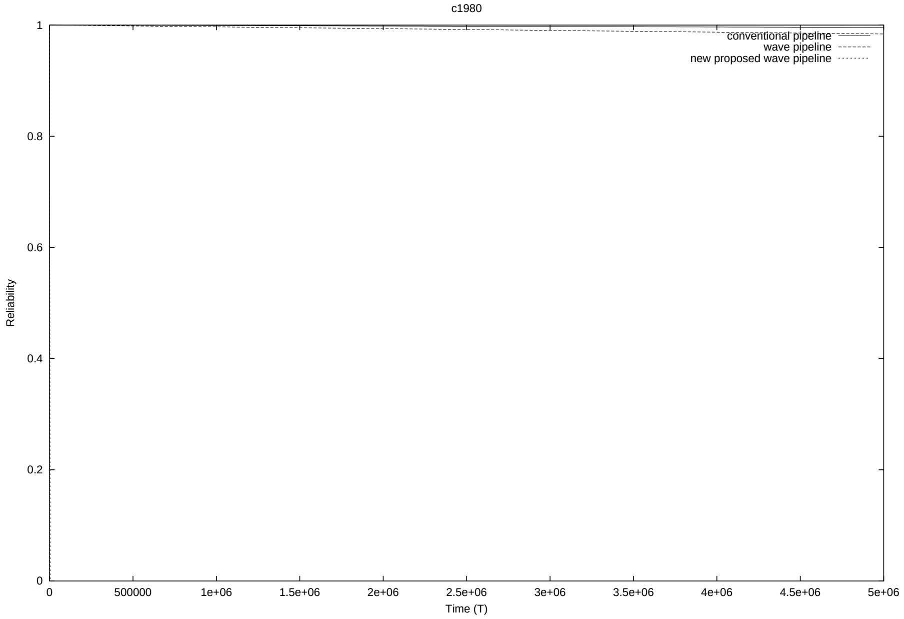


Figure 6.7: Reliability function of the circuit c1980

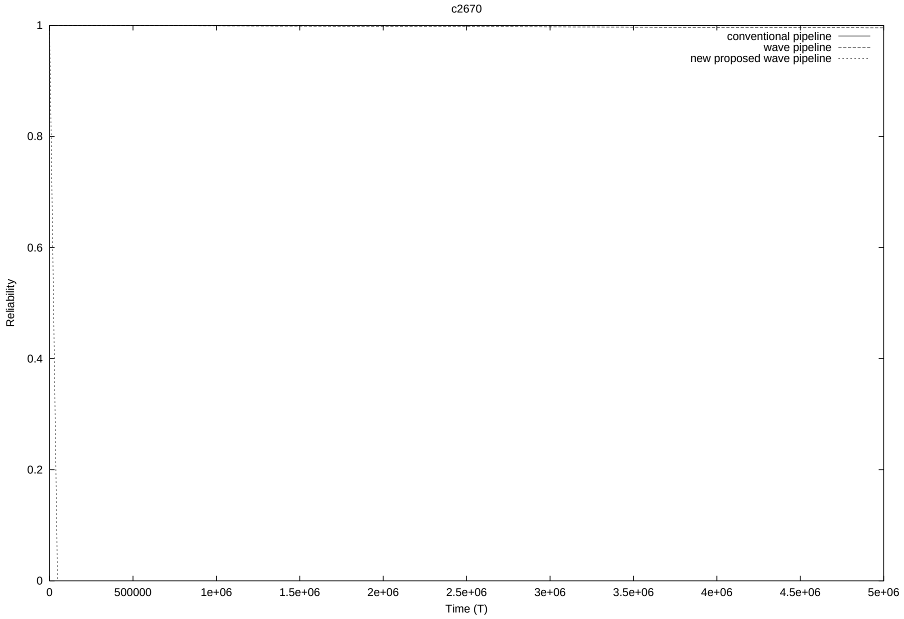


Figure 6.8: Reliability function of the circuit c2670

CHAPTER 7

CONCLUSIONS AND DISCUSSION

This dissertation has presented comprehensive and theoretical methodologies to model, assure, and optimize the reliability of clockless wave pipeline-based combinational logic design. A specific architectural model referenced in this dissertation is the two-phase clockless asynchronous wave pipeline [23]. The intra-wave fault model has been proposed and thoroughly identified as the unique fault of the clockless wave pipeline in comparison with wave and wave delay faults of conventional wave pipelines. The intra-wave fault rate is statistically yet practically modeled, and extensively studied with respect to various design parameters, such as Δ (data skew), L (request level length) and α ($\frac{L-\Delta}{2}$).

In order to achieve a reliable clockless wave pipeline, the fault tolerance for request signals and datawave approaches have been proposed under the fault models. It was demonstrated that a low-level glitch hit on a high request signal line may cause a broken datawave (i.e., intra-wave fault on P-switches) with no fault on N-switches assumed. Also, a high-level glitch hit on a low-request signal line may result in a broken datawave (i.e., intra-wave fault on N-switches) with no fault on P-switches assumed. Based on this principle, an AND gate was used to mask the high-level glitch hit on a low request signal to a P-switch; and an OR gate was used to mask the low-level glitch hit on a high request signal to a N-switch. This simple yet effective approach can mask all the glitches on request signal lines except when every redundant request signal line is impaired by a glitch at the same time. It has been demonstrated that the reliability of the clockless wave pipeline is increased exponentially as more redundant request signal lines are incorporated. Also, the reliability with respect to MTTG (Mean Time To Glitch), λ (glitch rate), and L (request signal level length) have been analyzed.

The fault tolerant design for datawave, referred to as the bipolar switch-based clockless wave pipeline in association with affiliate request signals, has also been proposed. It was demonstrated that the proposed bipolar switch-based 2-phase clockless wave pipeline with redundant affiliate request signals cause a reduction on the data skew along the data paths. As the intra-wave fault is caused by the data skew on different data paths, the bipolar switches in association with the affiliated request signals can effectively slow down the propagation speed of the faster data bits without affecting the propagation of the slower data bits. Therefore, the data skew rate has been reduced and consequently the intra-wave fault rate is effectively reduced as well. Experiments and simulations have demonstrated and verified the efficiency and effectiveness of the proposed reliability modeling and assurance methods and fault-tolerant design for reliability.

A new temporal/spatial system bounded delay model has been proposed and it expands the current bounded path delay into the system level. By extracting a complete set of paths from the circuit and relocating them into a 2-dimensional coordinate system, with the y-axis representing the path delays and the x-axis representing the order of the paths, it dynamically visualizes the datawave propagation situation at the system level. The new wave pipeline technique has been theoretically yet practically demonstrated to achieve the maximum circuit speed. Adjacent datawaves can be pushed closer together in the new wave pipeline than in the original wave pipeline by allowing the datawave-superimposition, while strict mutual exclusion is still assured and guaranteed during the datawave propagation process. The new clock cycle time achieved in the new wave pipeline is the maximum bounded path delay interval, i.e., $D_{max} - D_{min} - min[\delta(x)] = max(d_{j_{right}} - d_{j_{left}})$, where D_{max} and D_{min} are the longest path delay and minimum path delay for all paths, $min[\delta(x)]$ is the minimum value of the distance function $\delta(x)$, where $\delta(x)$ is the distance function of those two piecewise functions which represents the left and right border of the GMP, and $max(d_{j_{right}} - d_{j_{left}})$ is the longest path delay bounded interval within the GMP. Extensive simulations have been carried out on the ISCAS benchmark circuits, and the results have shown that the clock cycle time in the proposed new wave pipeline is significantly shorter than the one in conventional wave pipeline design.

Furthermore, the MTTF and reliability function of the proposed new wave pipeline has been analyzed. They have been compared with the MTTF and the reliability function of conventional pipeline and conventional wave pipeline. The MTTF and reliability functions for these three pipeline designs have been simulated and tested against ISCAS benchmark circuits. The results have shown that, the proposed new wave pipeline requires a strict and tight assurance of reliability as a cost for achieving high speed circuit.

REFERENCES

[1] S. Anderson, J.Earle, R. Goldschmidt, and D.Powers, ”The IBM system/360 model 91: floating point execution unit”, *IBM J.Res.Develop.*, vol.11, no.1, pp24-53, Jan. 1967.

[2] R. Anglada, and A. Rubio, ”An approach to crosstalk effect analysis and avoidance techniques in digital CMOS VLSI circuits”, *Int. J. of Electronics*, 65(1), 1988.

[3] D.W. Bailey, B.J. Benschneider, ”Clocking design and analysis for a 600-MHz Alpha microprocessor”, *IEEE Journal of Solid-State Circuits*, Vol.33 Iss.11 pp.1627-1633, 1998.

[4] W.P. Burleson, M. Ciesielski, F. Klass, W. Liu, ”Wave-Pipelining: A Tutorial and Research Survey”, *IEEE Trans. on VLSI system*, VOL.6, NO.3, September, 1998.

[5] Chuan-Hua Chang, ”Performance Optimization of Pipeline Circuits With Latches and Wave Pipelining”, *Phd Theiss, Department of computer Science and Engineering, University of Michigan*, 1996.

[6] J.F.Chappel and S.G.Zaky, ”EMI Effects and Timing Design for Increased Reliability of Digital Systems”, *IEEE Transactions on Circuits and Systmes: Fundamental Theory and Applicaitons*, Vol.44, No.2, pp.130-142, February 1997.

[7] L. W. Cotton, ”Maximum-rate pipeline system”, *AFIPS Proceedings Spring Joint Computer Conference*, vol. 34, Montvale, NJ: AFIPS Press, pp.581-586, May 1969.

[8] Alfred L. Crouch, ”Exploring the Basics of AC Scan”, *Evaluation Engineering*, July 2004.

[9] A.E. Dooply, K.Y Yun, ”Optimal clocking and enhanced testability for high-performance self-resetting domino pipelines”, *IEEE Anniversary Conference on Advanced Research in VLSI*, pp.200-214, 1999.

[10] B.C. Ekroot, ”Optimization of pipelined processors by inserting of combinational logic delay”, *Ph.D. Dissertation, EE.Dept., Stanford U.*, Sep. 1987.

[11] K.M. Fant, S.A. Brandt, ”NULL Convention LogicTM: a complete and consistent logic for asynchronous digital circuit synthesis”, *IEEE Proceedings of International Conference on ASAP*, pp.261-273, April. 1996.

[12] M. Favalli, and C. Metra, ”Sensing circuit for on-line detection of delay faults”, *Very large Scale Integration (VLSI) Systems, IEEE Transaction on* Volume: 4(1), pp. 130-133, March 1996.

[13] T. Feng, N.Park, M. Choi, ”Yield Modeling and Analysis of A Clockless Asynchronous Wave Pipeline with Pulse Faults”, *IEEE Defect and Fault Tolerance in VLSI Systems*, Nov. 2003.

[14] T. Feng, B. Jin, N.Park and F. Lombardi, ”Yield Optimization of Clockless Wave Pipeline with Intra/Inter-Wave Faults”, to appear in *IEEE IMTC’04*, May 2004.

[15] T. Feng, B.Jin, J.Wang, N.Park, Y.B. Kim and F. Lombardi, ”Fault Tolerant Clockless Wave Pipeline Design”, *ACM CF’04*, Apr. 2004.

[16] T. Feng, N. Park, Y. Kim, F. Lombardi, F. Meyer, ”Reliability Modeling and Assurance of Clockless Wave Pipeline”, *IEEE DFT* 2004.

[17] J. Fishburn, *Clock Skew Optimization*, IEEE Trans.Comput., 1990.

[18] T. Gray, T. Hughes, S. Arora, W. Liu, R. Cavin, ”Theoretical and Practical Issues in CMOS Wave Pipelining”, *VLSI Design*, pp, 9.2.1-9.2.5,1991.

[19] C. Gray, W.T. Liu, K.R. Cavin, ”Timing Constraints for Wave-Pipelined Systems” *IEEE Transactions on computer-aided design of integrated circuits and systems*, vol., 13, NO, 8, August 1994.

[20] C.T.Gray, W.Liu, and R,K.Cavin, ”Exact timing analysis condidering data depent delays”, *North Carolina State University., Tech.Rep* NCSU-91-04, 1992. Preliminary Version.

[21] D. Harris, M.A. Horowitz, ”Skew-tolerant domino circuits”, *IEEE Journal of Solid-State Circuits*, pp.422-423, 1997.

[22] O. Hauck, A.S. Huss, ”Asynchronous Wave Pipelines for High Throughput Data Paths”, *IEEE Int. Symp. on Electronics, Circuits and Systems*, Vol.1 pp.283-286, 1998.

[23] O. Hauck, M. Garg, A.S. Huss, ”Two-Phase Asynchronous Wave-Pipelines and Their Application to a 2D-DCT”, *IEEE Int. Symp. on Advanced Research in Asynchronous Circuits and Systems*, pp.219-228, 1999.

[24] O. Hauck, A. Katoch, A.S. Huss, ”VLSI system design using asynchronous wave pipelines:a 0.35μ CMOS 1.5 GHz elliptic curve public key cryptosystem chip”, *IEEE Int. Symp. on Advanced Research in Asynchronous Circuits and Systems*, pp.188-197, 2000.

[25] Soha Hassoum, ”Critical Path Analysis Using a Dynamically Bounded Delay Model”, *Annual ACM IEEE Design Automation Conference archive Proceedings of the 37th con-*

ference on Design automation table of contents, Los Angeles, California, United States Pages: 260 - 265, 2000.

[26] S. Hermanns and S.A. Huss, ”Embedding of asynchronous wave pipelines into synchronous data processing”, *SAME Conference 2001*, 14, 15 , Sophia Antipolis, Valbonne,France, November 2001.

[27] R. Heald, K. Shin, V. Reddy, W. Lynch, ”64-Kbyte-Sum-Addressed-memory Cache with $1.6 - ns$ Cycle and $2.6 - ns$ Latency”, *IEEE Journal of Solid-State Circuits*, vol. 33, no. 11, PP. 1682-1689, Nov. 1998.

[28] ME Hoff, Jr. ”IC technology: Trends and impact on digital signal processing” , *Proceedings of the IEEE-ICASSP IEEE*, New York, 1980.

[29] Y.I.Ismail and E.G.Friedman, ”Effects of Inductance on the Propagation Delay and Repeater Insertion in VLSI Circuit”, *IEEE Transactions on Very large Scale Integration (VLSI) Systems*, Vol.8, No.2, pp.195-206, April 2000.

[30] N. Itazaki, Y. Idomoto, K. Kinoshita, ”A fault simulation method for crosstalk faults in synchronous sequential circuits”, *Procs. of Annual Symposium on Fault Tolerant Computing*, pp.38-43, 1996.

[31] Barry W. Johnson, *Design and analysis of fault-tolerant digital systems*, Addison-Wesley Publishing Company, 1989.

[32] Gunok Jung; V.A. Sundarajan, G.E. Sobelman, ”A robust self-resetting CMOS 32-bit parallel adder”, *IEEE International Symposium on Circuits and Systems*, vol.1, pp.I-473 - I-476, May 2002.

[33] Gerry Kane, *PA-RiSC 2.0 Architecture*, Prentice Hall, 1996.

[34] Woo Jin Kim, Yong-Bin Kim, ”Automating Wave-Pipelined Circuit Design”, *IEEE Design and Test of Computers* 20(6): 51-58 2003.

[35] K.-Y. Khoo and A. N. Willson, Jr., ”Single-Transistor Transparent-Latch Clocking”, *Proceedings 16th Conference on Advanced Research in VLSI* pp. 331-341, March 1995.

[36] W. Kuang, J.S. Yuan, R.F. DeMara,M. Hagedorn, K. Fant, ”Performance analysis and optimization of NCL self-timed rings”, *IEE Proc. Circuits, Devices and Systems*, Vol.150 Iss.3, pp.167-172, June 2003.

[37] S. Kim, G.E. Sobelman, ”Digit-serial modular multiplication using skew-tolerant domino CMOS”, *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol.2, pp.1173-1176, 2001.

[38] R.A.Lawes, ”Future Trends in High-Resolution Lithography”, *Journal of Applied Surface Science*, Vol.154-155, No. 1-4, pp.519-526, February 2000.

[39] H.W. Lein, W. Burleson, ”Wave domino logic: theory and applications”, *Proc.Int.Symp.Circuits Syst.*, 1992.

[40] M. Ligthart, K. Fant, R. Smith, A. Taubin, A. Kondratyev, ”Asynchronous design using commercial HDL synthesis tools”, *IEEE Sixth International Symposium on ASAP*, pp.114-125, April 2000.

[41] A. Moser, ”Logic gates with shaped Josephson junctions”, *IEEE Journal of Solid-State Circuits*, Vol.14 Iss.4, pp.672-679, Aug. 1979.

[42] C. Metra, M. Favalli, and B. Riccz, ”On-line detection of logic errors due to crosstalk, delay, and transient faults”, *Test Conference, 1998. Proceeding. International*, 18-23 pp.: 524-533 Oct. 1998.

[43] V.Mehrotra, S.L.Sam, D. Boning, A. Chanddrakasan, R. Vallishayee, and S.Nassif, ”A methodology for Modeling the Effects of Systematic Within-Die Interconnect and Device Variation on Circuit Performance”, *Proceedings of the ACM/IEEE Design Automation Conference*, pp.172-175, June 2000.

[44] Karl M. Fant and Scott A. Brant, *US Patent 5,305,463*, April 19, 1994.

[45] A. Martinez-Smith, R. Sridhar, ”On-line detection of environmentally-induced delay fautls in CMOS wavepipelined circuit”, *ASIC Conference and Exhibit, 1996. Proceedings. Ninth Annual International*, pp.57-60, September 1996.

[46] A.R. Pelella, Pong-Fei Lu, et. al, ”A 2 ns access, 500 MHz 288 Kb SRAM macro”, *IEEE Symposium on VLSI Circuits*, pp.128-129, Jun. 1996.

[47] O.A. Petlin, S.B. Furber, ”Built-in self-testing of micropipelines”, *IEEE Int. Symp. on Advanced Research in Asynchronous Circuits and Systems*, pp.22-29, 1997.

[48] R.S. Parthasarathy, R. Sridhar, ”Double pass-transistor logic for high performance wave pipeline circuits” *VLSI Design, 1998, Proceedings, 1998 Eleventh International Conference on*, pp. 485-500,Jan 1998.

[49] M. Roncken, ”Detect-oriented testability for asynchronous ICs”, *Proceedings of the IEEE*, Vol.87, Iss.2, pp.363-376, 1999.

[50] M. Roncken, ”Partial scan test for asynchronous circuits illustrated on a DCC error corrector”, *IEEE Int. Symp. on Advanced Research in Asynchronous Circuits and Systems*, pp.247-256, 1994.

[51] A.K. Sakallah, N.T. Mudge, T.M. Burks, S.E. Davidson, ”Synchronization of pipeline”, *IEEE Trans.Computer-Aided Design*, vol.12, 1993.

[52] R.Saleh, S.Z.Hussain, S.Rochel, and D.Overhauser, ”Clock Skew Verification in the Presence of IR-Drop in the Power Distribution Network”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol.19,No.6, pp.635-644, June 2000.

[53] T. Santti, J. Isoaho, ”Modified SRCMOS cell for high-throughput wave-pipelined arithmetic units”, *IEEE International Symposium on Circuits and Systems*, vol.4, pp.194-197, May 2001.

[54] S.Sauter, D.Schmitt-Landsiedel, r. Thewes, and W.Weber, ”Effect of Parameter Variations at Chip and Wafer Level on Clock Skews”, *IEEE Transactions on Semiconductor manufacturing*, Vol.13,No.4, pp. 395-400, November 2000.

[55] C.L. Seitz, ”System Timing”, *In Introduction to VLSI System*, C.A.Mead and L.A. Conway, Eds., Addison-Wesley, 1980.

[56] C.J.Seger ”A bounded delay race model”, *Proceedings of the IEEE International Conference on Computer Aided Design*, November 1989.

[57] J.C. Shyur, H.P. Chen, T.M. Parng, ”On testing wave pipelined circuits”, *Proceedings of the 31st annual conference on Design automation conference, ACM Press, Series-Proceeding-Article*, pp.370-374, 1994.

[58] V.N. Shenoy, K.R. Brayton, Sangiovanni-Vincentelli.L.A, ”Minimum padding to satisfy short path constraints”, *proc. Int. Workshop logic Synthesis*, 1993.

[59] M.-D. Shieh, C.-L. Wey, and P.D. Fisher, ”A scan design for asynchronous sequential logic circuits using SR-latches”, *in Proc. Midwest Symp. Circuits and Systems*, pp.1300-1303, 1993.

[60] S.C. Smith, ”Speedup of self-timed digital systems using Early Completion”, *Proc. 2003 IEEE Computer Society Annual Symposium on VLSI*, pp.98-104, April 2002.

[61] R. Smith, K. Fant, D. Parker, R. Stephani, and C.-Y. Wang, ”An Asynchronous 2-D Discrete Cosine Transform Chip”, *Proceedings ASYNC’98*, pp. 224-233, April 1998.

[62] K.S. Stevens, ”Energy and performance models for clocked and asynchronous communication”, *IEEE International Symposium on Asynchronous Circuits and Systems*, pp.56-66, 2003.

[63] Ivan E. Sutherland, ”Micropipelines”, *Communications of the ACM*, 32(6): 720-738, 1989.

[64] M.-T. Sun, T.-C. Chen, and A. M. Gottlieb, ”VLSI Implementation of a 1616 Discrete Cosine Transform”, *IEEE Transactions on Circuits and Systems*, Vol. 36, No. 4, pp. 610-617, April 1989.

[65] Shang-Zhi Sun, David H.C. Du, and Hsi-Chuan Chen, ”Efficient Timing Analysis for CMOS Circuits Considering Data Dependent Delays”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 17, No. 6, June 1998.

[66] K.T.Tang and E.G.Friedman, ”Delay and Noise Estimation of CMOS Logic Gates Driving Coupled Resistive-Capacitive Interconnections” *Integration, the VLSI Journal*, Vol. 29, No. 2, pp.131-165, September 2000.

[67] K.T.Tang and E.G.Friedman, ”peak Crosstalk Noise Estimation in CMOS VLSI Circuits”, *Proceedings of the IEEE International Conference on Electronics, Circuits and Systems*, pp. 1539-1542, September 1999.

[68] D. Talukdar, R. Sridhar, ”An analytical approach to fine tuning in CMOS wave-pipelining ”, *ASIC Conference and Exhibit, Proceedings, Nith Annual IEEE International*, 23-27 Sep 1996, pp: 205-208

[69] F. Towler,J. Chu, et. al, ”A 128k 6.5 ns access/5 ns cycle CMOS ECL static RAM”, *IEEE International Solid-State Circuits Conference*, pp.30-31, Feb. 1989.

[70] A. Venkataraman, I. Koren, ”Determination of yield bounds prior to routing”, *Proc. of the 1999 IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems,* pp. 4-13, November 1999.

[71] Dimitrios Velenis, ”Delay Uncertainty in High Performance Clock Distribution Networks”, *PhD Thesis, Department of Electrical and Computer Engineering, University of Rochester*, Rochester, New York, 2003.

[72] A. Vittal, and M. Marek-Sadowska, ”Crosstalk reduction for VLSI”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 16(3):290-298 March 1997.

[73] Neil H.E Weste, Kamran Eshraghian, ”Principles of CMOS VLSI Design, A Systems Perspective”, *Addison-wesley Publishing Company*, 1994.

[74] D. Wong,G.De. Micheli, M. Flynn, ”Inserting active delay elements to achieve wave pipelining”, *Proc.Int.Conf.Computer-Aided Design*, Nov. 1989, pp. 270-273

[75] Derek C. Wong, Giovanni De Micheli, and Michael J. Flynn, ”Designing high-performance digital circuits using wave pipeline: algorithms and practical experiences”, *IEEE Trans. on computer aided design of integrated circuits and systems*, 12(1), January 1993.

[76] Burleson.W.P , Ciesielski.M, Klass.F, and Liu.W, ”Wave-Pipelining: A Tutorial and Research Survey”, *IEEE Transactions on very large scale integrate (VLSI)system,* VOL.6, NO. 3, September 1998.

[77] W.C. Wu, C.Len Lee, ”A probabilistic testability measure for delay faults”, *Proceedings of the 28th conference on ACM/IEEE design automation conference*, pp.440-445, June 17-22, 1991.

[78] H. You, M. Soma, ”Crosstalk Analysis of Interconnection Lines and Packages in High-Speed integrated circuits”, *IEEE Trans. Circuits Syst. I, Fundam, Theory Appl*, 37(8):1019-1026, August 1990.

[79] H. You, M. Soma, ”Crosstalk and transient analysis of high-speed interconnects and packages”, *IEEE J. of solid state circuit*, 26(3):319-329, March 1991.

[80] M.Yoshizawa and S. Moriya, ”Comparative Study of Resolution Limiting Factors in Electron Beam Lithography Using the Edge Roughness Evaluation Method”, *Journal of Vaccum Science and Technology (Microelectronics and nanometer Structures)*, Vol, 19, No.6,pp.2488-2493, November 2001.

[81] Intel Corp. $http : //www.intel.com$

[82] Don Bouldin, ”Estimiting Delays”, $http : //vlsi1.engr.utk.edu/ece/bouldin_courses/651$

[83] Xiang Lu, Weiping Shi, ”ISCAS Benchmark Circuit” $http : //dropzone.tamu.edu/ xiang/iscas.html$

VITA

Tao Feng

Candidate for the Degree of

Doctor of Philosophy

DISSERTATION: A STUDY ON THE MAXIMUM SPEED AND RELIABILITY ASSURANCE IN WAVE PIPELINE-BASED COMBINATIONAL CIRCUITS

Major Field: Computer Science

Biographical:

Personal Data: Born in Qingzhou, Shandong, China, on October 29, 1974, the son of Guangsheng Feng and Sumei Wang.

Education: Graduated from Qingzhou First High School, Shangdong, China in July 1989; Received Bachelor of Science degree in Natural Resources and Environment from Northwest A&F University, Xi'an, China in July 1996; Received Master of Science degree in Computer Science from Oklahoma State University, Stillwater, Oklahoma in May 2003; Completed the requirements for the Doctor of Philosophy degree in Computer Science at Oklahoma State University in December 2006.

Experience: Employed as an chemical engineer in Chinese Tobacco Company from July 1996 to August 2000. Employed as a graduate research assistant; Department of Computer Science, Oklahoma State University, 2003 to 2004. Employed as a teaching assistant; Department of Computer Science, Oklahoma State University, 2004 to 2006. .

Name: Tao Feng Date of Degree: December, 2006

Institution: Oklahoma State University Location: Stillwater, Oklahoma

Title of Study: A STUDY ON THE MAXIMUM SPEED AND RELIABILITY ASSURANCE IN WAVE PIPELINE -BASED COMBINATIONAL CIRCUITS

Pages in Study: 138 Candidate for the Degree of Doctor of Philosophy

Major Field: Computer Science

Scope and Method of Study: Wave pipeline is one of the revolutionary technologies beyond conventional pipeline in the microprocessor architecture research area. Clockless wave pipeline is the cutting-edge and innovative pipeline without relying on clock signal. Due to the stringent requirement for high density and performance of current VLSI technology, reliability is being considered as one of the most crucial issues. Reliability modeling and optimization techniques have been applied extensively. Clock frequency is one of the keys to achieve the fast circuit speed. A clock cycle time optimization and analysis method is proposed in order to achieve a ultra high clock frequency in the context of the proposed new wave pipeline.

Findings and Conclusions: The reliability-driven design and optimization techniques for the clockless wave pipeline are proposed. It is mainly focused on the two parts, i.e., request signal and datawave. A more aggressive technology beyond wave pipeline with ultra-high throughput and speed towards maximum circuit frequency is mainly proposed, analyzed, simulated, and verified.

ADVISOR’S APPROVAL: Dr. Nohpill Park