University of Newcastle upon Tyne

School of Electrical and Electronic Engineering

Design of Variation-Tolerant Synchronizers for Multiple Clock and Voltage Domains

By

Mohammed Saleh Abdullah Alshaikh

A thesis submitted for the degree of Doctor of Philosophy (PhD) at

Newcastle University

April 2014

Abstract

Parametric variability increasingly affects the performance of electronic circuits as the fabrication technology has reached the level of 32nm and beyond. These parameters may include transistor Process parameters (such as threshold voltage), supply Voltage and Temperature (PVT), all of which could have a significant impact on the speed and power consumption of the circuit, particularly if the variations exceed the design margins. As systems are designed with more asynchronous protocols, there is a need for highly robust synchronizers and arbiters. These components are often used as interfaces between communication links of different timing domains as well as sampling devices for asynchronous inputs coming from external components. These applications have created a need for new robust designs of synchronizers and arbiters that can tolerate process, voltage and temperature variations.

The aim of this study was to investigate how synchronizers and arbiters should be designed to tolerate parametric variations. All investigations focused mainly on circuit-level and transistor level designs and were modeled and simulated in the UMC90nm CMOS technology process. Analog simulations were used to measure timing parameters and power consumption along with a “Monte Carlo” statistical analysis to account for process variations.

Two main components of synchronizers and arbiters were primarily investigated: flip-flop and mutual-exclusion element (MUTEX). Both components can violate the input timing conditions, setup and hold window times, which could cause metastability inside their bistable elements and possibly end in failures. The mean-time between failures is an important reliability feature of any synchronizer and depends exponentially on the metastability recovery time constant τ and the delay through the synchronizer.

The MUTEX study focused on the classical circuit, in addition to a number of modifications at the circuit and transistor levels, to adjust the value τ and its tolerance, based on increasing internal gain by adding current sources, reducing the capacitive loading, boosting the transconductance of the latch, compensating

the existing Miller capacitance, and adding asymmetry to maneuver the metastable point. The results showed that some circuits had little or almost no improvements, while five techniques showed significant improvements by reducing τ and maintaining high tolerance.

Three design approaches are proposed to provide variation-tolerant synchronizers. First, the wagging synchronizer is proposed to significantly increase reliability over that of the conventional two flip-flop synchronizer. The robustness of the wagging technique can be enhanced by using robust τ latches or adding one more cycle of synchronization. The second approach is the Metastability Auto-Detection and Correction (MADAC) latch which relies on swiftly detecting a metastable event and correcting it by enforcing the previously stored logic value. This technique significantly reduces the resolution time down from uncertain to certain time. Finally, a pseudo level-shifting handshake synchronization technique is proposed to transfer signals between Multiple-Voltage Multiple-Clock Domains (MVD/MCD) that do not require conventional level-shifters between the domains or multiple power supplies within each domain. This interface circuit uses a synchronous set and feedback reset protocol which provides level-shifting and synchronization of all signals between the domains, from a wide range of voltage-supplies and clock frequencies.

Overall, synchronizer circuits can tolerate variations to a greater extent by employing the wagging technique or using a MADAC latch, while MUTEX tolerance can suffice with small circuit modifications. Communication between MVD/MCD can be achieved by an asynchronous handshake with internal resetting protocols without a need for adding level-shifters.

Acknowledgments

At first, All thanks and gratitude to Allah the most graceful the most merciful.

> *"In the name of Allah, Most Gracious, Most Merciful. Praise be to Allah, the Cherisher and Sustainer of the worlds; Most Gracious, Most Merciful; Master of the Day of Judgment. Thee do we worship, and Thine aid we seek. Show us the straight way, The way of those on whom Thou hast bestowed Thy Grace, those whose (portion) is not wrath, and who go not astray."* [1]

After that, I would like to express my deepest thanks to Prof. Alex Yakovlev. His advice, support and motivation kept me going forward throughout the years to finish my thesis. I would also want to say great thanks to Dr. Gordon Russell. His exceptional feedback and valuable comments have been a great value to me.

Also, I am grateful for the chance to work under the guidance of, full of inspiration and kindness, Prof. David Kinniment during my first year.

I want to thank all my friends and colleagues in the Microelectronics System Design group for their valuable discussions in many different occasions. I also want to thank all my friends, in the UK and Saudi Arabia, for their cheer and support.

In Saudi Arabia, I present my thanks to my mother, brothers and sisters for the encouragement and endless support. Especially, I would like to express a great love and thanks to my lovely wife, *Norah*, who kept standing by my side, encouraging, motivating and comforting me during health and sickness towards progressing over the years of my study and research, as well as sharing enthusiastically all my achievements. I am also grateful for my son '*Aziz*' who brings smile to us every day.

I would like to thank Umm Al-Qura University, Saudi Arabia, and the Saudi Cultural Bureau Attaché, in the Saudi Arabian Embassy in London, for providing me with the scholarship to finish my thesis.

1 Surat Al-Fatihah, The Holy Quraan, Chapter 1, Verses 1:7.

List of Publications

Conference Papers:

1. M. Alshaikh, D. Kinniment, and A. Yakovlev. “A synchronizer design based on wagging.” in Proceeding 2010 *IEEE 22nd International Conference on Microelectronics* (ICM 2010), Cairo, Egypt, Dec. 19-22, 2010, pp. 415-418.

2. M. Alshaikh, D. Kinniment, and A. Yakovlev. “On the trade-off between resolution time and delay times in bistable circuits.” in Proceeding 2009 *IEEE International Conference on Electronics Circuits and Systems* (ICECS 2009), Yasmin Hammamet, Tunisia, 13-16 Dec 2009, pp. 355-358.

Technical Reports:

3. M. Alshaikh, D. Kinniment, and A. Yakovlev, “Robust Synchronization using the Wagging Technique.” NCL-EECE-MSD-TR-2010-165, Microelectronic System Design Group, School of EECE, Newcastle University, December 2010.

Workshops:

4. M. Alshaikh, G. Russell, and A. Yakovlev. “Level-Shifting Handshake Synchronizers,” accepted fresh ideas workshop in ASYNC 2014.

List of Contents

List of Figures

List of Tables

Chapter 1 Introduction

Increasing unpredictability and vulnerability to process, voltage and temperature variations in sub-nano CMOS process technologies suggests that current optimal designs in cell libraries must be reviewed and refined. Many VLSI systems and architectures are designed with more asynchronous cells, which need to be made more resilient to these variations. Cells which particularly affect the performance of systems on silicon include synchronizers and arbiters, which affect the latency between independently clocked processors and asynchronous systems, and register bits which require a time to set and hold data. These effects are expected to increase as technology nodes reach the level of 32nm and beyond [1].

In the subsequent sections of this chapter, the sources of variability and their impact on device parameters and circuit performance are surveyed. Thereafter, some of the techniques to reduce the effects of variability are reviewed. Then, the thesis motivation is stated, followed by the thesis main contributions, and finally the thesis organization.

1.1 Technology and Variability

In this section, the nature of scaling CMOS transistors' process technology nodes and their effects are discussed. Then, the common sources of performance variability of CMOS devices and circuits are reviewed, mainly in terms of process, voltage and temperature, as well as radiation.

1.1.1 CMOS Transistor Scaling

The scaling down of transistor dimensions leads to reduction in cost and improvement in performance per unit transistor. Technology scaling main goals, for a logic gate, are to reduce its delay time, increase its density, and decrease its energy per switching operation [2]. At the present time, the scaling rate of the feature size is 0.7X of transistor dimensions per two to three years [1, 3, 4] corresponding to Moore's Law [4, 5]. The reduction of gate size showed a decrease

in the delay time by 30%, an increase of its density by two times and a reduction in the consumed energy per switching operation by 65%. This leads to an increase in clock frequency of 43% along with lowered power consumption by 50%. Over the years, since the start of integrated circuit technologies in the 1960s, the size of one transistor continued to shrink from a few microns down to tens of nanometers today, which kept leading to a greater integration capacity. Similarly, the amount of energy needed for charging and discharging capacitors is reduced due not only to the reduction in capacitor area but also the reduction in supply voltage down to 1V, as a result of that, the energy for writing or reading one data bit is decreased. Table 1.1 summarizes the impact of scaling on CMOS transistor parameters [4], where S is the scaling factor. Table 1.2 shows an example highlighting the scaling impact on transistor parameters of UMC CMOS process technology nodes from 250nm down to 65nm[6].

Table 1.1 Summary of scaling impact on CMOS transistor parameters [4]

Device parameters	Scaling effect
Dimensions: L , W , t_{ox}	$1/S$
Voltages: V_{DD} , V_{THn} , V_{THp}	$1/S$
Doping concentration density	S
Drain Current: I_{DS}	$1/S$
Conductance: g_{out} and g_m	1
Gate capacitance: C_{gate}	$1/S$
Delay: RC	$1/S$
Clock frequency	S
Power and Area	$1/S^2$
Energy	$1/S^3$

On the other hand, scaling down of global interconnect dimensions do not follow the scaling of local ones nor the transistor dimensions, which lowers the overall performance [2]. Basically, scaling wires results in the increase of the wire resistance per unit length by a factor of S^2, whereas, the wire capacitance per unit length is constant. Overall, the chip area should be halved every two consecutive technology nodes, however, as more transistors and functions are integrated on a single chip in recent designs, the need to increase the area of the chip is accompanied by an increase in global interconnect length and RC time constant.

Table 1.2 Scaling of UMC CMOS technology (logic/mixed-mode data) [6]

Technology node	250nm	180nm	130nm	90nm	65nm
Substrate Type	P-substrate	P-substrate	P-substrate	P-substrate	P-substrate
Poly Layers	1	1	1	2	1
Metals Layers	5	6	8	9	10
L_{min} (μm)	0.24	0.18	0.12	0.08	0.06
W_{min} (μm)	0.3	0.24	0.16	0.12	0.08
T_{OXn} (nm)		7.0	2.63	2.25	2.05
V_{DD} Core (V)	2.5	1.8	1.2	1	1.0~1.2
V_{ton} NMOS (V)	0.54	0.51	0.47	0.33	0.23
V_{ton} PMOS (V)	-0.58	-0.5	-0.42	-0.277	-0.19
Core delay (ps/stage)*	40	27	19.6	10.6	6

* A stage accounts for one logic inverter gate delay in a ring oscillator without load.

1.1.2 Sources of Variability

The performance of circuits is dictated by the characteristics of devices and interconnects and operating conditions. Parametric variability is any change in the design due to deviations in the chip's internal or external characterizing parameters. Variability in performance could be permanent or temporary [7-9]. Process variations cause permanent physical alterations to the wafer down to atomic level. On the other hand, deviation in supply voltage and surrounding temperature only impact on circuit operation briefly unless it exceeds a maximum value. Also, external radiation particles striking the transistor lattice could introduce temporary failures. Hence, these are considered temporary environmental variations. Variations also can be categorized as environmental or physical variations [7-9]. Environmental variations typically include changes in surrounding temperature and supply voltage. Physical parameters normally are independent of time and constitute the majority of process variations that occur during the manufacturing process.

In the following section the sources of process, supply voltage and temperature (PVT) variations, as well as radiation, together with their impact on the performance of CMOS circuits will be outlined.

1.1.2.1 Process Variability

Manufacturing process imperfections, such as poor masking, are the main source

of process variations. These variations include any alteration in process parameters which characterize the behavior of passive and active devices. These parameters include transistor dimensions, threshold voltage, oxide-thickness and carrier mobility, as well as the capacitance and resistance of interconnects. Process variations can be grouped into two broad areas: spatial and temporal [7, 10, 11].

Spatial process variations are divided into two categories: die-to-die variations and within-die variations [7, 10, 11]. Die-to-die variations refer to the deviation from the chips' mean value and affect all devices in the same chip, they are sometimes referred to as inter-die variations. On the other hand, within-die variations (also known as intra-die variations) occur when there is a spatial deviation or mismatches across a single chip, they are mainly caused by random doping levels, and lithograph limitations, which therefore affect the certainty of device threshold voltage and mobility [12, 13].

The effects of fluctuation and randomness of doping in a device channel become more challenging with scaling of technology, because the number of dopant atoms in the channel of a transistor has reduced exponentially [14], as shown in Figure 1.1. Therefore, the total number and location of atoms in the device channel becomes highly significant in the deviation of the threshold voltage, and similarly in the source and drain, random dopants cause an irregular edge of the source and drain which adds variation to their resistance and capacitance. Moreover, the lithographic wavelength used to form transistor patterns below 250nm technology nodes has remained constant at 193nm [14, 15], as shown in Figure 1.2, which causes physical layout imperfections.

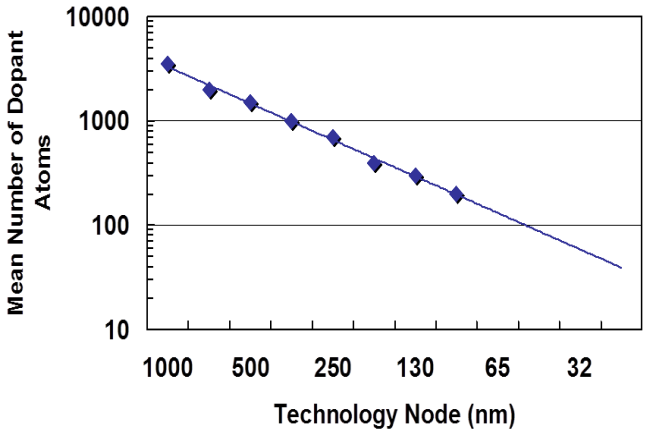


Figure 1.1 Dopants levels [14]

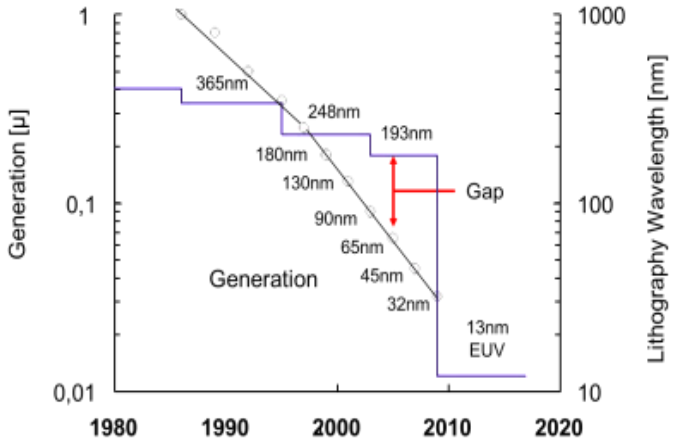


Figure 1.2 Lithography limitations [15]

Based on recent reports from the International Technology Roadmap for Semiconductors (ITRS) [16], the anticipated amount of variability in threshold voltage due to random doping variations is very large, and it could exceed 50% in 2018 technology nodes. The variability trend is plotted in Figure 1.3. In 2003, *Borkar et al*. [17] showed that normally distributed threshold voltage variations in microprocessor chips fabricated in 180nm CMOS logic technology caused frequency variations of 30% and leakage current variations of around 20 times, as shown in Figure 1.4. *Bowman et al.* [18] found that the number of critical paths increases as the variability increases and the maximum operating frequency also suffers as process variations increase with scaling.

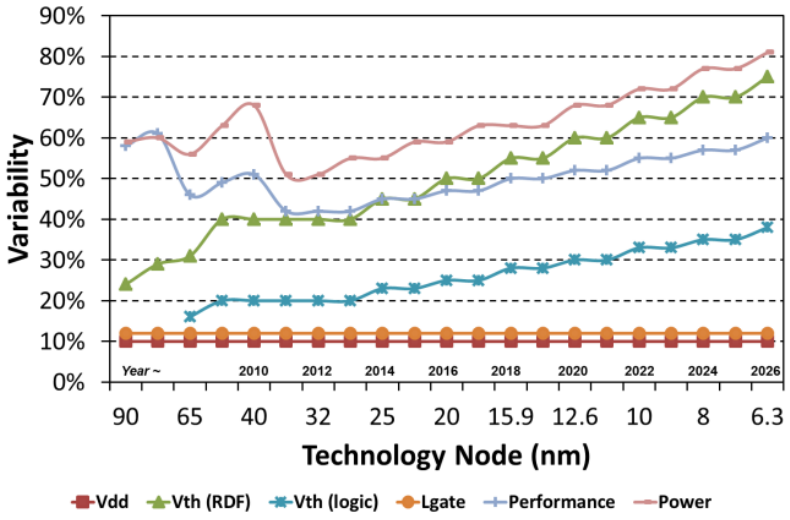


Figure 1.3 ITRS design for manufacturability requirements [16]

There are some physical stress parameters that change rather slowly during the operational life of the chip and participate in the aging process of devices and interconnects [12]. Some stresses do not have a significant impact until a

catastrophic failure occurs, others can have a significant effect even in the early stages of degradation. These are considered temporal process variations and mainly include Negative Bias Temperature Instability (NBTI), Hot Electrons Injection (HCI) and electromigration [8, 9, 12, 19]. For example, NBTI could increase the threshold-voltage of a PMOS transistor by which it becomes slower, and HCI on the silicon-oxide gate of an NMOS transistor could increase its threshold-voltage, by which it becomes slower. The electromigration of metal interconnect could increase its propagation delays. This is caused by significant current densities and increased pressure of carrier collisions on metal atoms which causes a slow displacement of the metal interconnects.

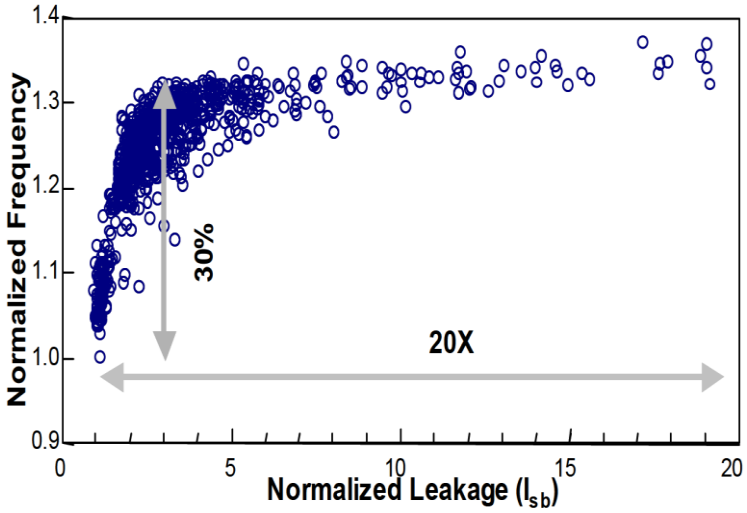


Figure 1.4 Process variation impact frequency and leakage distribution [17]

HCI occurs, at device level, when an electron or a hole gains enough energy to overcome the potential barrier between silicon and oxide, thus becoming a hot carrier [8, 12, 19]. Hot carriers can degrade the dielectric material causing trap structures for electrons and holes, which increase leakage currents and alter threshold voltages prior to failing. It mostly impacts on NMOS transistors which become slower. From a circuit perspective, HCI occurs when both gate and drain voltages are significantly higher than the source voltage.

NBTI mostly affects PMOS devices, particularly during the device operation in the linear region. It is a result of the generation of silicon dangling bonds which form interface traps [8, 12, 19] and cause an increase in the absolute threshold voltage and a decrease in the drain current and transconductance and, as a result, a PMOS transistor becomes slower.

1.1.2.2 Voltage Variability

The supply voltage plays a fundamental factor in the design and analysis of integrated circuits. It determines the amount of current flowing through devices, the noise-margins of digital circuits and the power dissipation, it also limits the maximum switching frequency of a transistor. Therefore, any reduction in the supply voltage affects the circuit operational speed. Supply voltage fluctuations are generally a result of rapid variations in switching activity and poor power grid design, which causes a large amount of charge drawn from supply rail and results in an unbalanced dynamic power consumption across the chip and a droop in the supply voltage [17, 20, 21], as shown in Figure 1.5. A voltage droop is a small reduction in the supply voltage due to the fast rate of current change through the inductance of packaging and grid distribution. These variations may result in slower speed and temperature hot spots.

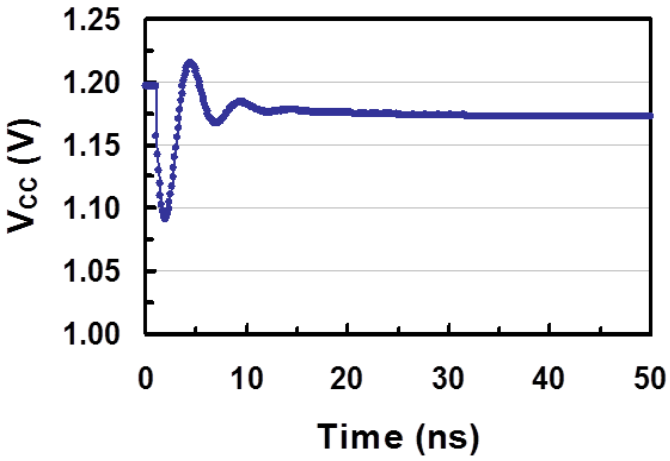


Figure 1.5 Supply voltage droop [22]

1.1.2.3 Temperature Variability

Variations in the surrounding temperature impacts on the operation of the circuits. Temperature variations may be influenced by the circuit during operation, and fluctuate across the chip as in Figure 1.6, depending on the circuit configuration, the switching frequency and power supply fluctuations, as well as the surrounding temperature, and whether it is provided with heat-sinks or not, as a result of hot spots may occur and the circuit's performance degraded [17, 20, 21]. This is because of the device and interconnects parameter dependence on temperature which decreases circuit performance. The empirical formulas, in Equation (1.1) below, show the effect of temperature on threshold voltage and mobility [4]. For instance, an increase in temperature causes a reduction in carrier mobility,

saturation mobility, threshold voltage and saturation current of transistors, whereas transistor sub-threshold leakage and interconnects resistance are significantly increased.

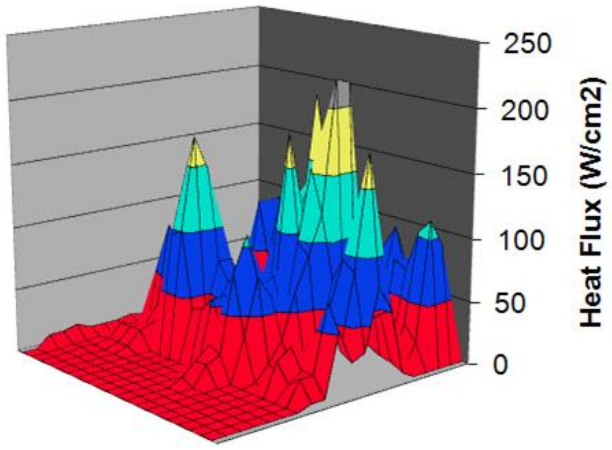


Figure 1.6 Heat fluctuation across die [14]

$$V_{TH}(T) = V_{TH}(T_0) \cdot \left(1 + TCV_{TH}(T - T_0)\right)$$
$$\mu(T) = \mu(T_0) \cdot (T_0/T)^{1.5} \quad (1.1).$$

1.1.2.4 Radiation particles

Soft errors are random transient errors in digital circuits caused by alpha particle radiation striking the substrate region in a device [4], as shown in Figure 1.7. They are emitted during the decay process of radioactive impurities in the packaging material. They are also induced by high-energy neutrons from cosmic rays. An alpha particle strike of a few nanoseconds creates a trail of hole-electron pairs in the substrate that could result in ion diffusion into the depletion region of a p-n junction collecting the charge. This charge is visible to the circuit as a current spike or a glitch and, if it exceeds a critical amount, may alter the state of the node causing a single event upset fault. This critical point decreases with technology scaling because it is dependent on the node capacitance and voltage as well as the circuit structure, increasing the probability of soft error failures. In order to reduce the probability of soft error failures, the critical nodes need to be high in capacitance and voltage.

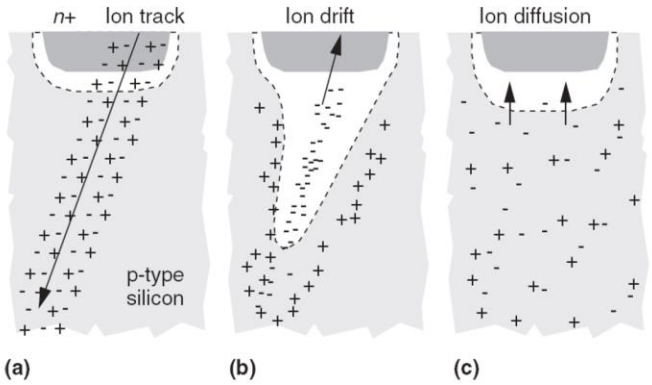


Figure 1.7 Alpha-particles striking substrate [4]

1.2 Variation-Tolerant Circuit and Microarchitecture Techniques

Several techniques have been proposed to compensate for the impact of variability on performance and power. Most of them target particular problems, such as threshold voltage variation at device level, chip overheating, and timing variations. Several approaches combine two or more techniques to improve performance against variability. In the following sections, some of the variation reduction design techniques will be discussed.

1.2.1 Adaptive Voltage Techniques

One of the serious problems of variability is the variations of the threshold voltages, which define device and circuit performance. To increase the number of accepted chips produced, several techniques were developed to improve the devices' performance by applying an appropriate body to source voltage to each die after fabrication to control both threshold voltage and leakage current, which helps reduce threshold voltage variations. The amount of voltage applied shifts the chip mean threshold voltage towards the target range of performance. The most commonly used of these techniques is to measure the variability in performance and leakage current of the chip with respect to the targeted tolerance, then apply a feedback body bias and then repeat measurements until the variability is mitigated. According to the literature [23-25], body bias is applied to all PMOS and NMOS transistors in a chip and can be forward, zero or reverse. The forward body bias increases the operating frequency, whereas applying a reverse body bias reduces the leakage current.

One of the earliest researchers [23] who adopted this technique towards die-to-die variations proposed the adaptive body bias technique to reduce the spread of

threshold voltage values for large number of die samples and to enhance their production yield. A different body bias was applied to different dies, that is, one body bias per die, depending on the difference between the target threshold voltage mean and that of the die. Their findings showed improvement in the threshold voltage variations between dies but also showed an increase in the within-die variations with channel length variations of 5%. Further research [24] considered an improved adaptive body bias technique to reduce die-to-die variations further, by finding the best PMOS and NMOS body bias combination for each die. Furthermore, improvements [24] on the previous technique [23] to compensate for within-die variations, where a unique body bias combination is applied to each circuit in the chip, increases the yield to three times of that of [23]. Both techniques showed improvement, but this improvement depends on the body bias voltage resolution, which may add to the complexity of the chip.

Another technique to reduce the variability of performance is known as adaptive supply voltage. Using this method, variations are reduced by decreasing the supply voltage of dies with high leakage current and increasing it for the ones with lower maximum frequency. The more the supply voltage resolution, the more effective is this technique. Both adaptive supply voltage and adaptive body bias can be combined together [25] to further reduce the impact of within-die variations and increase the yield. This enhanced technique showed a significant improvement in the number of accepted dies, which was nearly 98%. However, it consumes more area and increases the design complexity and power requirements [25].

In contrast, dynamic supply voltage variations have been an issue with increased switching activity and could not be solved using the adaptive supply voltage method. One known approach is adding on-die decoupling capacitors [26], which reduces the dynamic variations in the supply voltage. This improvement is dependent on the number of decoupling capacitors, but comes at the cost of area and increased gate oxide leakage in the sub-90nm technologies [17, 27]. An alternative method is to disable the power supply of clocked circuit blocks during idle stages, for example using sleep high-threshold-voltage transistors, which reduces the amount of excessive switching [27].

Temperature variations across the chip can strongly affect the chip's performance, since the material's electrical properties depend on temperature. To control the

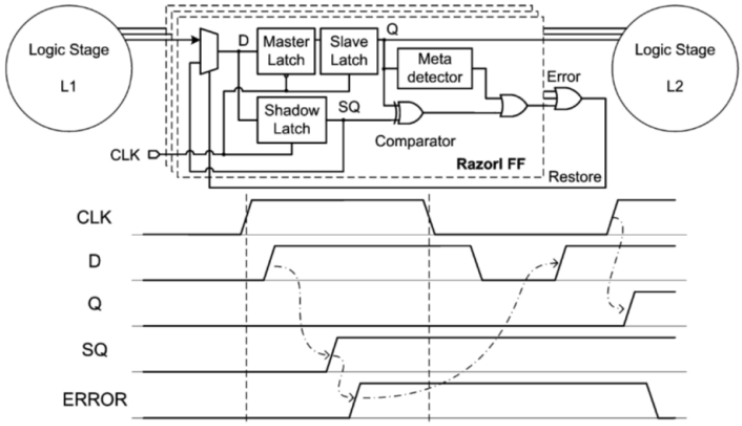
temperature on a chip, the throttling technique [17, 27] can be used to reduce both the operating frequency and the supply voltage when the chip's ambient temperature reaches its maximum limit, followed by a drop in power dissipation and then temperature. Adaptive body bias and supply voltage techniques can be combined together with internal thermal sensors at different points across the chip to control the temperature rise, as described by *Tschanz et. al.*[28]. Temperature and supply voltage readings are used to find the best combination of body bias, supply voltage and frequency by means of a look-up table, and then tune the circuit operation to reduce the operation temperature.

1.2.2 Error Detection and Recovery Circuits

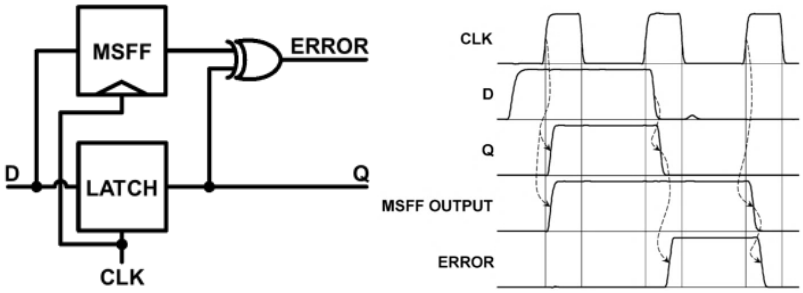
Microarchitecture techniques can be very effective in reducing the impact of dynamic variations. One traditional method is the error detection and correction scheme, which has been used commonly in different circuits and architectures. Most error detection techniques are based on checking output signals, while some techniques focus their attention on input signals. Any error has a number of different causes. Usually, transient errors are of concern during operation, in other words, they might occur during normal operation of the system. Some transient errors are caused by longer critical path delays resulting in timing errors, or logical errors. At circuit level, the common sources of transient errors could be as a result of PVT variations [21], timing violations causing a metastability failure [29], or alpha particle strikes causing soft error faults [4, 5, 30, 31].

ARM proposed the Razor processor [32, 33] which utilizes timing error detection and correction techniques combined with dynamic voltage scaling and error rate monitoring techniques, to operate correctly at a critical supply voltage, that is, adequate under PVT variations. In general, the Razor flip-flop uses a datapath master-slave D flip-flop accompanied by a shadow latch and metastability and error detection circuits, as shown in Figure 1.8(a). Input data is sampled in the datapath flip-flop at the positive edge of the clock, and then sampled in the shadow latch at the negative edge of the clock. Then, the outputs of the flip-flop and latch are compared and if they differ, an error signal is produced to enable an error recovery mechanism to flush the pipeline. A similar technique was presented by Bowman [34, 35] known as the Double Sampling with Time Borrowing (DSTB) flip-flop and comprised a datapath latch and shadow master-slave flip-flop with a

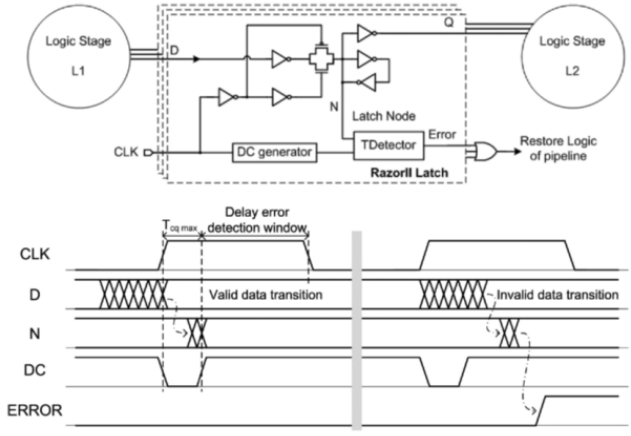
comparison circuit to detect errors as shown in Figure 1.8(b). Its main advantages are design simplicity and removing metastability from the datapath.



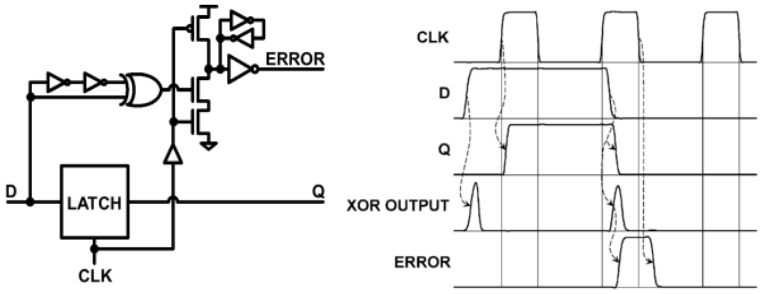
(a) Razor flip-flop [33]



(b) DSTB flip-flop [34, 35]



(c) Razor II flip-flop [36]



(d) TDTB latch [34, 35]

Figure 1.8 Error detection/recovery in flip-flop and latch circuits

On the other hand, the early error detection technique, which is based on input transition detection, was presented in Razor II flip-flop [36] and Transition-Detection Time-Borrowing (TDTB) latch [34, 35], as shown in Figure 1.8(c) and Figure 1.8(d). Both techniques use a positive level-sensitive latch with a data input transition detector and clock detector. The transition detector senses any input data transition during the positive clock phase, and in the case of a flagged transition, the operating frequency is reduced and the instruction is replayed to correct the signal. Both techniques show an increased complexity compared to error detection being applied at the output.

Overall, novel error detection techniques along with adaptive techniques improved error rate as well as energy efficiency in the presence of different timing variations and faults. Transition detection techniques increase the design complexity, whereas output error detection techniques keep it simple.

1.2.3 Hardware Redundancy

An alternative tactic to reduce the impact of soft errors is to add two more redundant circuits in parallel, followed by a majority voting circuit to produce the appropriate output, as shown in Figure 1.9. This technique is known as Triple-Modular-Redundancy (TMR) [4, 30]. In addition, adding one or more redundant paths within the cell itself can improve the robustness towards current spikes caused by alpha particle strikes, this is known as circuit-level hardening [30] , for example radiation hardened flip-flops designs [4, 37] and soft-error tolerant memory cell designs [31, 38, 39].

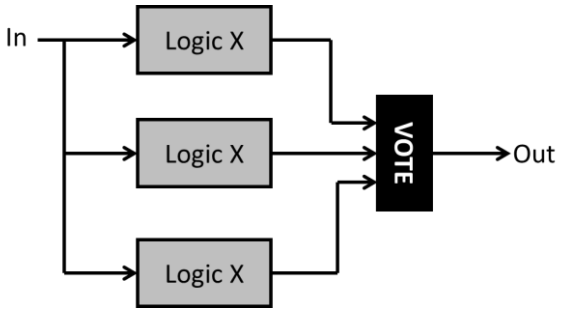


Figure 1.9 Triple-Modular-Redundancy technique

A hardware redundancy technique can be implemented using two or more circuits in addition to monitoring and control circuits. The monitoring circuit tests all the circuits and decides which one has the best performance, for instance in terms of process variations, then enables the best circuit and disables the others. This

technique was used in the synchronizer selection scheme in [40]. This technique improves the circuits' performance against permanent faults and aging process effects, but they have a negative impact on power and area overhead [30].

1.2.4 Asynchronous Circuits and Systems

Asynchronous systems tend to have a unique advantage towards timing variations, as they require either more than one local clock or no clocks at all, which eliminates the problem of clock distribution and timing constraints, even with PVT variations spread across the chip [41, 42]. In addition, they would be the inevitable choice with the increase in intra-die variability against new technologies [21]. According to ITRS reports, the utilization of asynchronous global signaling on chip is expected to increase to 30% of the chip design by 2016 [16], as shown in Figure 1.10.

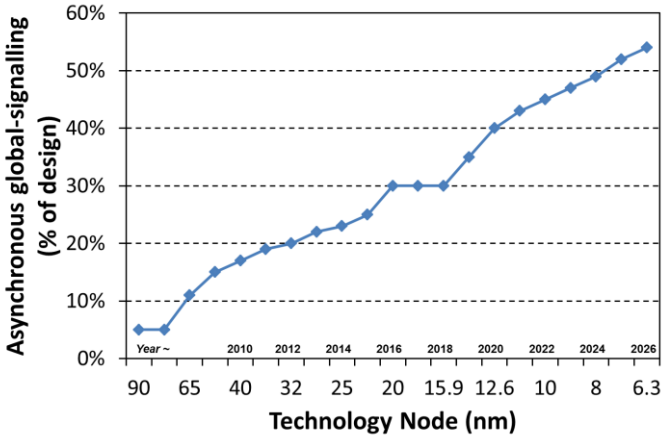


Figure 1.10 ITRS asynchronous design requirements [16]

Some approaches in the asynchronous world, such as Quasi-Delay Insensitive (QDI), which has no clocks and assumes delays only exist at isochronic forks, can actually tolerate timing variations and should never lead to failure [43] as well as consuming less dynamic energy.

Another approach called Globally-Asynchronous-and-Locally-Synchronous (GALS) [44], which is constructed from large synchronous blocks that communicate with each other using asynchronous interfaces, can eliminate the global clock distribution problem, even though they may have to face up to some inherited problems [43], for example, metastability operation in arbiters and synchronizers. The asynchronous links between the clocked regions require controlled handshake synchronization which leads to reduced maximum frequency and an increased

area overhead, especially under the limited availability of design tools [44-47]. The GALS architecture, in contrast with a global-synchronous architecture, can provide faster performance by at least 8% under within-die variations in gate length and thermal distribution [48]. These will be discussed further in Chapter 2.

1.3 Thesis Motivations

Integrated circuit scaling has some advantages and a few disadvantages; one of its disadvantages is the increase of parametric variability with every new technology node [16], which consequently reduces the chip production yield. These parameters may include transistor process parameters, supply voltage and temperature, which could have a huge impact on the circuit speed and power consumption if varied outside their design margins [17]. This effect is expected to continue to increase as the fabrication technology reaches the level of 32nm and beyond. Many VLSI system architectures, such as network-on-chip, are designed with more asynchronous circuits, which need to be more resilient to parametric variations.

Literature shows a number variation tolerant design techniques for parameters within different levels of abstraction. On the device level, there are some solutions to control variations in threshold voltage and leakage current by using adaptive body bias techniques [17, 23, 24, 45], and in a similar manner voltage supply variation can be improved with an adaptive supply voltage [17, 25-28, 45]. Others offered hardware redundancy on circuit level [4, 30], such as soft-error tolerant designs [31, 37-39]. Micro-architecture error detection and correction techniques, which are based on voltage scaling, timing errors detection and self-correction [34, 35], such as the Razor techniques [32, 33, 36, 49].

A System-on-Chip (SoC) cannot operate efficiently with a single global clock, due to parametric variations across the chip which makes the control of all the delays unlikely to be achieved. Asynchronous techniques, such as GALS [44], are the solution to SoCs [16, 21, 41, 42, 50]. Synchronizers and arbiters are special circuits commonly used in systems with multiple clock domains or clockless domains. A synchronizer is used to synchronize signals between Multiple-Clock Domains (MCD) while an arbiter selects between two requests. Their circuits are constructed based on one or more bistable elements, for example flip-flops, which

must receive stable input signals for a short specified window of time. Because synchronizers and arbiters may receive input signals with arrival times violating that specification, they are more susceptible to metastability problems [29, 41, 50, 51], which lead to apparently chaotic transient behavior that draws more current from the supply rail, and can result in an unpredictable state in the following stage.

The main objective of multiple voltage design is to reduce the overall power consumption, as well as to reduce the effects of process variations [52-54], by providing different voltage domains that are either constant or variable [55]. Sometimes, the reduction of voltage supply requires a reduced clock frequency as in the case of Dynamic Voltage and Frequency Scaling (DVFS), which creates an MCD challenge. Hence, there is a need to study the design of synchronizers placed between multiple voltage domains.

The aim of this thesis is to investigate how synchronizers and arbiters should be designed to tolerate parametric variations. All investigations focused mainly on circuit-level and transistor level designs and were modeled and simulated in the UMC90nm CMOS technology process [6]. Analog simulation was run using the Cadence Spectre Virtuoso [56] CAD tool to measure timing parameters and power consumption along with a "Monte Carlo" statistical analysis [57] to account for process variations.

Two main components of synchronizers and arbiters were primarily investigated: flip-flop and mutual-exclusion element (MUTEX). Both components can violate the input timing conditions, setup and hold window times, which could cause metastability inside their bistable elements and possibly end in failures. The mean-time between failures is an important reliability feature of any synchronizer and depends exponentially on the metastability recovery time constant τ and the delay through the synchronizer. Both circuits were optimized to reduce the impact of metastability and tolerate process variations. Subsequently, proposed of circuit level-techniques were investigated to decrease these impacts. Lastly, synchronization schemes between multiple voltage and multiple clock domains were investigated.

1.4 Thesis Contributions

In this thesis, a number of techniques have been proposed, to enhance the performance and robustness of synchronizers and arbiters. The study has focused on the impact of variations of PVT on the performance of circuits in terms of propagation delay time, metastability resolution time, power consumption. The contributions made to the state of the art in this area of research are outlined below:

- Analysis of the trade-off in Flip-Flops when used as registers or used in a synchronizer application. Flip-flops used to store a bit in a register have different requirements to flip-flops used in a synchronizer application. The data input must be held stable during the setup and until the Q output appears, these times determine the remaining part of the clock cycle available for computing. On the other hand, the data input can violate setup and hold times in a synchronizer, and the reliability of the synchronizer depends on the metastability recovery time constant. This work shows how these parameters can be traded off in a simple edge triggered D flip-flop and other cells. (Published IEEE ICECS 2009)

- The improvement of reliability and latency through the adoption of the Wagging synchronizer design. An alternative structure to the conventional two-flip-flop synchronizer is proposed based on the Wagging principle. The aim of the wagging synchronizer is to increase the time allowed for metastability to resolve, hence improve the synchronizer reliability. (Published IEEE ICM 2010).

- Modifications of the MUTEX circuit to improve the metastability resolution time and tolerance to the effects of process variation. The MUTEX study focused on the classical circuit, in addition to a number of modifications at the circuit and transistor levels, to adjust the value τ and its tolerance, based on increasing internal gain by adding current sources, reducing the capacitive loading, boosting the transconductance of the latch, compensating the existing Miller capacitance, and adding asymmetry to maneuver the metastable point. The results showed that some circuits had almost little or no improvements, while five techniques showed significant improvements by reducing τ and maintaining high tolerance.

- The design of robust and reconfigurable wagging synchronizer to improve reliability and latency tolerance to the effects of process variation. The robustness of the wagging technique can be enhanced by using robust τ latches or adding one more cycle of synchronization by a reconfigurable circuit.
- The design of Metastability Auto-Detection and Correction (MADAC) latches that are able to detect and correct metastable events within the cell. This approach relies on swiftly detecting a metastable event and correcting it by enforcing the previously stored logic value. This technique significantly reduces the resolution time uncertainty.
- Finally, the design of pseudo Level-Shifting Handshake Synchronization (LSHS) techniques, which are able to operate between multiple-voltage multiple-clock domains that do not require conventional level-shifters between the domains or multiple power supplies within each domain. This interface circuit uses a synchronous set and feedback reset protocol which provides level-shifting and synchronization of all signals between the domains, from a wide range of voltage-supplies and clock frequencies.

1.5 Thesis Organization

The content of this thesis is organized in total seven chapters. Following the introduction:

In **Chapter 2**, a background study of the metastability behavior in bistable elements and synchronizers are presented. Followed by, studies of synchronizers and arbiters. Then, reviews of the synchronization techniques on-chip and the multiple voltage design are presented. At the end, the performance metrics of flip-flops and how to obtain them are defined.

In **Chapter 3**, the analysis of the trade-off in Flip-Flops when used as registers or used in a synchronizer application is discussed. Then, the wagging synchronizer design is described, accompanied by its results showing improvements of reliability and latency.

In **Chapter** 4, variation-tolerant arbiter design is presented. It focuses on modifications of the MUTEX circuit to improve the metastability resolution time

and tolerance to the effects of process variation design modification of the conventional. Then, simulation results, showing the impact of gate-size and load-size, as well as, PVT variations on τ and delay time, are presented.

In **Chapter** 5, two variation tolerant design techniques for synchronizers are presented. First, the robust and reconfigurable wagging synchronizers to improve reliability and latency tolerance to the effects of process variation are presented. Then, the MADAC synchronizers that are able to detect and correct metastable events within the cell. Followed by simulation results of the MADAC against PVT variations on timing and power parameters are presented.

In **Chapter** 6, a multi-voltage synchronization technique is presented. First, the dual and single-supply level-shifting latch is discussed. Then, the LSHS approach, followed by a modified approach LSHS2 to adapt to wider clock cycles range, are presented. Then, a bidirectional LSHS approach is proposed.

Finally, **Chapter** 7 comprises the results discussion and thesis conclusions and outlines the future prospective research opportunities.

Chapter 2 Background

The continued scaling of semiconductor technology creates the potential of SoC integration, that is, the integration of a complete electronic system, including interfaces to the outside world on a single die. An SoC consists of several mixed components with different implementation styles such as programmable processors, dedicated hardware to perform specific tasks, on-chip memories, input-output interfaces, and on-chip communication architecture that serves as the interconnection fabric for communication between these components.

This scaling of integrated circuits has been a challenging step in the industry to maintain an acceptable production yield. Chip scaling provides the opportunity to reduce area, cost and power consumption and improve speed. Yet, its drawbacks must be realized. Manufacturing variations can cause deterioration in the chip's performance and functionality and, consequently, in the production yield. Parameter variability is expected to increase with every new technology node and significantly increase the effects on circuit performance, in terms of power consumption and delay [1, 14, 45].

Multiple Systems-on-Chip are designed with more asynchronous circuits and techniques rather than synchronous ones to tolerate the variations at the transistor level. Some of these techniques use arbiters and synchronizer as the interface block between modules. Arbiters recognize the order of events, and synchronizers help to receive data from other synchronous or asynchronous domains. They have been used frequently in VLSI systems and architectures, such as GALS wrappers for a network-on-chip (NoC) [58-61] or network adapters and routers [61-64]; and they will be utilized even more in the future by the increase in asynchronous global signaling being exploited within a single SoC, which is predicted to double by 2016 [1].

In the subsequent sections of this chapter a background study of the metastability behavior along with the current synchronizer and arbiter circuits are discussed, as well as their design challenges. After that, different on-chip synchronization

techniques are discussed including a single global-clock and multiple-clocks, asynchronous techniques, and GALS. Then, multiple-voltage domain approaches and concerns are discussed. At the end, the metrics for characterizing the performance, particularly of flip-flops, are defined.

2.1 Metastability, Synchronizers and Arbiters

In a synchronous system, data signals always meet input timing requirements of flip-flops, because the relationship between data and clock is fixed; therefore, metastability does not occur. Nevertheless, in most multiple clock systems and asynchronous systems, input data regularly violates the setup and hold timing conditions of bistable elements, because input data, clock switching rates and phase relationship is inconsistent. This violation results in delayed output signals and possibly leads to metastable outputs, which add further delays to produce a valid and stable output value, logic '1' or logic '0'. Therefore, it is important to carefully analyze and design bistable elements prone to metastability for minimum metastability time without impacting on performance. Figure 2.1 shows the circuit diagram of a data latch and a timing diagram with clock, input data D and output value Q. The input data D transits within the setup and hold region around the clock rising-edge results in a metastable output Q, which may last a period of uncertain time.

Metastability is a hazardous anomaly phenomenon that can take place in any bistable or sequential circuit; particularly, more often, in synchronizers and arbiters. It is known to be an unstable equilibrium voltage point between the valid voltage logic levels (0 and V_{DD}) and usually around $½V_{DD}$. This voltage point is equivalent to the middle voltage which is the switching/inverting point of the gates comprising the bistable circuit. If the bistable circuit has a long feedback path, then metastability develops into an oscillation around the middle voltage, which was observed in some obsolete technologies, for example a set-reset latch comprised of TTL NAND gates [29] and CMOS NOR gates with buffered output either off chip [65] or on chip [66].

An analogy to a bistable latch is a ball transiting over a hill having two stable points at either side of the bottom of the hill and one metastable point at the top of the hill, as shown in Figure 2.2. If the ball transition force is not enough to cross the

hill, then it will fall back to the bottom. However, if it is enough to cross the hill, the ball will fall to the other stable point, whereas if the force is only sufficient to reach the top of the hill, then the ball will stay still unless there is a disturbance in the environment due to wind for example.

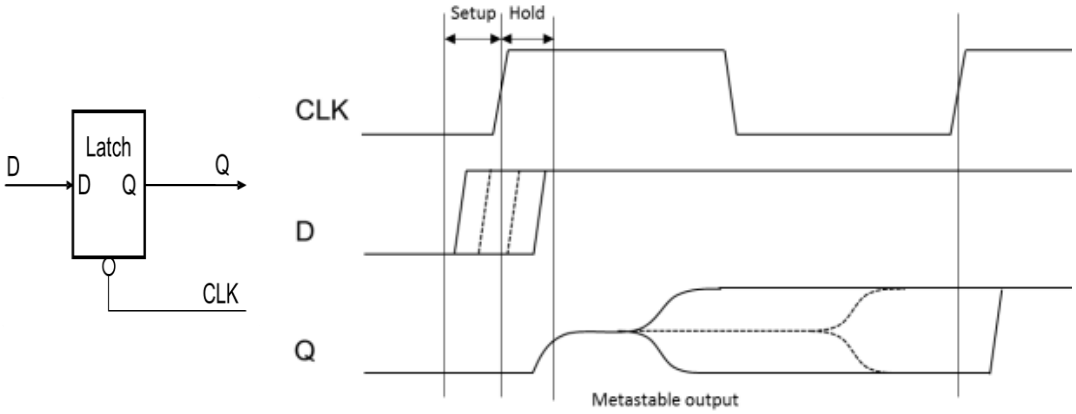


Figure 2.1 Data transition violation causes metastable output Q

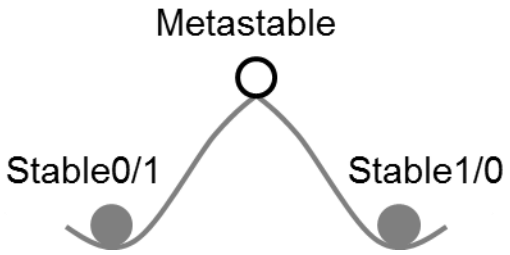


Figure 2.2 Metastability analogy to ball over a hill

The main cause of recurring metastable events is the conflict between incoming signals with the timing restrictions. Likewise, metastability may be initiated by the resolution in a preceding sequential stage violating the next stage's timing conditions, which is known as the back edge of the clock effect [50]. For example, if a master latch in a master-slave flip-flop exhibits metastability that holds the master latch a long time and resolves near the slave-latch timing condition at the back edge of the clock, this may initiate a new metastability event in the slave-latch needing more time to recover. Furthermore, metastability may be transferred between logic gates or from master latch to slave latch if not designed properly. Metastability may also occur due to a very short pulse gated clock or even a poorly timed clear or reset signal [50]. Moreover, on the occasion of a single event upset due to alpha particle strikes, a current spike could last a sufficiently long time to flip a cell or induce metastability.

However, this thesis only concentrates on metastable events that are caused by

asynchronous input signals from an asynchronous system or a differently clocked system because it is significantly the most frequent recurring cause of metastability, and it is considered one of the most difficult problems to deal with in synchronization.

In case of metastability at the output of a latch driving some logic stage, the subsequent logic stage will behave unpredictably, and some may interpret this invalid voltage level as a logic one while another as a logic zero. As a result, metastability may produce failures appearing as data being lost, corrupted or duplicated, which causes a system failure and in particular circumstances a system deadlock.

A simple latch circuit modeled in UMC 90nm process technology and simulated in SPICE-level. The latch simulation waveforms, shown in Figure 2.3, demonstrate the latch going metastable at different data arrival times, stepped at 1ps closer to the falling edge of the clock.

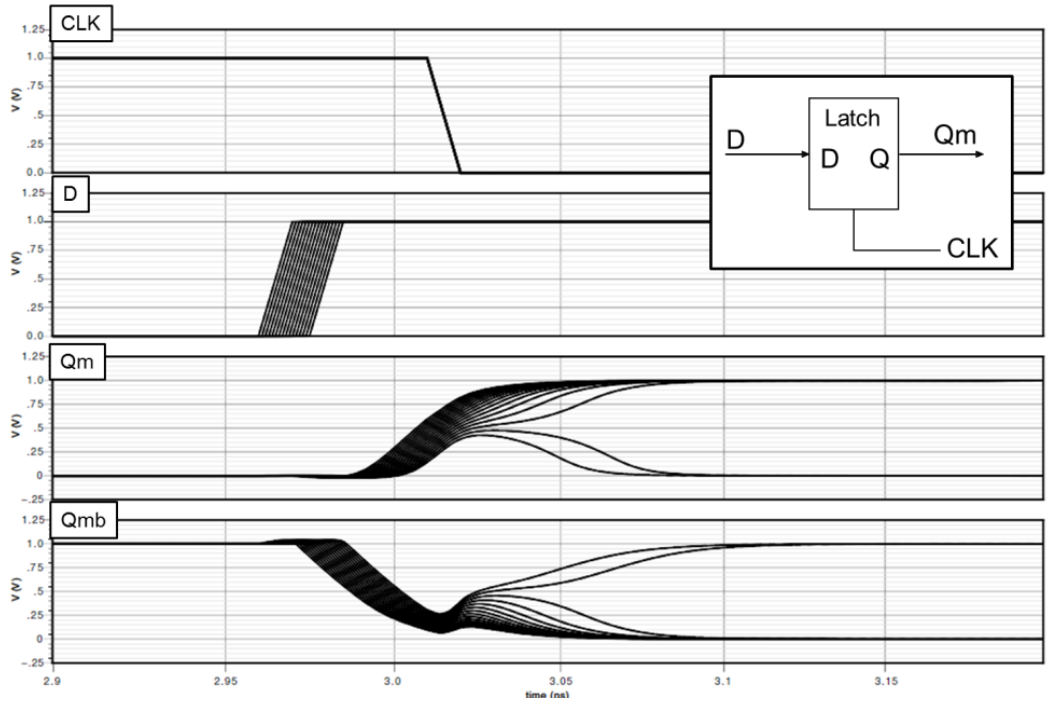


Figure 2.3 Simulation waveforms of a latch going metastable

In a noise-free environment, if metastability in the latch is at the exact balance point, it could remain at this point for until the next clock edge. Whereas, the existing noise in the environment may assist to resolve metastability faster. The existing negligible numerical deficiency in the simulator acts as numerical noise between the potential difference between Qm and Qmb. The latch may spend a

stretched period of time virtually balanced at an unstable point of equilibrium between the two stable states. Since the probability of staying at that balance point approaches zero, in theory, there is a low probability this will happen. If the resolution time is reduced due to increasing clock frequency, there is a growing probability that the latch will remain metastable until the next clock edge.

In a multiple clock system, metastability is unavoidable, but there are several design techniques to reduce the chance of failures due to metastability. This section provides an overview of synchronizers and metastability behavior and analysis in bistable elements in general, followed by metastability impact reduction techniques, and synchronizers performance and circuits.

2.1.1 Metastability Behavior Analysis

A simple latch comprising two back to back symmetric inverters, shown in Figure 2.4 below, will be used to explain and analyze the nature of metastability.

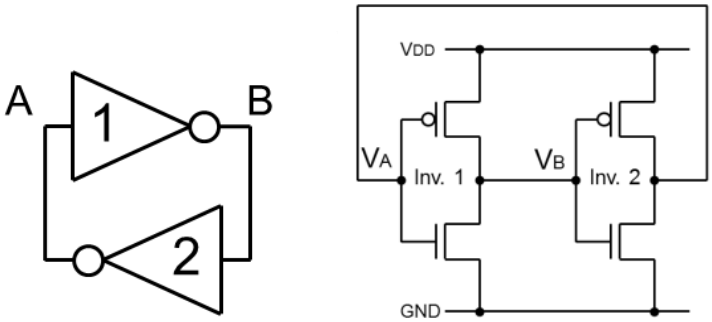


Figure 2.4 Cross-coupled inverters

2.1.1.1 Large Signal Analysis

Large signal analysis determines the metastable DC voltage level of the latch shown in Figure 2.4. The DC voltage-transfer characteristics of both inverters can be superimposed on each other as shown in the graph in Figure 2.5. This graph shows three intersection points, two stable ones at the sides and an unstable one in the middle. The stable points signify the inversion of voltage A and voltage B from 0 to V_{DD} and *vice versa*. For instance, regarding inverter 1, stable0/1 is for inversion from low to high, and the opposite for stable1/0. The middle point denotes the unstable switching voltage V_m at which inversion occurs and metastability upheld. During the metastability event the power consumption is increased because all the latch transistors are actually turned on, creating a short circuit path from the voltage supply to the ground and continuously draws a large

current until that event resolves.

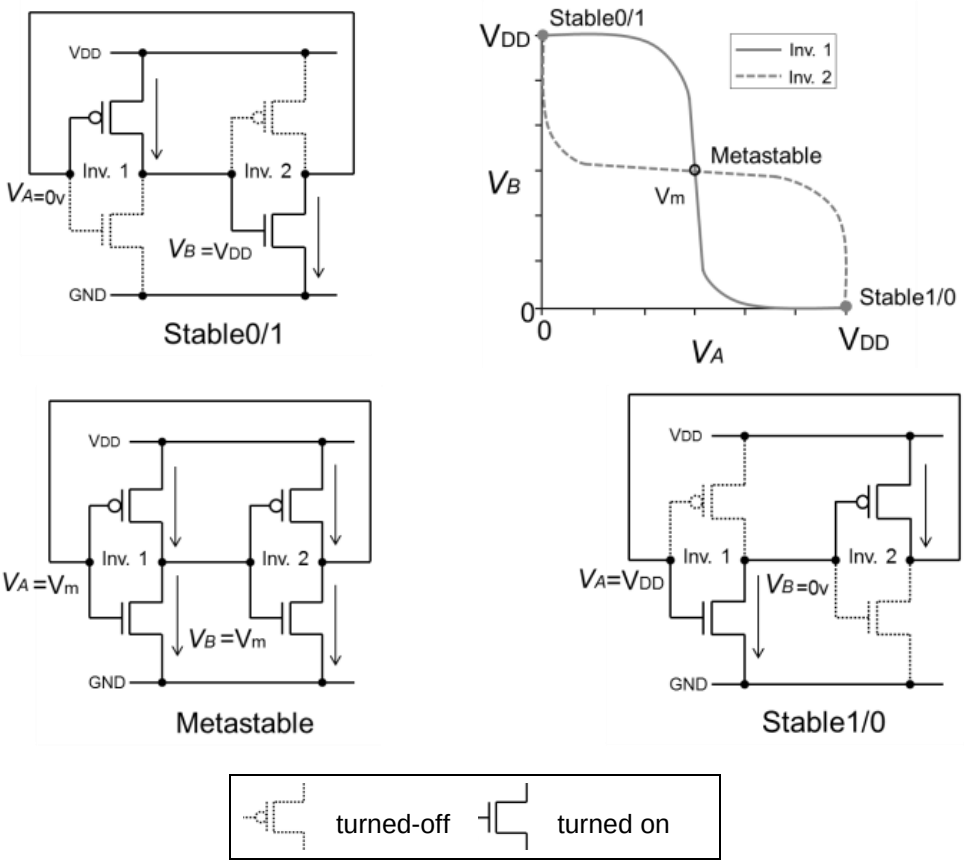


Figure 2.5 Intersecting point of voltage transfer characteristics of the latch

The metastable DC voltage level can be computed using the switching voltage formula of the inverter [4, 5, 67] as shown in Equation (2.1). The mid-voltage point is considered the balanced point of equilibrium, as depicted by the bell shape in Figure 2.2, if both latch nodes reach the mid-voltage point simultaneously, this diminishes the push and pull force of the inverters.

$$\left.\begin{aligned} V_m &= \frac{V_{DD}+r_m\times V_{THn}-|V_{THp}|}{1+r_m} \\ r_m &= \sqrt{\frac{W_n\times\mu_n}{W_p\times\mu_p}} \\ r_m &= \frac{W_n\times v_{sat,n}}{W_p\times v_{sat,p}} \end{aligned}\right\} \quad (2.1).$$

From Equation (2.1), r_m is the ratio of NMOS transistor to PMOS transistor. This ratio r_m is an important factor to define the inversion and metastable point to a lower or a higher voltage. For instance, the inverters have all transistors with similar absolute threshold voltages and, if their ratios are equal to one, the metastable level is $V_{DD}/2$, whereas, ratios above 1, lowers the metastable point

below $V_{DD}/2$, and ratios smaller than 1, lifts the metastable point over $V_{DD}/2$, as shown by the white circles in Figure 2.6. Since the process parameters μ_n/μ_p and $v_{sat,n}/v_{sat,p}$ are technology dependent then the only design parameters available to skew the inversion point is transistors width ratio W_n/W_p. In the case where the inverters' ratios were asymmetric, then the inversion point for each inverter will be different and the intersection may take place at one the grey circles shown in Figure 2.6.

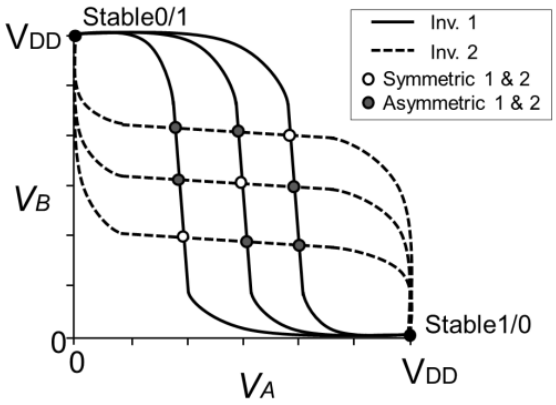


Figure 2.6 Symmetry and asymmetry between ratios of the two inverters

2.1.1.2 Small Signal Analysis

The small signal analysis determines the time dynamics of metastability behavior in the latch to characterize the length of an event. Literature outlines [50, 68-72] two different models to analyze the timing behavior of metastability. It is achieved under the assumptions that nodes A and B are at the metastable DC level at time t = 0. A simple method verified in [50] represented each inverter gate by a linear amplifier model composed of a voltage amplifier with gain $-A$, in series with a resistance and a capacitance. Another simple model appeared in [69] included second order effects and based on a two-port transconductance amplifier with an output resistance, an output capacitance and a Miller capacitance. The Miller effect accounts for the 'Miller' capacitance between the input and the output and the gain of the amplifier. The Miller capacitance is the sum of the gate-to-drain capacitances of the PMOS and NMOS transistors in an inverter.

Based on the circuit shown in Figure 2.7, the output capacitance, output resistance, transconductance and Miller capacitance of inverter 1 are computed as in Equation (2.2), and similarly for inverter 2.

$$\left.\begin{aligned} C_1 &= C_{GS1n} + C_{GS1p} + C_{GB1n} + C_{GB1p} + C_{DB2n} + C_{DB2p} + C_{Load,A} \\ R_1 &= R_{O1n} \| R_{O1p} \| R_{Load,B} = 1/g_{out1} \\ g_{m1} &= g_{m1n} + g_{m1p} \\ C_{M1} &= C_{GD1n} + C_{GD1p} \end{aligned}\right\} \quad (2.2).$$

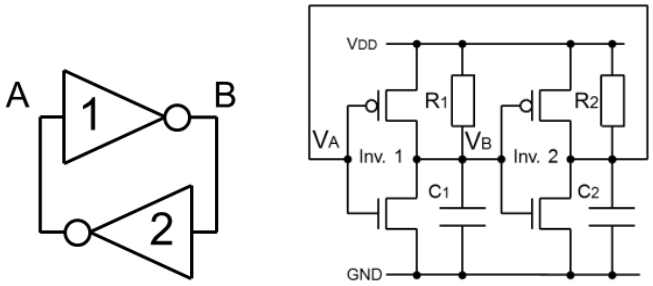


Figure 2.7 Cross-coupled inverters parasitic capacitance and resistance

The evaluation of the absolute voltage gain $|A_1|$ and bandwidth $\omega_{3dB,1}$ of inverter 1 can be derived in a similar manner to that of the push-pull inverting amplifier based on [73]. They are estimated using Equation (2.3) as well as the Gain-Bandwidth Product (GBP).

$$\left.\begin{aligned} |A_1| &= g_{m1} \cdot R_{out1} \\ \omega_{3dB,1} &= \frac{1}{R_{out,1} \cdot (C_{M1} + C_{out,1})} \\ GBP_1 &= |A_1| \cdot \omega_{3dB,1} = \frac{|A_1|}{R_{out,1}(C_{M1} + C_{out,1})} = \frac{g_{m1}}{(C_{M1} + C_{out,1})} \end{aligned}\right\} \quad (2.3).$$

a) Linear Amplifier Latch Model

The voltage amplifier model of the latch depicted in Figure 2.8 is analyzed in the following system of differential equations expressed in Equation (2.4).

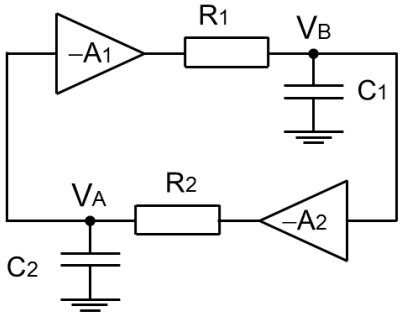


Figure 2.8 Cross-coupled inverters linear amplifier model

$$\frac{A_2}{R_2}\cdot V_A+\frac{1}{R_2}\cdot V_B+C_2\cdot\frac{dV_A}{dt}=0$$
$$\frac{A_1}{R_1}\cdot V_B+\frac{1}{R_1}\cdot V_A+C_1\cdot\frac{dV_B}{dt}=0 \quad (2.4).$$

Suppose the inverters parameters are identical; that is C_{out}=C_1=C_2, A=A_1=A_2, and R_{out}=R_1=R_2, and the inverters have high gain ($A\gg1$), then results in Equation (2.5), which is equivalent to GBP in Equation (2.3).

$$\therefore\frac{1}{\tau}=\frac{A-1}{R_{out}\cdot C_{out}}\approx\frac{A}{R_{out}\cdot C_{out}}\Rightarrow\tau=\frac{R_{out}\cdot C_{out}}{A}=\frac{C_{out}}{g_m} \quad (2.5).$$

Solving the equations for the differential-mode voltage V_{DM}= V_A-V_B ;

$$-\frac{A-1}{R_{out}}\cdot V_{DM}+C_{out}\cdot\frac{dV_{DM}}{dt}=0$$
$$V_{DM}=V_{dm0}\cdot exp\left(t\cdot\frac{(A-1)}{R_{out}\cdot C_{out}}\right)=V_{dm0}\cdot e^{t/\tau_{dm}} \quad (2.6).$$

Solving the equation for the common-mode voltage V_{CM}= (V_A + V_B)/2;

$$\frac{A+1}{R_{out}}V_{CM}+C_{out}\frac{dV_{CM}}{dt}=0$$
$$V_{CM}=V_{cm0}\cdot exp\left(-t\cdot\frac{(A+1)}{R_{out}\cdot C_{out}}\right)=V_{cm0}\cdot e^{-t/\tau_{cm}} \quad (2.7).$$

The voltage at node A can be written as:

$$V_A(t)=2V_{CM}+V_{DM}=2V_{cm0}\cdot e^{-t/\tau_{cm}}+V_{dm0}\cdot e^{t/\tau_{dm}} \quad (2.8).$$

The values of V_{dm0} and V_{cm0} are determined from the initial conditions before metastability is initiated. This model is only valid within the linear region around the metastable level. The common-mode voltage is an exponentially decaying term that diminishes quickly and can be ignored, whereas the differential-mode voltage is an increasing exponential term representing the response.

b) Miller Effect Latch Model

Considering the second-order small-signal model of the latch shown in Figure 2.9 and using nodal analysis to find the node voltages V_A and V_B gives the following equation.

$$\frac{1}{R_1}\cdot V_B + (C_1 + 2C_{M1})\cdot\frac{dV_B}{dt} + g_{m1}\cdot V_A - 2C_{M1}\cdot\frac{dV_A}{dt} = 0$$
$$\frac{1}{R_2}\cdot V_A + (C_2 + 2C_{M2})\cdot\frac{dV_A}{dt} + g_{m2}\cdot V_B - 2C_{M2}\cdot\frac{dV_B}{dt} = 0 \quad (2.9).$$

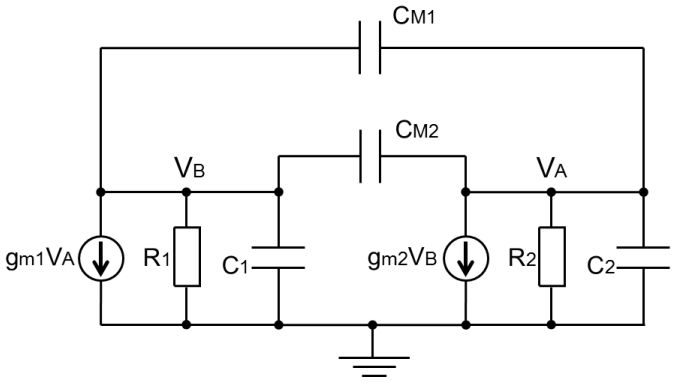


Figure 2.9 Cross-coupled inverters transconductance model with Miller-effect

Assuming symmetric inverter parameters, namely C_{out}=C_1=C_2, g_m=g_{m1}=g_{m2}, C_M=C_{M1}=C_{M2} and R_{out}=R_1=R_2, and supposing that the transconductance at the metastable level is much greater than the output conductance and the output capacitance is greater than the Miller effect, that gives a similar time constant value to the one in Equations (2.3) and (2.5).

$$\therefore \frac{1}{\tau} = \frac{g_m - 1/R_{out}}{C_{out}+4C_M} \approx \frac{g_m}{C_{out}} \Longrightarrow \tau = \frac{C_{out}+4C_M}{g_m - 1/R_{out}} \approx \frac{C_{out}}{g_m} \quad (2.10).$$

Solving the equations for the differential-mode voltage V_{DM}= $V_A - V_B$;

$$-(g_m - 1/R_{out})\cdot V_{DM} + (C_{out} + 4C_M)\cdot\frac{dV_{DM}}{dt} = 0$$
$$V_{DM} = V_{dm0}\cdot exp\left(t\cdot\frac{(g_m - 1/R_{out})}{C_{out}+4C_M}\right) = V_{dm0}\cdot e^{t/\tau_{dm}} \quad (2.11).$$

Solving the equation for the common-mode voltage V_{CM}= (V_A + V_B)/2;

$$(g_m + 1/R_{out})V_{CM} + C_{out}\frac{dV_{CM}}{dt} = 0 \quad (2.12).$$
$$V_{CM} = V_{cm0}\cdot exp\left(-t\cdot\frac{(g_m + 1/R_{out})}{C_{out}}\right) = V_{cm0}\cdot e^{-t/\tau_{cm}}$$

The voltage at node A can be written as:

$$V_A(t) = 2V_{CM} + V_{DM} = 2V_{cm0}\cdot e^{-t/\tau_{cm}} + V_{dm0}\cdot e^{t/\tau_{dm}} \quad (2.13).$$

In contrast to Equation (2.6), the Miller capacitance has a significant effect on the differential-mode voltage in Equation (2.11), because it increases the time

response constant and in turn increases the time required for the absolute V_{DM} to increase beyond metastable region.

2.1.1.3 Failure Rate and Mean-Time-Between-Failures

The probability of a flip-flop being metastable for some time t_R or longer is equivalent to the probability of entering metastability times the probability of exiting it [5, 46, 50]. Firstly, the probability that the flip-flop will enter metastability, if input data and clock transitions occur close together within a time window of T_w (metastability window), and under the assumption of uncorrelated average switching frequencies f_d and f_c of the input data and clock signals, is equivalent to $P(m|_{t=0}) = T_w \cdot f_d \cdot f_c$. Secondly, the probability that the flip-flop will exit metastability after time t_R is equal to $P\left(m|_{t=0\rightarrow t_R}\right) = e^{-t_R/\tau}$, where τ is metastability recovery time constant, which indicates the strength and speed of a flip-flop resolving metastable events. Therefore, the product of these probabilities defines the failure rate of a flip-flop, and the inverse of the failure rate is the Mean Time Between Failure (MTBF) [29, 46, 50, 70] as shown in Equation (2.14). The equation for the MTBF is an important figure of merit to assess the reliability of flip-flops to operate as synchronizers. This equation has been confirmed in theory and by simulations and experiments in [50, 74-79] and improved in [76].

$$Failure\ rate = P(m|_{t=0}) \cdot P\left(m|_{t=0\rightarrow t_R}\right) = \frac{1}{MTBF}$$
$$MTBF = \frac{e^{t_R/\tau}}{T_w f_d f_c} \qquad (2.14).$$

In general, a flip-flop metastable failure occurs when an input data transition violates the setup or hold times of the flip-flop. The failure rate of a flip-flop is not a guaranteed matter; it is only a good estimation of the reliability of a flip-flop based on the probability of input violations and the probability of resolving the metastability.

2.1.1.4 Metastability Behavior with Technology Scaling

Generally, metastability behavior is a function of process technology and environment, because as process technology scales down the metastability resolution time decreases [29, 71, 80, 81]. This is because the metastability resolution time is directly proportional to capacitance, which reduces with scaling down, and inversely the gain-bandwidth product increases with scaling down. In a

similar manner to the propagation delay, metastability resolution time increases significantly with reduction in the nominal supply voltages [80, 82, 83] and increased load capacitance [74], especially under low temperatures [82]. This is because low supply voltage reduces the drain current and hence reduces the transconductance. Low temperatures shift the threshold voltage up which, in turn, reduces the current as well. On the other hand, increasing the load capacitance adds more demand on charge to be supplied by the drain current [84]. Process parameter variations have a considerable impact on metastability time response and window [40, 83, 85, 86].

To observe the metastability dependence on technology in an inverter-based latch, the metastability resolution time constant τ Equation (2.10) is further broken down to consider the process parameters associated with identical load inverters equivalent to a latch, with the assumption that the dominant parasitic capacitance is the gate-to-source capacitance, then the Miller capacitance and output resistance can be considered to be negligible. Equation (2.15) shows the dependence of τ on the channel length L of transistors and the saturation velocity v_{sat}.

$$\begin{aligned} \tau &\approx \frac{C_{out}}{g_m} = \frac{2(C_{GSn}+C_{GSp})}{g_{mn}+g_{mp}} = \frac{\frac{4}{3}(W_n+W_p)LC_{ox}}{W_nC_{ox}v_{sat,n} + W_pC_{ox}v_{sat,p}} \\ \Rightarrow\ \tau &= \frac{(W_n+W_p)C_{ox}}{(W_n+W_p)C_{ox}} \cdot \frac{4L}{3v_{sat}} \\ \therefore\quad \tau &= 4L/3v_{sat} \end{aligned} \tag{2.15}$$

Let τ, L, C_{out}, and g_m be parameters at a given technology node, and let τ', L', C_{out}', and g_m' be parameters of another technology node scaled by a factor of S. From Table 1.1 and Equation (2.2), Equation (2.10) and Equation (2.15), the scaled value of τ is derived in Equation (2.16) under the assumption that V_{DD} and temperature remain constant. The impact of scaling on the metastability resolution time constant is seen as if L for a given technology node has a time constant τ, then for another, that is at node L/S will have a time constant τ/S.

$$\begin{aligned} \tau' &\approx \frac{C_{out}'}{g_{m}'} = \frac{2(C_{GSn}'+C_{GSp}')}{g_{mn}'+g_{mp}'} = \frac{2(C_{GSn}+C_{GSp})/S}{(g_{mn}+g_{mp})\cdot S/S} = \left(\frac{4L}{3v_{sat}}\right)/S = \tau/S \\ \Rightarrow \tau' &\approx 4L'/3v_{sat} = (4L/3v_{sat}\ )/S = \tau/S \end{aligned} \tag{2.16}$$

2.1.2 Metastability Impact Mitigation

To reduce the impact of metastability on bistable circuits, different resolution techniques are described in the following sections.

2.1.2.1 Latch Sizing and Loading

Under normal operating conditions, the strength of metastability in any latch is primarily dependent on the size of the latch and the total capacitive load it is driving. To resolve metastability faster, the latch needs to have stronger transistors and driving smaller loading capacitances. Also, the ratio of the transistors in the inverters contributes to the behavior of metastability.

From Equation (2.10), the metastability resolution time constant τ is an important factor in the flip-flop reliability. As shown previously in Section 2.1.1.2, the value τ of cross-coupled inverters was modeled and analyzed, which showed that τ is equivalent to the inverse of the gain-bandwidth-product of the cross-coupled inverters at the metastable DC level, which is approximated to the total node output capacitance plus the Miller capacitance and all divided by the total transconductance of the logic gate. Assuming the load inverter size is equivalent to the latch inverter size times a ratio α_{LL} , defined as the load width to latch width ratio (W_{Load}/W_{Latch}). Equation (2.17) below shows approximately how the effect of the load to latch size directly affects the value of τ. For instance, if the load becomes greater, then τ will be longer, and vice versa.

$$\tau \approx \frac{C_{out}}{g_m} = \frac{C_{GSn}+C_{GSp}+\alpha_{LL}(C_{GSn}+C_{GSp})}{g_{mn}+g_{mp}}$$
$$\Rightarrow \tau \approx \frac{\frac{2}{3}(W_n+W_p)\cdot LC_{ox}+\alpha_{LL}\frac{2}{3}(W_n+W_p)\cdot LC_{ox}}{W_nC_{ox}v_{sat,n}+W_pC_{ox}v_{sat,p}} = \frac{(W_n+W_p)C_{ox}}{(W_n+W_p)C_{ox}} \cdot \frac{2L\cdot(1+\alpha_{LL})}{3v_{sat}} \qquad (2.17).$$
$$\therefore \quad \tau \approx \frac{2L\cdot(1+\alpha_{LL})}{3v_{sat}}$$

In the case when a crossed-coupled inverter latch enters metastability, then the time needed to resolve its metastability is directly dependent on the value of τ. The larger value of τ, the longer the time that is needed for metastability resolution, and the smaller value of τ, the shorter the time.

2.1.2.2 Extending Resolution Time

In general, the available metastability resolution time is not a design factor in

synchronous systems and it is determined based on system requirement. In a single flip-flop, the available resolution time or settling time [4, 5, 50, 72, 74, 86] is mainly dependent on the remainder of the clock cycle T_C after subtracting the clock-to-output delay t_{CQ} of the flip-flop and the setup time t_{SU} of the following stage and any combinational logic delay in between, which can be interpreted as the "lost time". This is written in the following equation.

$$t_R = T_C - t_{Lost} = T_C - \left(t_{CQ} + t_{SU}\right) \quad (2.18).$$

In the context of using flip-flops as a synchronizer, the available metastability resolution time (t_R) becomes a design factor to improve the MTBF based on Equation (2.14). In order to design for a longer resolution time to do this, depending on the design requirements, there are three approaches based on Equation (2.18).

- First, the clock frequency may be reduced if the design specification is flexible or has wide timing margins.
- Second, the lost time may be reduced if replaced by flip-flops with faster output time delay.
- The last one would be to increase the number of cycles, by directly pipelining two or more flip-flops, without any logic insertions, to synchronize and increase the available metastability resolution time, then Equation (2.18) could be rewritten as in Equation (2.19); assuming N identical flip-flops.

$$t_R = (N-1) \cdot (T_C - t_{Lost}) = (N-1) \cdot (T_C - t_{CQ} - t_{SU}) \quad (2.19).$$

2.1.2.3 Metastability Filters

Metastability filters have been used to prevent metastability from progressing to the next stage. In general, they are placed just after the latch outputs and basically interpret the metastable levels as logic '1' or logic '0', which is the process of filtering. The simplest filter circuit is based on skewed-inverters that could have a low or high switching voltage (threshold) point V_T, namely high-V_T or low-V_T inverting filters. Based on the threshold point the filter passes metastable levels as

logic ‘0’ with low-V_T inverter or as logic ‘1’ with high-V_T inverter. This type of filter is custom designed and not available in FPGA or standard-cell library, but it can be formed in a standard-cell designs using a four-input NOR-gate with all four inputs connected to the latch or flip-flop output [50].

A full-custom metastability filter is commonly used as part of the MUTEX (Mutual-Exclusion) circuit, which is shown in Figure 2.15 and described later in Section 2.1.4.1. It is based on two subsequent inverters after the NAND gates output nodes, where each inverter has the PMOS source terminal connected to the input of the other inverter instead of V_{DD} to sense the potential voltage difference between the output nodes of the NAND gates. In case of metastability, the filter’s PMOS devices remain inactive because their absolute gate-to-source voltage is zero, as a result the output of this filter is held to logic zero until metastability resolves, that is the NAND gate output voltages diverge enough so that there is sufficient difference, more than the absolute threshold-voltage, to activate one of the PMOS devices, then eventually one inverter output will rise to a logic high.

An alternative metastability filtering circuit utilizes the hysteresis property of the voltage-transfer characteristic of Schmitt-Trigger inverters to filter out metastable levels, as presented in [72, 87]. The hysteresis is a shift in the threshold (inverting) voltage lower value towards zero volts if zero and higher towards V_{DD} if the output is already logic ‘1’. In the case of metastability, the Schmitt inverter sees the metastable level as the previous logic value, unless the metastable level passes its threshold point at that time.

The skewed-inverter filters are another metastability resolution scheme but are more sensitive to noise [50], however they can offer a faster transition during normal operation, unlike mutual-exclusion filters. Although they can tolerate noise, it introduces more delay [50, 76] because of its design requirements.

Generally, in synchronous circuits, the use of mutual-exclusion filters tends to have a number of drawbacks. For instance, they add more propagation delay to deal with metastability and may not resolve it rapidly. Also, they keep the outputs of bistable circuits clear from metastable levels. However, due to late resolved metastability, these outputs may violate the timing restrictions of the following sequential stage and initiate new metastability events, previously discussed as the back edge of the clock effect [50]. Although, mutual-exclusion filters let

metastability resolve arbitrarily to any value zero or one, the new output value is considered to be uncertain and therefore additional circuitry may be needed to provide channels for detecting errors and correcting them.

2.1.2.4 Transconductance Booster Feedback

An alternative technique focuses on improving the resolution time of metastability rather than filtering it, especially at low supply voltages and temperatures. This technique utilizes two voltage controlled current-sources, one on each output node of the latch. During the occurrence of metastability, both current-sources are switched on to increase the transconductance of the metastable latch and hence enhance the metastability resolving time constant. The voltage controlled current sources can be replaced by PMOS transistors, as used in the boost synchronizer [88, 89] and, the robust synchronizer [82], shown in Figure 2.13. This technique shows a great improvement towards low supply voltages and temperatures in comparison to a simple latch without a booster [83, 86].

2.1.2.5 Metastability Error Detection/Correction Feedback

An alternative method to deal with metastability is to use a metastability detector such as the one proposed in the Razor flip-flop [32], which was discussed earlier in Chapter 1. The metastable-detector is shown in Figure 2.10. In the Razor flip-flop, the node voltage of the slave latch is used to drive two skewed gates (one buffer and one inverter) in parallel connection to the inputs of an AND gate; the buffer is comprised of two inverters in series designed with large NMOS transistors to see metastable levels as a logic high input value, whereas the inverter has a large PMOS transistor observes metastability as a logic '0' input value. If both produce logic '1', then the AND gate produces logic '1' indicating that metastability has occurred and then flag an error signal. One drawback of this method is that if metastability in the slave latch resolves while the error signal is produced, which may cause a glitch in the error signal.

Another approach to detect metastability is by using a circuit to sense transition conflicts between the clock and the input data before it arrives to the input of the master latch. This is known as the transition-detection approach, and is applied in Razor II flip-flop [36] and in TDTB latch [32], which were shown previously in Figure 1.8(c) and Figure 1.8(d) in Chapter 1. The idea behind this approach

basically comes from locating the transitions of both input signals (clock and data) that coincide in time early enough before this causes any faults progressing in the system. Then, if it senses that a flag is set high, the flawed signal is either dropped-out or corrected based on a stored value in the shadow latch or flip-flop.

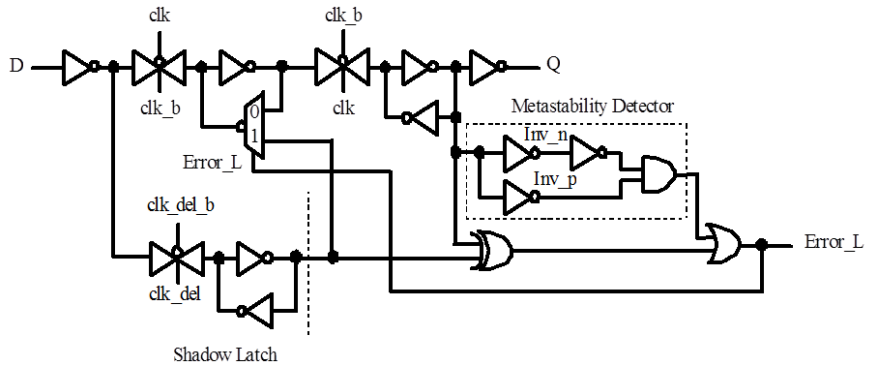
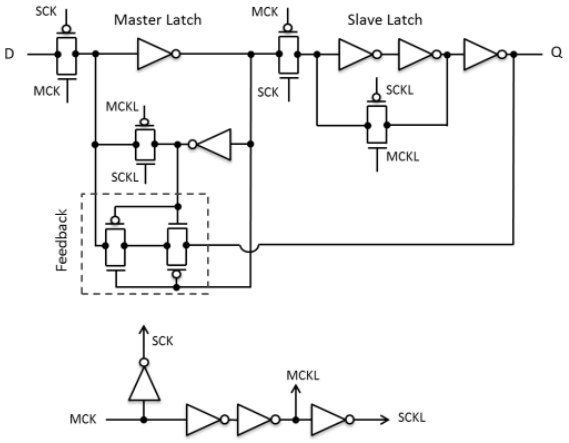


Figure 2.10 Metastability detection in Razor FF [33]

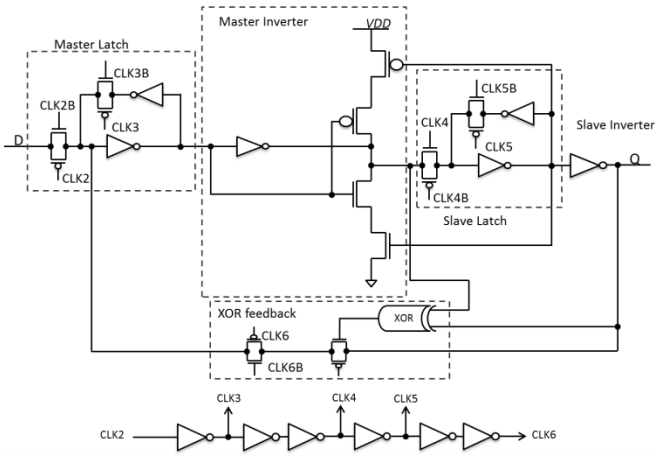
Another technique relies on detection and correction of metastability events in the master latch of a flip-flop using fast combinational logic “to detect” these events, whereupon a connection path is opened between the first node of the master latch and the output signal Q, as depicted in Figure 2.11, (or a stable known value) within the flip-flop cell so as to pull-down or push-up that metastable event depending on the state of Q, in other words, it does so to correct it to the previous value. This technique [90, 91] shortens the metastability life in the master latch.

The circuit proposed in [90], shown in Figure 2.11(a), uses a feedback path using two Transmission-Gates (TG) that are controlled by both nodes of the master latch, and conducts only if the nodes are both at the middle voltage state. For instance, if metastability occurs, the feedback path opens between the output of the flip-flop and the metastable node, which forces it into a stable state similar to the state of the output. The other technique, offered in [91] and shown in Figure 2.11(b), also uses a feedback path created from two TGs; one conducts at the negative clock (when the slave latch is transparent), and the other is controlled by an XOR gate comparing the flip-flop output and the second node in the master latch. This approach has not been popular due to its design requirement. Overall, as mentioned before, the main drawback of the metastability detection followed by another task, such as correction in the last two flip-flop circuits, is that the metastability occurring in the master latch may resolve during the detection or the correction process, which may lead to a new conflict between the resolving

metastability and the forced correction value.



(a) TG feedback [90]



(b) XOR+TG feedback [91]

Figure 2.11 Flip-flop with internal detection and correction feedback

2.1.3 Synchronizers

Synchronizers are necessarily employed to synchronize an asynchronous signal with a clock at the interface between asynchronous and synchronous domains. It is also needed to pass signals between two synchronous domains, which have different frequencies or phases from the other, or both, where it retimes the transition of the arriving Async data signal from the sending domain with the clock frequency of the receiving domain. Otherwise, without using a synchronizer the arriving signals will violate the timing conditions of the next logic circuit and induce a metastability failure in the reset of the system. The main purpose of a synchronizer is to provide enough time for any metastable output to resolve and settle down to a stable logic state at the receiving domain. Common synchronizers

are composed of master slave positive edge triggered D flip-flops [50, 74].

In a flip-flop synchronizer, the propagation delay time to the output Q may take longer if the transition edge of the input signal D and the latching or triggering edge of the clock are very close to each other, but the synchronizer has to make a decision within less than a clock period. At some point between the clock and data signals for the synchronizer, the latch output nodes are drawn to the metastable level, which stops the synchronizer from making any decision. If metastability persists in the synchronizer and exceeds its time constraint, the synchronizer will fail and could cause a system failure [29, 50, 70, 74]. That is why metastability directly affects the reliability of the synchronizer, which has driven researchers to further investigate alternative more reliable synchronizers with different bistable circuits and different clocking mechanisms.

Typically, designers would use a synchronizer composed of Two-Flip-Flop (2FF) in series, as shown in Figure 2.5, to provide enough resolution time, however it requires two clock cycles, as described previously in this section. Figure 2.6 shows an example of the operation of a 2FF synchronizer. The synchronizer reads the Async data at the first positive clock edge, then writes it to the output at the following edge, except if it arrives within the forbidden region, for example, transition number 3, it will either be written to the output on the following second or third edge of the clock.

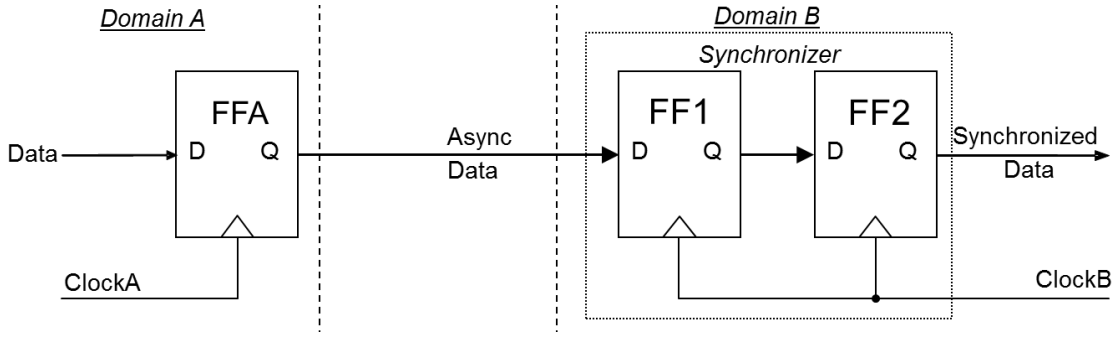


Figure 2.5 A Two-Flip-Flop (2FF) synchronizer

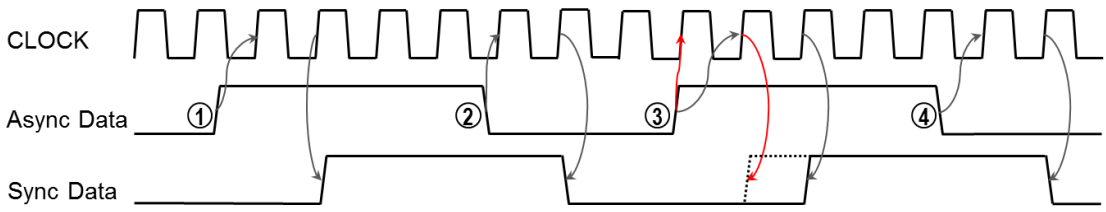


Figure 2.6 2FF synchronizer operation example

2.1.3.1 Synchronizer Circuits

a) Flip-Flops

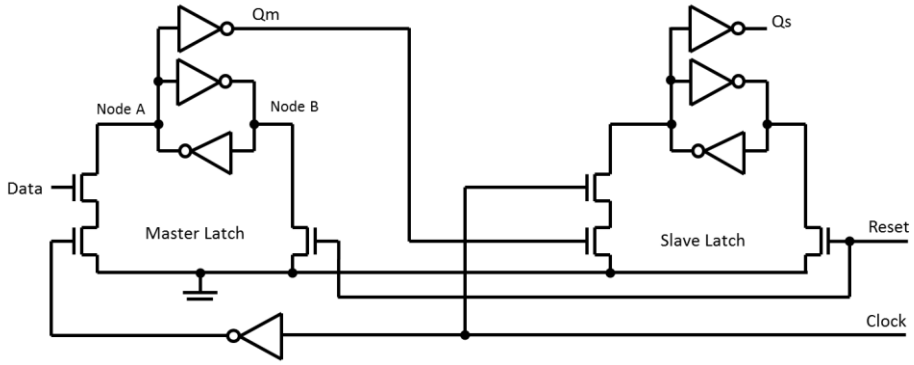
Different flip-flop circuits can be used as storage elements or synchronizers. In storage elements, flip-flops need to set and reset quickly and correctly, that is shorter clock to Q time and small setup plus hold times, whereas in synchronizers they should have a small resolution response time and a small metastability window. These conditions and other aspects, like power and variation-tolerance, determine which flip-flop design is better in any given application, but sometimes a compromise is required. For example, the transmission-gate (TG) flip-flop is the fastest flip-flop and very suitable for low power applications [85, 92]. However, it has the worst hold time variability against process variations compared to other static flip-flops [85].

A unique class of flip-flops, known as Dual-Edge Triggered (DET) flip-flops, exploit both clock transitions and store and hold data for half the frequency of a Single-Edge Triggered (SET) flip-flops and save even more power for the same duty cycle [93, 94]. DET flip-flops have not been investigated as synchronizers in the literature.

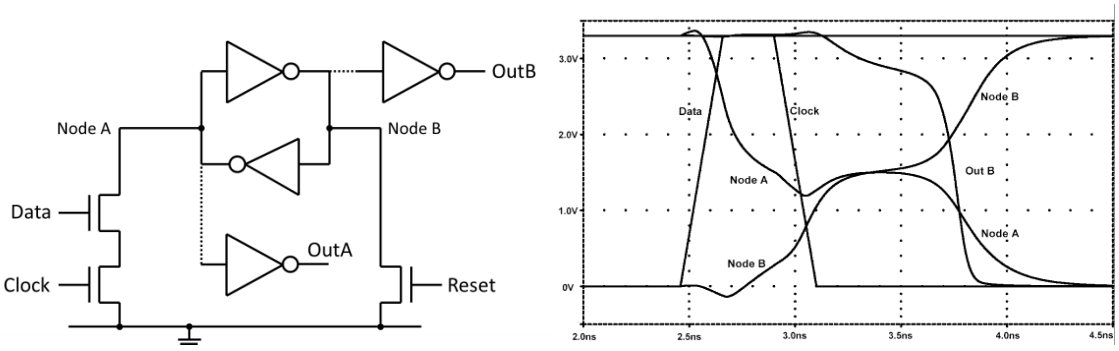
b) Jamb Latch and Flip-Flop

One unique flip-flop used as a synchronizer is the Jamb latch flip-flop. It comprises two similar Jamb latches as master and slave latches; where each is constructed from two large cross-coupled inverters, three NMOS transistors to gate the input data, clock and reset signals, and a small output inverter taken from either node (A or B), as shown in Figure 2.12. This latch structure provides higher gain and lower load on the feedback loop nodes [74, 76]. The Jamb flip-flop can be set when data and clock signals are high, which shorts node A to ground, and when the reset signal is high node B is shorted to ground and the latch reset. The output nodes are buffered using low threshold inverters to filter metastability levels. In this flip-flop, there is only one way in which metastability can happen, namely it occurs in the master latch only if the input data signal rises from logic ‘0’ to logic ‘1’ within the setup region near the clock edge, in this case node A will be slowly pulled down and node B slowly pulled up, which may reach a metastable level or resolve to stable levels [76]. This circuit is mentioned several times across this thesis to address some the differences to the proposed techniques as in Chapter 5 and to

introduce new application as in Chapter 6.



(a) Jamb Flip-Flop [74]



(b) Jamb Latch and metastability waveforms [76]

Figure 2.12 Jamb Latch and Flip-Flop circuits

c) Latch with τBoost

As typical latches, the Jamb latch shows poor reliability under low supply voltages and low temperatures, due to metastability resolution time dependence on voltage supply and temperature. One method to improve the metastability response of a Jamb latch synchronizer was proposed in [82], shown in Figure 2.13 and referred to as “the Robust synchronizer”, which showed that increasing the current in the latch during metastability would reduce the impact of voltage supply reduction on resolution time. Their approach is to add two PMOS transistors on the latch nodes, and control them using a metastability detector (flipped mutual-exclusion filter) followed by a NAND gate to switch the additional PMOS devices on when the circuit develops any metastability. Their technique showed significant improvement in the metastability resolution response compared to the Jamb latch during nominal V_{DD} and lower voltages. The only drawback of the robust synchronizer is the overhead delay of the latch due to the use of small transistors in the metastability detector.

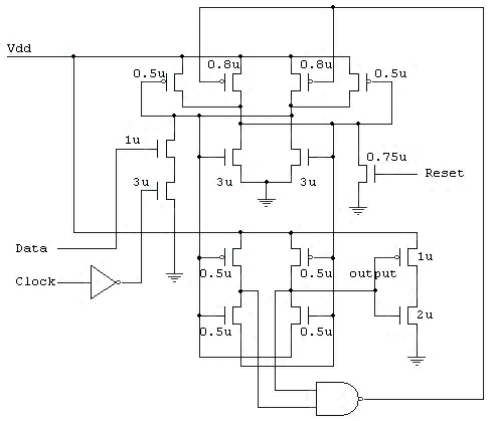


Figure 2.13 A robust synchronizer [82]

2.1.3.2 Synchronizer Performance

Reliability and latency are essential matters to be considered during the design of synchronizers due to the effect they have while implementing techniques to avoid metastability and the trade-offs between them. In general, the reliability of a single synchronizer is characterized by MTBF [50], previously presented in Equation (2.14). Latency is defined as the delay time for input data to propagate via a unit to generate the corresponding output data. For a synchronizer, latency can be defined as the time taken for an input data signal to go from the synchronizer input to produce a stable output value synchronous with the receiver clock.

To estimate the MTBF and latency, information is required about the circuit structure, the measured parameters and the system requirements, in terms of (τ, T_w, t_{SU}, t_H, t_{CQ}, t_{DQ}, f_{DATA}, f_{CLK} and, t_R). To assess the effectiveness of a synchronizer circuit, MTBF and total latency are estimated for available resolution time of 30τ to 40τ [50].

2.1.4 Arbiters

The main function of an arbiter is to organize the access between two or more independent clients and a common resource, based on a predefined protocol. The arbiter receives request signals from different clients that need access to the common resource, and then grants one client access using an acknowledgment signal. After the granted client completes its request, the arbiter may grant another client. The process of arbitration can be explained in three cases using the example in Figure 2.14 which shows a two-way arbiter, two clients C1 and C2 as well as a common resource CR**.** The first case illustrates the following. At the time when CR

is not busy, if client C1 asserts a request signal R1, then the arbiter will assert signal R to CR and wait for its permission by signal A; thereafter, the arbiter may send A1 to C1 to gain access to CR. After C1 completes its request, it de-asserts R1, so that the arbiter releases R and A1. The second case illustrates the following. When one client request is already granted, for instance C1, and the other client C2 sends its request signal, the arbiter will block R2 until C1 finishes and de-asserts R1. After that the arbiter will release A1, and then it grants client C2 with signal A2. The third case illustrates the following. At time of no requests and when both clients send their request signals simultaneously and the arrival time difference between the request signals is very small that it is not enough for the arbiter to make a decision and goes into metastability, which holds the decision of the arbiter longer than typical time. Later on, it should reach an arbitrary decision. In this case, there is an equal chance that the arbiter will grant any of these clients eventually. In the following section, arbiters and the impact of metastability on arbiters are discussed.

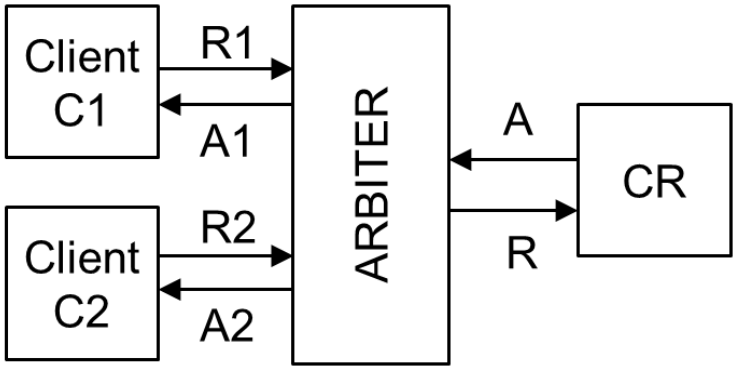


Figure 2.14 Example of two-way arbiter

2.1.4.1 Mutual Exclusion Element

The Mutual Exclusion element or MUTEX, an important component in asynchronous systems, is a simple two-way arbiter that is frequently used in the design of almost all arbiters. It commonly uses a set-reset latch composed of two cross-coupled NAND gates followed by a metastability filter to eliminate metastable events progressing in the following circuits, as depicted in Figure 2.15 [5, 46, 50].

The behavior of the MUTEX can be explained as follows. First, consider the status of no requests to the MUTEX, that is when both input request signals (R1 and R2) are zero volts, the internal nodes (N1 and N2) will be at V_{DD}, which turns ON both

NMOS transistors of the metastability filter pulling down both grant signals (G1 and G2) to 0V. In the case that one of the request signals going high, for example, if R1 rises to V_{DD} while R2 remains at 0V, the latch sets node N1 to 0V and N2 is held at V_{DD}, then, node N1 turns 'ON' the PMOS transistor at the bottom of the circuit to drive the output G1 to V_{DD} , while node N2 holds the NMOS transistor at the top 'ON', which keeping the output G2 pulled down to 0V. In a similar manner, when R2 becomes high while R1 is low, G2 will rise to V_{DD} while G1 remains at 0V. Another inevitable case is when both R1 and R2 rise to V_{DD} at the same time, nodes N1 and N2 will fall down together to a metastable level, usually around $V_{DD}/2$, which is observed by the metastability filter as logic '1' and both acknowledgment signals are kept low . The time the latch is held in metastability is dependent on the arrival time difference between the rising edges of R1 and R2 and the noise level in circuit. Only one of N1 and N2 will go down to 0V while the other will go back to V_{DD}, then the corresponding acknowledgment signal can rise to high V_{DD}, which indicates the end of metastability and the decision time of the MUTEX. Once the latch output voltage difference $|V_{N1}-V_{N2}|$ has increased enough over the threshold voltage of the transistors in the filter circuit, the metastability will be resolved and causes one of the acknowledgment signals to go high.

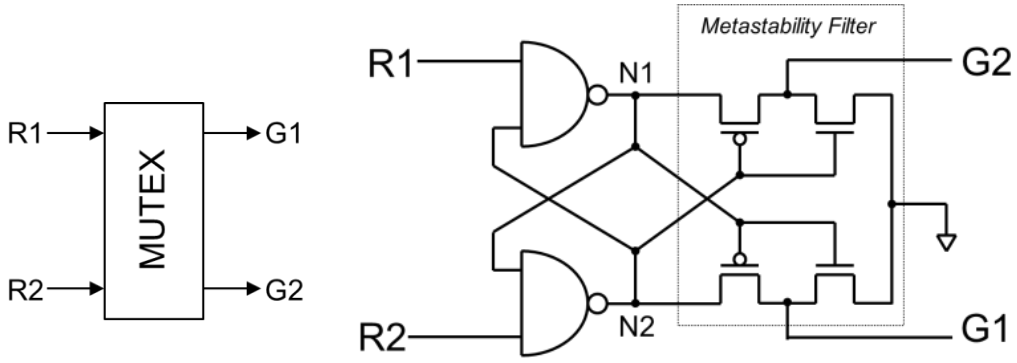


Figure 2.15 MUTEX symbol and circuit

The MUTEX is used to build handshake arbiters, for example a two-way arbiter as shown in Figure 2.16. It uses completion-detection gates, which are known as C-elements and shown in Figure 2.17. The MUTEX ensures that signals G1 and G2 are mutually exclusive, and then the two NAND gates following the MUTEX ensure that handshakes A1 and A2 are mutually exclusive, that is A2 can only go high if A1 is low and A1 can only go high if signal A2 is low. Thus, if handshaking is in progress along one channel, it blocks handshaking on the other channel. Once the MUTEX

decides which to grant, the C-element holds that grant until the acknowledgment A is issued by the common-resource and then the corresponding acknowledgment is issued.

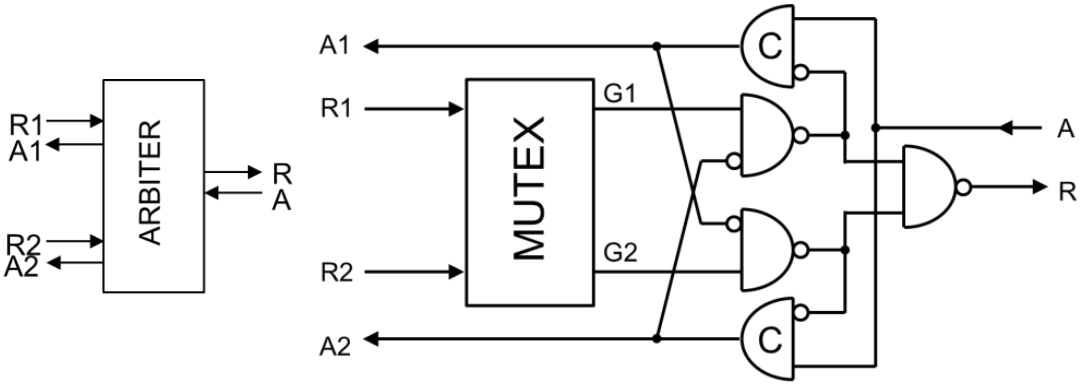
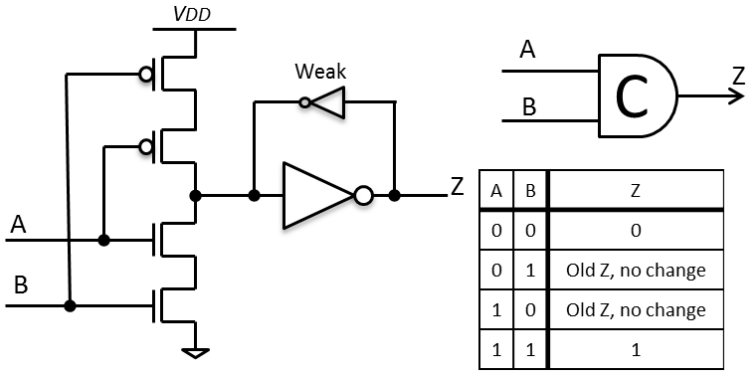


Figure 2.16 MUTEX-based two-way arbiter



A	B	Z
0	0	0
0	1	Old Z, no change
1	0	Old Z, no change
1	1	1

Figure 2.17 C-element circuit

2.1.4.2 MUTEX Performance Estimation

At some arrival input time differences between the request signals to a MUTEX arbiter, the latch output nodes will be forced to the metastable level. This event halts the arbiter from making any decision because of metastability may take longer time to recover and the nodes of the latch reach normal logic levels. It is important to estimate the reliability and speed of a two-way arbiter to differentiate between multiple designs.

In general, the reliability of a time constrained MUTEX arbiter is computed similarly to the flip-flop, as in Equation (2.14). For instance, if the input request signals are arriving at the MUTEX inputs at frequencies f_{R1} and f_{R2}, and the MUTEX is given a limited time t_m to resolve metastable events. Then, the Mean Time Between Failure (MTBF) of this MUTEX is computed by Equation (2.20).

$$MTBF = \frac{e^{t_m/\tau}}{T_w \cdot f_{R1} \cdot f_{R2}} \quad (2.20).$$

Typically, asynchronous arbiters do not have specified timing constraints, as a result the numerator of Equation (2.20) is infinity, therefore there will be no failure caused by a metastability event, even if it takes a very long time to resolve [50], under the assumption of using a metastability proof filter.

The performance of the MUTEX depends on the value of τ and the decision time t_d. The decision time t_d is the path delay time through a MUTEX receiving one request signal from the input request rising edge to its corresponding grant signal rising edge. A typical delay time of a MUTEX can be defined as the value of t_d plus the average time ($t_{average}$) taken to resolve metastability [50]. This average time is defined as Equation (2.21).

$$t_{average} = \tau \cdot \int_0^{T_n} ln\left(\frac{T_w}{\Delta t_{in}}\right) d\Delta t_{in} = \tau \cdot \left[1 + ln\left(\frac{T_w}{T_n}\right)\right] \quad (2.21),$$

where Δt_{in} is the input time difference between the two request signals, and T_n is the amount of variation in Δt_{in} due to noise or jitter. When $T_n \geq T_w$, then the average time is just τ, while if Δt_{in} variations are smaller than T_w, then the average time might be two or four times τ. For input variations ten times less than the metastability window, the MUTEX typical delay is around the value given by Equation (2.22).

$$t_{Typical} \cong t_d + 2 \cdot \tau \quad (2.22).$$

The typical value of t_d is much larger than the value of τ, due to propagation delays in the NAND gates and the metastability filter, which considerably limits the MUTEX overall speed.

2.1.4.3 Multi-Way Arbiters

Some multi-way arbiters can be constructed using multiple interconnected two-way arbiter MUTEX circuits, sometimes called multi-way mutual-exclusion elements. In the following section, multi-way arbiter structures are discussed, including the basic ones, such as mesh, tree and ring arbiters, and advanced techniques to build large arbiters, such as the ordered and priority arbiters.

The choice of multi-way arbiter structure depends on some of its characteristics, including complexity, latency, fairness and orderliness. In a multi-way arbiter these terms are defined as:

- *Latency* is the minimum decision time for an input request to propagate through the arbiter and be granted.
- *Fairness* is a principle of a multi-way arbiter with *N* inputs. The arbiter is considered fair if it guarantees that any input request will be granted after at most *N*-1 other requests.
- *Orderliness* is the original arrival sequence of input requests being preserved throughout arbitration.
- *Complexity* is the number of connecting wires, devices and cells.

a) Mesh Arbiters

The mesh arbiter [46, 50], cascades MUTEX circuits between request signals, to have each request signal arbitrated with each other request. For an n-way mesh arbiter, the number of MUTEXes used is on a 2-out-of-n basis (C_n^2). For instance a three-way arbiter needs three MUTEX elements, and a four-way arbiter as shown in Figure 2.18 requires six elements.

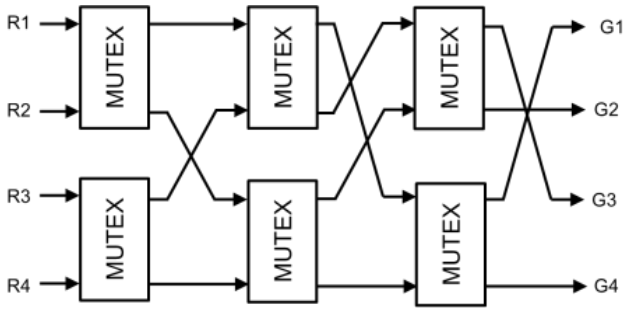


Figure 2.18 Four-way mesh arbiter [50]

The density of the mesh arbiter grows quadratically, while its latency is proportional to n-1. Consequently, this technique is not practical for arbiters with a large number of inputs, because latency and complexity will be very high. Another drawback of this design, is that it does not preserve the order of the incoming requests.

b) Tree Arbiter

The tree-arbiter [46, 50, 95], as the name suggests, uses a number of standard 2-way arbiter cell connected in a tree fashion. For instance, a 4-way tree arbiter would need three arbiter cells, as shown in Figure 2.19. In this structure, the requests are grouped into pairs and each pair is arbitrated through a two-way arbiter at the first stage. After each arbiter at the first stage has generated a new request, the following stage arbitrates the new requests. Then, at the final stage, one request that has propagated through the arbiter cells in the previous stages is granted and the acknowledgment is produced to the corresponding client. This technique was improved further by detecting the request signals separately from the arbiter, which saves any increased latencies in any of the MUTEXes. Although, the tree arbiter uses fewer MUTEX elements than the mesh arbiter, their complexity and latency are quite similar, due to the extra circuitry of the tree arbiter. The structure of a tree arbiter ensures fairness, but it does not guarantee orderliness.

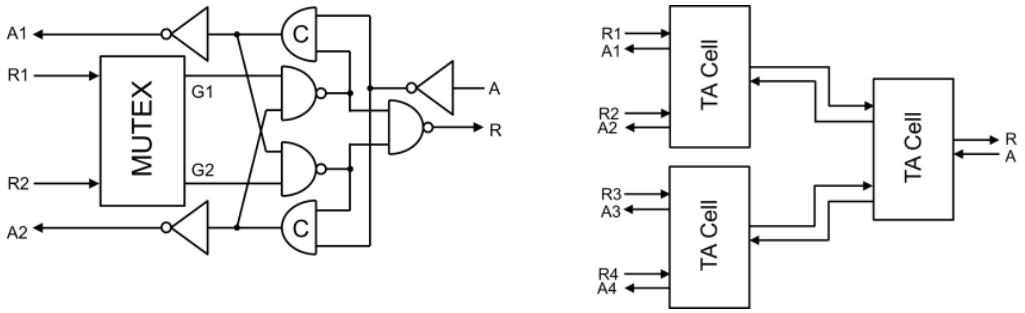


Figure 2.19 Four-way tree arbiter [50]

c) Ring Arbiter

One more traditional arbiter structure is called the token ring arbiter [46, 50]. It is constructed from a number of 2-way arbiters connected one after another in a ring structure as shown in Figure 2.20. A token signal rotates through each arbiter one at a time based on their topographical order, where each arbiter defines a contact node to one independent client, which is one MUTEX per input. In this way, each client is given a separate window of time to acquire this token and the arbiter grants this client without the need to arbitrate with other requests arriving at the same time. This technique consumes more power compared to the mesh and tree arbiters, because the token pulse signal keeps cycling the ring even there are no

request. The ring arbiter guarantees fairness, but not necessarily orderliness.

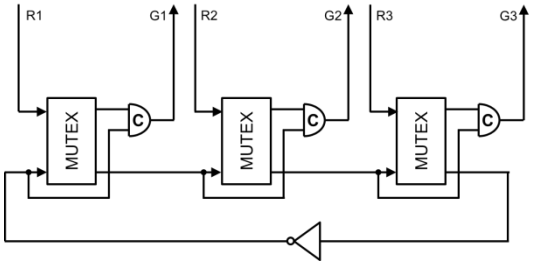


Figure 2.20 Three-way busy ring arbiter [50]

d) Multi-Flop Arbiter

Multi-way arbiters can also be constructed using multiple multi-input NAND gates, followed by a network of metastability filters. For example, a three-way arbiter using triple three-input NAND gates, connected as shown in Figure 2.21, called a tri-flop, followed by metastability filters. The advantage of such a circuit is that it may grant any request arriving unaccompanied by other request with very short latency compared to other multi-flop arbiters. But the tri-flop arbiter has three possible different metastable events if any two of the three requests arrive closely together, and has one possible ternary metastable event where all requests arrive simultaneously. According to a number of studies [96-98], the latter case may lead to oscillation in the nodes N1, N2 and N3, which cannot be filtered out from outputs G1, G2 and G3, under the assumption of symmetric gates and loads with a mismatch between the inputs of each of the NAND gates that are connected to feedback nodes N1 to N3.

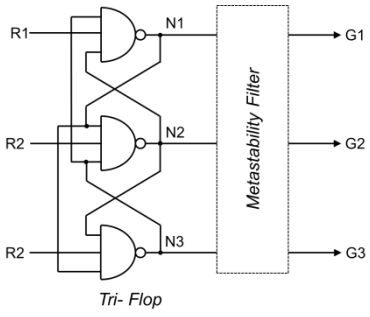


Figure 2.21 Tri-flop arbiter [50]

e) Ordered FIFO Arbiter

The ordered arbiter proposed by Bystrov *et. al.* [99] is a class of arbiters that

arbitrates the incoming requests by the order of their arrival. Its structure is composed of an input/output interface and request mask followed by an n-way MUTEX, followed by a spacer and n-1 FIFO (First-In-First-Out), as shown in Figure 2.22. It operates by storing incoming requests in a FIFO to preserve their original order of arrival for the granting process and freeing the MUTEX element to arbitrate the next request. Once the shared resource is released from the first client, the next client in the FIFO queue is immediately granted. The three-way ordered arbiter shown on Figure 2.22 is a possible implementation. D-elements [100] were used to perform the interface and request masking process. The size of this arbiter increases quadratically as the MUTEX size is increased in depth and the FIFO in breadth. The n-way MUTEX could be based on a mesh arbiter or a multi-flop arbiter. This arbiter is considered to be fair because a request will be granted after a sequence of requests that arrived before that one.

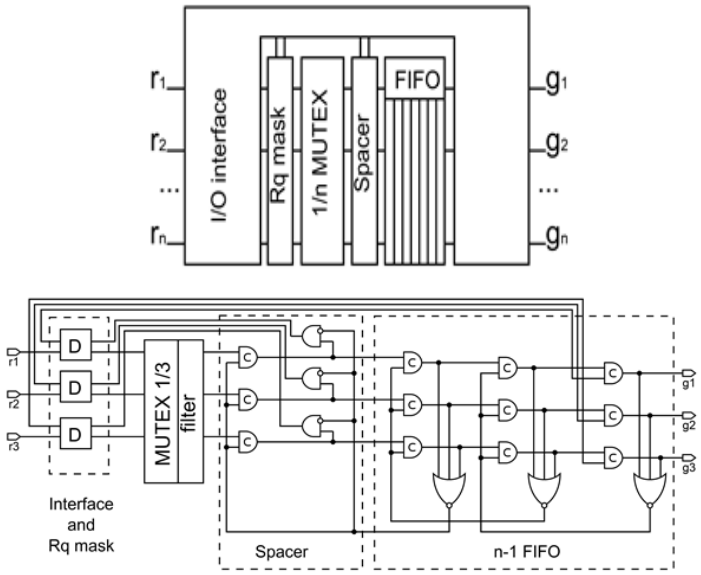


Figure 2.22 Ordered FIFO arbiter [50, 99]

f) Low-Latency Ordered Arbiter

A multi-way arbiter, based on a 2-way MUTEX, called ordered arbiter is proposed in [101]. For an n-way ordered arbiter, it requires an array of a combination of 2-out-of-n (C_n^2) MUTEX elements followed by n n-input AND gates. For example, a 4-way ordered arbiter requires six MUTEXes (MEes) and four 4-input AND gates, as shown in Figure 2.23. In this structure, each request is arbitrated with each other request by a MUTEX, which is similar to the mesh arbiter but it is done in parallel,

and then if one of the requests wins all the arbitrations, the outputs of these MUTEXes will switch the AND gate output of the winner request to a logic high level.

The main advantage of this structure is that it guarantees orderliness, and has low latency, which is about 4-gate delays for any number of inputs. This does not mean it has fixed latency for larger arbiters, because as the number of inputs increases, there will be more capacitive loading at the inputs node and larger AND gates will be required; as a result there will be an increase in the minimum latency in larger ordered arbiters.

One drawback of this design is that it may lead to deadlock, which halts the entire system because it is waiting for a signal transition from the arbiter. For example, if the 4-way ordered arbiter had three requests (R1, R2 and R3) arriving close together, three MUTEXes will go metastable and they may resolve differently (R1 wins in MUTEX 1, R2 in 2 and R3 in 3). This way all the outputs of the AND gates will remain at zero, because each of the three of the AND gates is missing one high input.

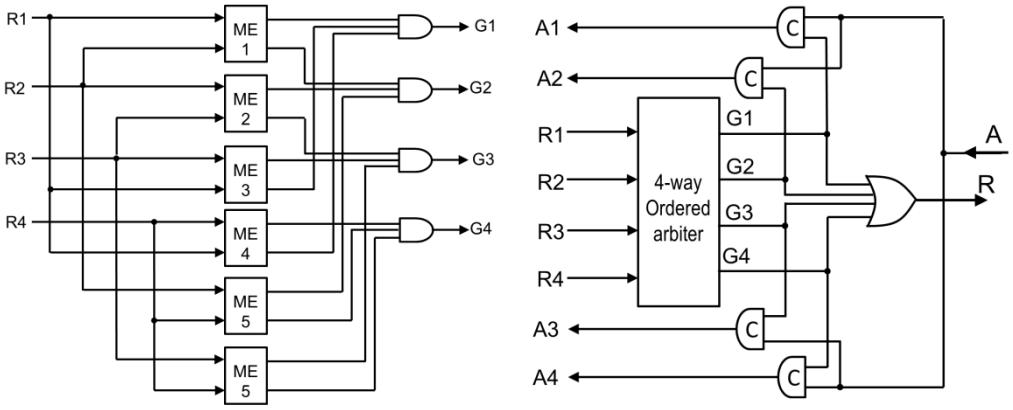


Figure 2.23 Low-latency ordered arbiter [101]

g) Priority Arbiter

A priority arbiter uses two separate processing blocks, as proposed in [102], its structure is shown in Figure 2.24. One block is called lock register, which is used to register the incoming requests, and the other block is a combinational circuit, which implements the priority function. Priority arbiters can be constructed with a static or dynamic priority function. Static priority arbiters use single wires as request inputs, whereas dynamic priority arbiters employ request inputs with

buses carrying priority data.

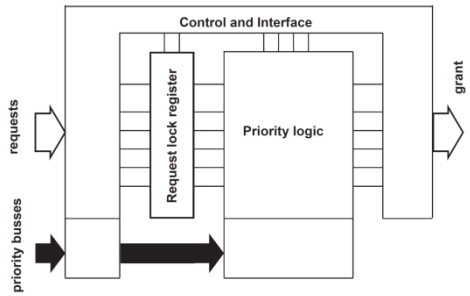


Figure 2.24 Priority arbiter [50, 102]

2.2 Synchronization Techniques in SoCs

Synchronous systems are composed of combinational circuits with sequential circuits triggered to read or write by a global clock signal. Sequential circuits include most digital circuits with one or more feedback connections, such as in bistable circuits, latches, flip-flops, registers, and memory cells. In synchronous systems having a single clock domain, the flow of data is coordinated with a global clock, which must arrive to all the internal sequential circuits simultaneously in order to secure the transfer of data between the subsystems without any uncertainties.

2.2.1 Global Clock Technique

In fully-synchronous systems with a global clock, the flow of data is coordinated with the global clock, which must arrive at all internal clocked elements and circuits simultaneously to secure the transfer of data between the subsystems without any uncertainties. Global clocks in fully-synchronous SoC have a number of disadvantages with the increased system size and complexity. The main three issues are, first, constructing a balanced and buffered clock tree distributed to each block and to each circuit within the block has become rather challenging and it is a critical task, because the clock signal may not arrive at the same time to different locations within the system, which is known as the clock skew, and fundamentally increases with technology scaling, clock speed and the number of clocked elements in the system. Second, the dynamic power consumption is directly related to the switching frequency and the number of clocked elements. Finally, the design complexity becomes greater when optimizing the whole system to run on one clock, which may introduce unnecessary constraints on some levels of the

subsystems. For example, a microprocessor needs to operate at the highest possible frequency, whereas external interfaces may operate at lower frequencies.

2.2.2 Asynchronous Technique

As a consequence of the growing challenges of synchronous systems caused by the use of a global clock, it has become desirable to remove the clock from the system and utilize asynchronous methods. Typically, asynchronous circuits are based on handshake interfaces with either dual-rail or bundled signaling. Overall, the removal of the clock results in a substantial improvement in power efficiency. Moreover, asynchronous circuits consume no dynamic energy if the components are inactive.

The design process for asynchronous circuits struggles with difficulties and complexity of using the existing design and verification tools, because they are primarily oriented towards designing synchronous systems based on synchronous libraries [44-47]. Additionally, asynchronous techniques consume more silicon area on SoC compared to synchronous ones, because as an alternative to buffered distributed clock signals, it enables and disables registers using unconventional logic circuits, such as completion-detection circuit.

These techniques lower the power consumption and timing variations, nevertheless the circuit area and design complexity become greater [41, 46, 47].

2.2.3 Multiple Clock Domain (MCD) Techniques

Eliminating the global clock from a system by employing multiple clocking schemes reduces the impact of the fully synchronous systems problems. Dividing the system into smaller domains with different performance and power conditions as well as utilizing a separate clock signal can eliminate the worry of clock skew, and reduce the circuit complexity and power budget, and simplify the design process. It also poses the option to integrate multiple voltage domains as well, which basically brings the power budget down.

To employ the MCD design concept, there are two major design concerns; first the nature of clock signals of each corresponding domain; and second the synchronization techniques to pass control and data signals between some of these domains. Multiple clock signals can be provided as either derived clock signals or independent clock signals. The synchronization techniques are dependent on the

relationship between the clocks of the communicating domains.

Derived clock signals are obtained from the original clock generator, such as a PLL, using specialized circuits, for example a clock divider or circuit delays, to derive the new clocks with different frequencies or phases. Independent clock signals are locally generated using a PLL or a ring oscillator. The first method suffers from the clock's distribution overhead starting from the main clock generator and down to different derived clock signals, whereas the second has less distribution overhead. Both methods reduce the impact of clock skew caused by variations in interconnect delays, because each domain is verified for a separate clock, but requires careful synchronization between the domains.

Synchronization design is based on the relationship between sender clock and receiver clock in terms of the difference in frequency and in phase [5, 103]. Generally, it can be either synchronous, mesochronous, plesiochronous, periodic or asynchronous. In a synchronous relationship, the difference in frequency and phase is zero and no synchronization is needed. In a mesochronous relationship, there is a small difference in phase between the clock and input, usually due to known delays in the master clock or derived clock, and it needs phase compensation. In a plesiochronous relationship, there is a small difference in frequency and variable phase difference due to derived clock and skew, adaptive phase compensation is needed. In a periodic relationship, there is a difference in frequency and variable phase but the relation between the sender clock and receiver clock can be predicted, commonly due to local clocks being derived from a master clock by a division or multiplication, and requires a predicative synchronizer. In an asynchronous relationship, there is a large difference between the frequencies with an undetectable relationship between them and variable phase difference, due to derived or independent clock signals, and it needs asynchronous clock-domain crossing synchronization, which can be accomplished via a brute-force synchronizer, handshake signaling or a FIFO, or by stopping the clock if the receiving clock is generated in a local ring oscillator.

Overall, the SoC design flow becomes more complicated when implementing a multiple clock domain concept than a fully synchronous one, therefore, the trade-off between the benefits and potential complexities must be considered.

2.2.4 GALS Techniques

An alternative approach to overcoming the global clock distribution problems of synchronous systems is to implement a GALS architecture, which is constructed from multiple independently clocked synchronous subsystems, which are communicated via asynchronous interfaces and protocols. This allows designers to continue using synchronous methods locally and build larger systems.

The purpose of the asynchronous interface involves exchanging data between two independent clock domains, which is a major difficulty. Because both domains are unaware of each other's clock details, signals crossing domains may switch near the receiving clock edge which leads to a violation of the setup and hold time and to a metastability failure which would be unavoidable in this case [50]. Given that, the interface design must focus on a recovery method during or after the occurrence of such a timing violation. The interface design cannot be automated as synchronous circuits, and it mostly needs more detailed analysis and design time, especially with the limited availability of design tools [44, 46, 47, 50, 104].

A number of approaches to the design of asynchronous interfaces between two clocked regions are presented in literature [44, 104]. One approach focuses on synchronizing the data with the clock, for example handshake and FIFO synchronization. The handshake synchronization is based on flip-flop synchronizers and has a small area overhead but has to deal with metastability failures and the increased latency due to number of synchronization cycles between the domains which therefore reduce the rate of communication. On the other hand, an asynchronous FIFO synchronizer adds no metastability failures to the data path but increases the latency and area overheads. An alternative approach concentrates on synchronizing the local clock with the arriving signals, known as a 'pausable clocking' technique [44, 50, 104], which takes out the metastability failure from the data path towards the clock path but requires an arbiter circuit with a local clock generator, such as a ring oscillator, to control the pausing.

The GALS architecture, in contrast with a global-synchronous architecture, can provide faster performance by at least 8% under within-die variations in gate length and thermal distribution [48]. Therefore, the nature of synchronization failure and circuits needs to be carefully analyzed and designed with attention to

the effects of PVT variability.

2.2.5 Handshake Signaling

A set of data signals can be synchronized between two clock domains using a handshake synchronizer and bundled-data protocol based on a set of control signals. A bundled-data protocol needs a single request signal and a single acknowledge signal bundled with a number of data signals.

The handshake signaling protocol can be four-phase or two phase communication [4, 46], shown in Figure 2.25. The 4-phase protocol follows four steps, first, the sender starts handshake by asserting the request signal (*Req*) by setting it to logic '1'. Second, the receiver accepts the request signal and stores the data, after that the receiver asserts the acknowledgment signal (*Ack*) by setting it to logic '1'. After that, the sender receives the *Ack* signal, then it de-asserts the *Req* signal by resetting it to logic '0'. Finally, the sender de-asserts the *Ack* signal, by resetting it to logic '0', to end the handshake. On the other hand, a 2-phase protocol uses two transitions for a handshake cycle. The sender starts the handshake asserts the *Req* signal by changing it logic state, then, after the receiver receives *Req* and saves the data, it asserts the *Ack* signal by changing it logic state, which ends this handshake. Therefore it is more time efficient than the 4-phase protocol.

Figure 2.26 shows a typical two-phase handshake synchronizer [4]. The request and acknowledgment signals are synchronized using 2FF synchronizers to reduce metastability failures, therefore two-cycles for each of signal transition are required, this leads to the main disadvantage of handshake signaling is the latency for data to be delivered and acknowledged.

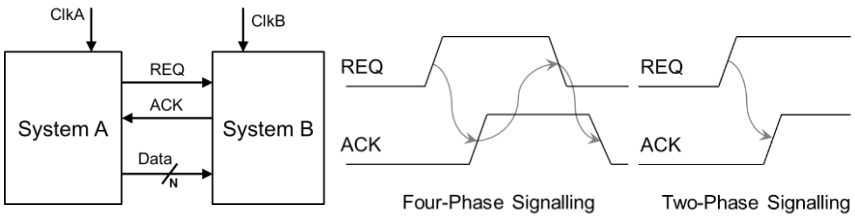


Figure 2.25 Handshake signaling protocols

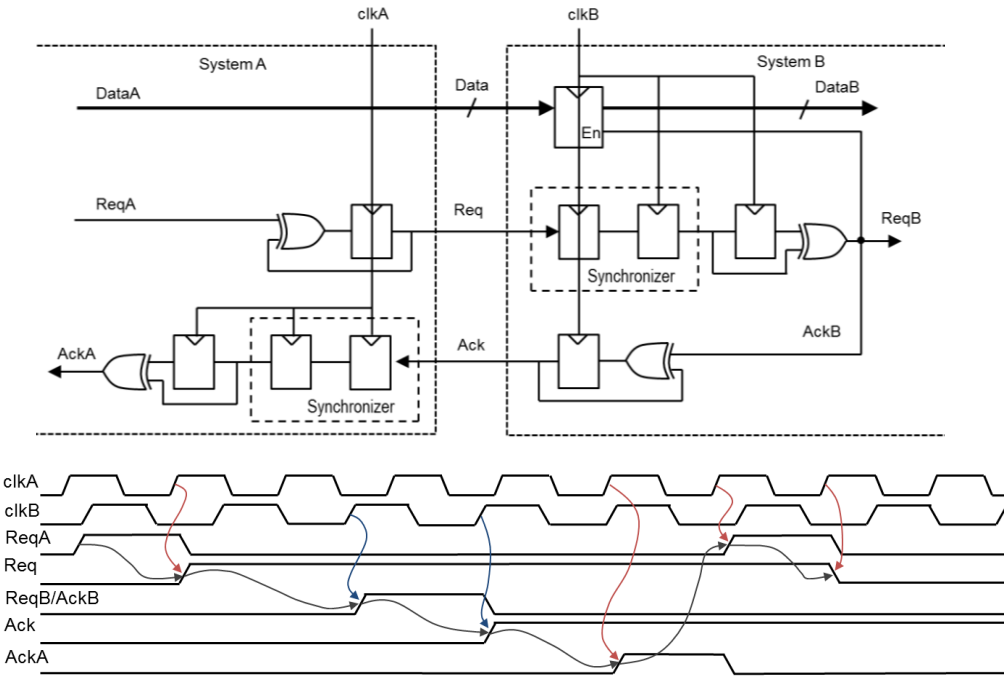


Figure 2.26 Two-phase handshake synchronization [4]

2.2.6 FIFO

A FIFO synchronizer, shown in Figure 2.27, is a common method for transferring data with high throughput between two clock domains. A memory block with two ports is typically used to store data in the FIFO.

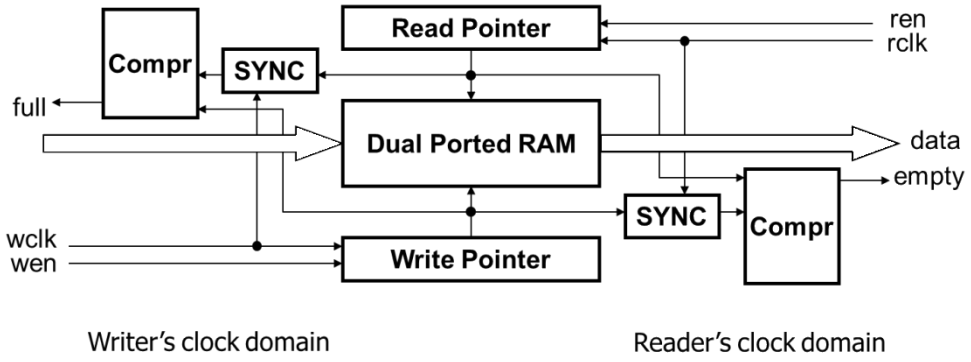


Figure 2.27 FIFO synchronizer [70]

In general, one port is connected to the sender to write data into the memory. The other port is linked to the receiver to read the stored data from the memory. The transfer rate between the sender or receiver and the FIFO is at one data word per clock of the sender or the receiver. The FIFO needs two flag signals to indicate the status of its memory as either empty or full. In principle, sending the data with one clock through a shared memory and receiving it with another clock appears to be the ideal solution for passing data between two clock domains and avoids

metastability problems in the datapath, but the generation the flag signals (full and empty) can be a challenging process.

2.3 Multiple Voltage Domain (MVD) in SoCs

A fundamental approach to lowering the overall power per SoC is known as multi-voltage design. This method is based on partitioning the internal circuitry of the chip into Multiple Voltage Domains (MVD), where each one has its separate supply. The partitioning in a modern SoC design is created by realizing the different requirements of different blocks, such as targeted performance and constraints. For example, a processor may need to operate on the fastest permissible clock for a given semiconductor technology, which requires a higher voltage supply. On the other hand, a peripheral interface block may operate at low frequencies, which may satisfy its timing constraints if a lower voltage supply is used, and therefore, it will have lower power consumption.

There are a number of multi-voltage design strategies. According to [4, 55], they can be categorized in order of complexity as Static Voltage Scaling (SVS), Multi-level Voltage Scaling (MVS), Dynamic Voltage and Frequency Scaling (DVFS), and Adaptive Voltage Scaling (AVS). The SVS is the simplest method because it gives each block in the system a single different fixed voltage supply. The MVS is an SVS in addition to one or more blocks given two or more separate fixed voltage levels to provide different operating modes by switching between them, which requires more routing of different power rails. The DVFS is a MVS in addition to at least one block where voltage and frequency can be scaled dynamically together providing multiple voltage levels in one block. Lastly, AVS is a block operating under DVFS with a feedback to adapt the voltage.

2.3.1 Dynamic Voltage and Frequency Scaling (DVFS)

Dynamic Voltage Frequency Scaling (DVFS) is a popular method for developing energy-efficient systems. The key concept of DVFS is reducing the voltage supply and clock frequency based on the work load. The voltage and frequency values can be determined analytically in discrete pairs for different loading conditions. These pairs are stored in a look-up table for the processing element to decide which pair to use based on current load.

The vast majority of microprocessors [105] are designed nowadays with a CMOS process that has a limited operating frequency dependent on process as well as the voltage supply. For instance, a processor can operate at lower voltage supply, if low frequency is sufficient. It also may need to switch to a high performance mode only for a short period of time for some applications, while low-performance and low-power mode would be enough for the rest of the time. In this way, DVFS significantly reduces the energy overhead by dynamically scaling the voltage and frequency.

To employ DVFS within a processor, a programmable power supply and programmable PLL clock generator at least are necessary to scale up and down the voltage and frequency. For low-power, the DVFS scaling down operation is executed in two steps. First, the processor decides its minimum clock frequency based on the workload, then it reduces the voltage supply to the minimum value that can support that frequency. For high performance, scaling up is performed by increasing the voltage supply first to the target voltage until it stabilizes, then the processor programs the clock frequency. In both cases of scaling, the operation of the processor continues during the scaling procedure except only if the original frequency of the PLL has to be changed, whereupon all of the clocks in the system are disabled until the new frequency is reached and settled.

2.3.2 Voltage Level-Shifters

There are numerous challenges in the design of multi-voltage SoC. One of the main challenges is level-shifters between different power rails.

The main reason to use level-shifters is stacked NMOS and PMOS transistors in CMOS logic gates normally cause short circuit currents during logic transitions in normal cases, that is, input driver and gate are supplied by the same voltage. If the gate input is driven by logic 1 from a circuit with a lower voltage supply, larger short circuit currents will flow through the stack and cause excessive static power consumption. This is because the PMOS transistors will still be turned on as its gate to source voltage is lower than its threshold voltage. Therefore, both NMOS and PMOS transistors are on and conducting short circuit currents.

Utilizing level-shifters between multiple voltage domains is a good and easy solution to ensure that each domain achieves full voltage swings. The benefit of this approach is that any voltage swing and timing characterization concerns will

be limited to the periphery of each voltage domain, and therefore the timing conditions within each domain can be used securely.

a) Downshifting and Upshifting

High to low voltage conversion or "downshift" can be easily implemented using a CMOS buffer (two inverters in series) powered with the lower voltage. The high voltage simply overdrives the input of the buffer, which outputs a faster transition.

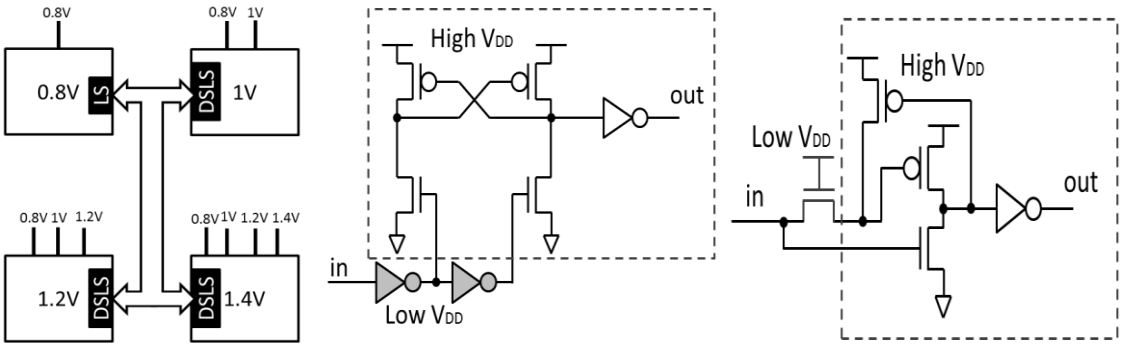
On the other hand, up-shifting incoming signals degrades the switching time the inputs of the receiver and could lead to increased short-circuit currents and reduced noise margins. This is more critical for the buffering of clock signal between two voltage domains, as the clock skew may increase.

There are a number of techniques to design an upshifting level-shifter [55]. This class of shifters must be carefully tested with the receiving block timing conditions, because upshifting introduces critically long delays during the transitions of the input signal. Level-shifting cells can be classified as either dual-supply or single-supply.

b) Upshifting: Dual-Supply and Single-Supply

The Dual-Supply Level-Shifter (DSLS) technique requires two voltage supply connections that must be connected to a mutual ground, the lower one is for the sender and the higher one is for the receiver. This requirement becomes a critical problem if the receiving block takes signals from multiple senders with different voltage supplies, and is called power supply connection congestion. If the block with highest voltage supply needs to communicate with all the other blocks, then each of the other voltage rails must be wired to this block, as shown in Figure 2.28(a).

The conventional design of a DSLS is shown in Figure 2.28(b) which is called a Differential Cascode Voltage Switch (DCVS). This design receives at the inputs the low voltage signal and its inverted form to switch a cross-coupled PMOS transistor powered by the higher voltage.



(a) DS power pins (b) DCVS (c) PGLS

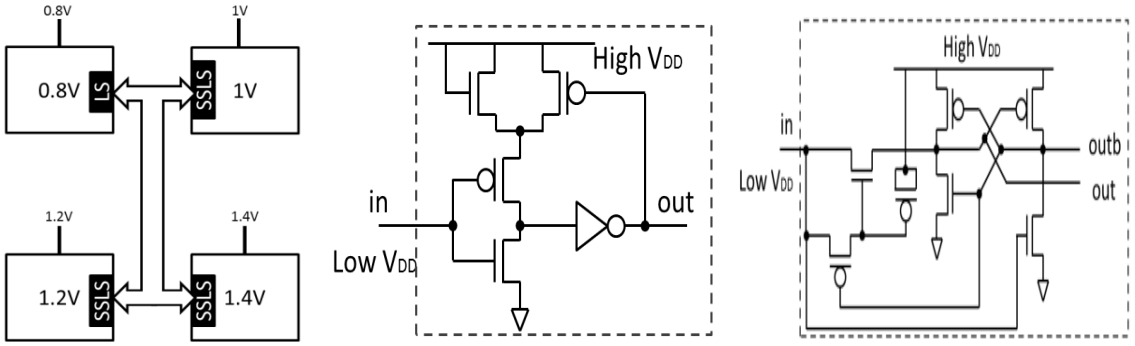
Figure 2.28 Dual-Supply (DS) Level-Shifters (LS)

An alternative design called the Pass-Gate Level-Shifter (PGLS) is based on a weak feedback PMOS transistor with an input NMOS pass-gate transistor. When the input voltage changes from 0V to V_{DDL}, then the NMOS transistor isolates the gate of the PMOS transistor from the previous logic stage, while the feedback PMOS transistor pull ups the gate-terminal of the PMOS transistor to V_{DDH}. The PGLS has a better speed and less energy consumption than the DCVS shifter [106-108]. If the pass gate NMOS transistor fails to isolate the feedback PMOS transistor, a reverse current will flow from V_{DDH} passing through the feedback PMOS transistor, the pass-gate NMOS, the PMOS transistor of the previous stage to V_{DDL}, which creates a short-circuit between the two voltage supplies and will change the voltage at the input node.

Level-shifters with a single-supply allow communication between MVD in a system without the need for additional power rails and remove the dual-supply problem of routing congestion of power supplies wires as shown Figure 2.29(a), resulting in reduced complexity of the SoC. These are known as Single-Supply Level-Shifter (SSLS).

The circuit in Figure 2.29(b) appeared in [106, 108] as an SSLS circuit. The basic operation of this circuit exploits the threshold voltage drop across the diode-connected NMOS transistor to maintain a virtual low voltage supply to the inverter below. Another circuit proposed by [109] is shown in Figure 2.29(c), and it is based on the pass-gate LS with the addition of a PMOS capacitor-connected to control the pass-gate NMOS transistor. In the case the input 'in' being at a logic '1' of V_{DDL} , the output NMOS transistor will turn on and pull the output node 'outb' to logic '0', by which the cross-coupling effect the node 'out' becomes logic '1' of V_{DDH}.

Also, the pass-gate PMOS transistor is switched on to charge the following node at the capacitor gate to V_{DDL}, and because V_{DDH} is higher than V_{DDL} the pass-gate NMOS transistor remains off. In the other case, where the input is at logic '0', the charged node turns on the pass-gate NMOS transistor, and discharge the output node 'out' to logic '0', by which the output node 'outb' becomes logic '1' of V_{DDH}.



(a) SS power pins (b) Diode-connected SSLS (c) PG-SSLS

Figure 2.29 Single-Supply (SS) Level-Shifters (LS)

c) Unidirectional and Bidirectional Level-Shifters

Conventional level-shifters can only be designed to either upshift or downshift, that is in one direction. This limitation poses a problem for the MVD methods that can vary the voltage supply during the system operation, as in the case of AVS. Figure 2.30 shows an example of a bidirectional level-shifter depicted [110] based on the pass-gate technique. Although these circuits may be attractive, they require unconventional techniques to design and tools to implement and test. As bidirectional level-shifters need to shift voltages correctly between two domains, they also need to verify the timing conditions to transfer across the domains.

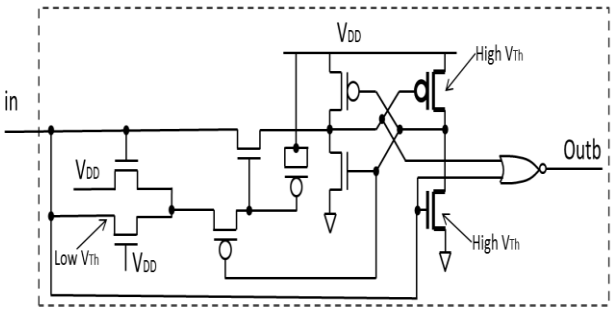


Figure 2.30 Example of bidirectional level-shifters [110]

d) Placement of Level-Shifters

Level-Shifters should be placed between two communicating voltage domains at the input of the receiving domain, and at least one of the domains should have a static voltage supply. In the case of level-shifting between two domains each with a separate scaling voltage supply, it is recommended to always upshift to a fixed voltage domain and then down-shift to the other domain [111] as shown on the right of Figure 2.31, unless bidirectional variable voltage to variable voltage level-shifters are available.

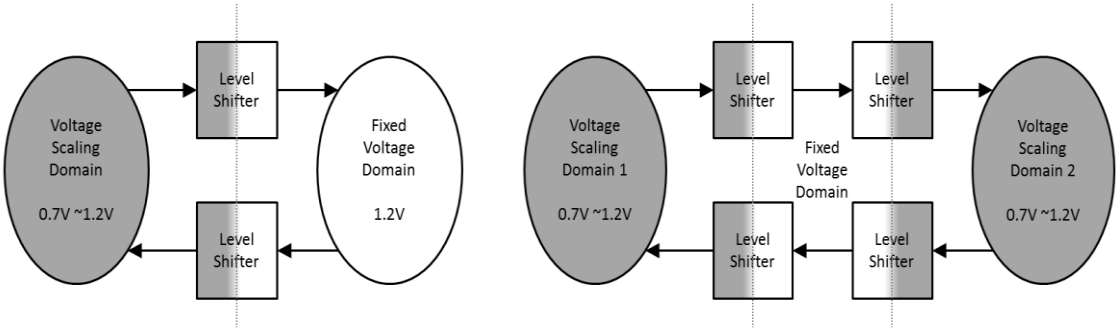


Figure 2.31 Placement of Level Shifters [111]

2.3.3 MVD and MCD Concerns

For signals crossing domain boundaries to be recognized correctly at the destination, voltage re-leveling is required in an MVD and retiming is required in an MCD. In an MVD that scales frequency with voltage, either at a predictable point as in the DVFS or an unpredictable point as in the AVS, retiming is as important as level-shifting because signal timing will certainly vary with voltage. Voltage scaling goes together with clock frequency scaling and the boundaries of a clock domain are also the boundaries for voltage scaling.

A synchronous interface between the scaled-voltage-frequency domain and the rest of the system is incapable of operating efficiently as voltage and frequency are varied, because the clock tree delays and skew will vary too. On the other hand, an asynchronous interface with synchronizers will resolve the wide variation in frequency and voltage. Figure 2.32 shows two examples of handshake synchronization and level-shifters; the top one is between a fixed V_{DD1} domain and variable V_{DD2} domain with different clocks, and one of the bottom is between two variable-voltage/clock domains separated by an interface voltage-domain as recommended by [111].

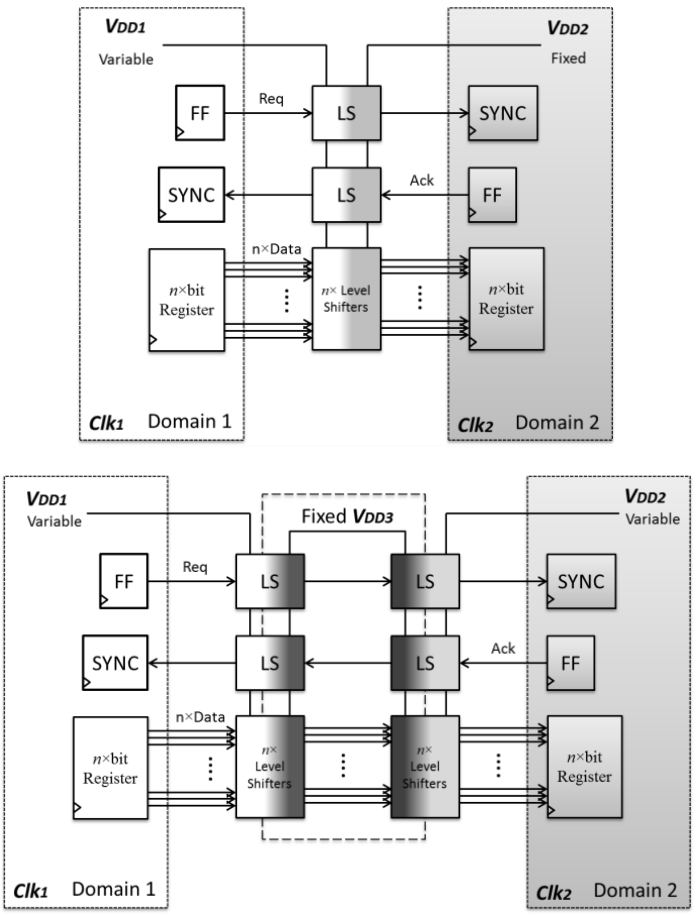


Figure 2.32 Handshake synchronization level-shifting requirements

2.4 Characterizing Flip-Flops' Performance

Flip-flops are commonly designed as edge-triggered storage elements based on latches. They pass the input data at the clock rising or falling edge, holding the signal stable until the next edge triggers. Any transition at the flip-flop input propagates to its output only if it arrives earlier than the triggering edge of the clock cycle. To pass the data securely to the output, the input signal and the clock must satisfy some timing restrictions, commonly known as setup and hold times. These are key metrics in identifying the maximum clock frequency of a digital system; therefore, they need to be precisely specified.

2.4.1 Delay Time

In general, the delay relationship between transitions of two signals is measured from the 50% point of full voltage scale (V_{DD}) of the transition of the first signal to the 50% point of full-scale voltage (V_{DD}) of the transition of the second signal. In a logic gate, the propagation delay is approximated to $0.7\times RC$ of the switched branch

of the circuit that transit the output from low-to-high or high-to-low, where R is the resistance of the conducting branch, either to ground or V_{DD}, and C is the total equivalent capacitance at the output node. For a flip-flop, the clock-to-output propagation delay (t_{CQ}) is the time difference measured between the clock triggering edge and the output Q transition (from 50% V_{DD} of the clock edge to 50% V_{DD} of output transition), when the input data signal is stable near the clock edge, that is data-to-clock time difference (t_{DC}) is wide enough and does not violate the setup and hold time restrictions. Besides the flip-flop configuration, the value of this delay is a function of t_{DC}, clock edge rise/fall time, voltage supply, temperature, process parameters and the output load [93]. In general, the delay through a flip-flop experienced by a rising input transition is different from a falling input transition.

The data-to-output delay is the delay measured from the input data transition to the output, if appropriate clocking is applied to the flip-flop. This delay is not a good metric of a flip-flop performance because of its dependence on the data arrival to clock edge offset time. The data-to-output delay time (t_{DQ}) for a flip-flop is estimated at the minimum allowed data-to-clock time.

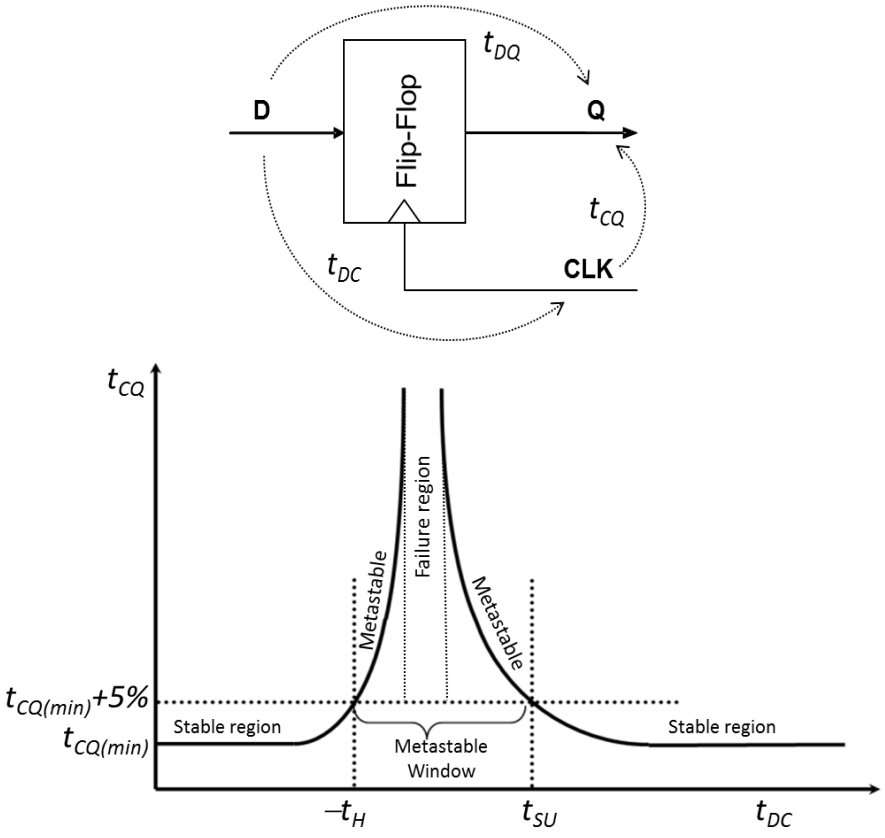


Figure 2.33 Flip-flop timing characteristics

2.4.2 Setup Time

The setup time (t_{SU}) is commonly defined as the minimum allowed time between the input data transition and the triggering edge of the clock, to produce a valid output [4, 5]. It is characterized as the point where a specific increase in clock-to-output delay of a flip-flop caused by data getting closer to clock as shown in Figure 2.34. The increase in the offset beyond this point leads to an extended clock-to-output delay, and can even result in a flip-flop failure, if data arrives too late that the flip-flop is unable to record this input data transition.

In general, setup times of storing logic '1' is different from storing logic '0'. The setup time of logic '1' is measured at data a rising edge transition from '0' to '1', and the setup time of logic '0' is measured at data falling edge transition from '1' to '0'.

The right-hand curve in Figure 2.33 shows the input time values (data-to-clock time or t_{DC}) plotted against the output time values (clock-to-output time or t_{CQ}). This is plotted by recording measurements of both values (t_{DC} and t_{CQ}) taken at different points of arrival times of new input value approaching the clock edge. This plot gives a clear view of the regions of normal operation and failure of the flip-flop.

An alternative definition of the setup time is the data-to-clock offset time which results in the minimal data-to-output delay [92].

Setup time value is dependent on the flip-flop configuration, process parameters, the voltage supply, the temperature, and the simulation setup.

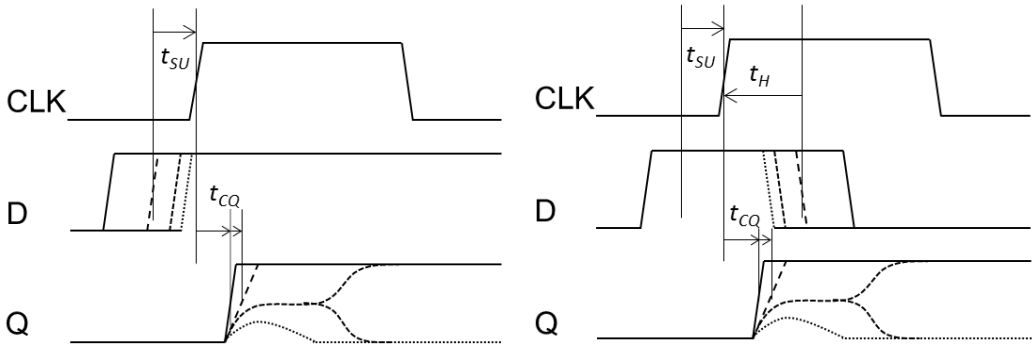
2.4.3 Hold Time

The hold time (t_H) is defined as the minimum time interval for which the data signal must be kept stable at the input of a flip-flop after the clock triggering edge to maintain a stable and valid output value [4, 5]. If the input signal changes earlier before the hold time and then changes back to its previous state during or after the clock transition, the clock-to-output delay of the flip-flop will increase, as shown in Figure 2.34.

In a similar manner to the setup time, hold time is evaluated at the input-to-clock signal that causes a 5% increment for typical application, or 10% increment for

variability studies, above the minimum clock-to-output delay. The hold time for retaining logic '1' is different from retaining logic '0'. The hold time of logic '1' is measured at data rising edge, and the hold time of logic '0' is measured at data falling edge.

Similar to the setup time, the hold time is dependent on the flip-flop configuration, process parameters, the voltage supply, the temperature, and the simulation setup.



Setup-time Hold-time

Figure 2.34 Flip-flop setup-time and hold-time

2.4.4 Finding Setup and Hold Time

Typically, finding the setup and hold times [4, 5, 92] is a binary search process and it necessarily requires comprehensive SPICE-level transient simulations of latches or flip-flops using accurate device models. From the definition of setup and hold times, the conventional method of finding their values is to set the time interval between the arrival of the input data signal and the clock signal [4, 5] and run simulations to measure the clock-to-output time, then repeat process with a narrowed time interval until the targeted increase in clock-to-output time is reached. Accordingly, determining latch/flip-flop setup and hold times is a more computationally challenging process than finding delays of combinational circuits.

An alternative direct method to estimate the setup and hold times of static flip-flops in one or two SPICE-level transient simulations was presented in [112] and partly in [113]. This method is based on measuring two path delays in the circuit; the first value is related to the transition of the data signal via the data path within a flip-flop to a predefined internal node before the master or slave latch, whereas the second value is associated with the transition of the clock signal through clock path within the flip-flop to the predefined internal node before the master or slave latch.

Setup time is computed by obtaining the difference between the two delay time values; the first value is taken from the transition of the input data signal to the input of the gate guarding the slave latch; and the second value is taken from the transition of the clock signal to the input of the gate guarding the slave latch.

Similarly, hold time is computed by obtaining the difference between the two delay time values; the first value is taken from the transition of the input data signal to the input of the gate guarding the master latch; and the second value is taken from the transition of the clock signal external input to the input of the gate guarding the master latch.

2.4.5 Metastability Metrics

The degree of significance of metastable events occurring in a latch or flip-flop can be predicted by obtaining the metastability resolution time constant and the window for that circuit [50, 74, 75]. In a flip-flop circuit, Dike and Burton [74] divided the propagation delay time into a deterministic region and a true metastability region. The former region is located by the finding the setup and hold times, and the flip-flop is close to metastability. The time for flip-flops, in the latter region, to reach one of its stable states is not deterministic [74]. As mentioned before in Section 2.1.1.3, the time for metastability in a flip-flop to resolve is limited, and therefore, a failure occurs.

2.4.5.1 Metastability Resolution Time Constant

The value of the metastability resolution time constant τ can be determined directly from two methods using a transistor-level transient circuit simulator with high accuracy [50, 74, 81].

The first method needs a number of measurements within the setup and hold window. Then the time constant τ is the slope of the input values t_{DC} and output values t_{CQ} within the exponential region of the metastability window from Figure 2.33. However, this only gives an estimation of the true value of τ, because true metastability occurs within tens or less of femto-seconds time differences between the edges of the data and clock signals and should be time stepped at least at 10fs or less [50, 74]. The slope of the exponential region is a semi-log slope, and can be written as:

$$\tau = \frac{t_{CQ1} - t_{CQ2}}{ln\left|\frac{t_{DC2}}{t_{DC1}}\right|} \quad (2.23).$$

An alternative method is a direct method [50, 74] to find the true value of τ. First, the latch of interest is shorted by a controlled switch and an offset DC voltage source between the latch nodes (see Figure 2.35), which forces the latch to be in deep metastability. Then, the switch is opened at time t_0 to let the latch node voltages diverge, one to V_{DD}, the other one to ground. The resultant value of τ is the slope of the differential voltage between nodes $V_{A\text{-}B}$ during the resolution, using the following equation:

$$\tau = \frac{t_1 - t_2}{ln\left|\frac{V_2}{V_1}\right|} \quad (2.24).$$

This method is effective if the latch is symmetric. In the case of an unsymmetrical latch, then the offset voltage must be varied to find the balance point of the latch, before which the latch node A, for instance, would resolve to V_{DD} and after which to ground. This requires a binary search.

2.4.5.2 Metastability Window T_w

There are a number of definitions for T_w in the literature [50, 71, 74, 114, 115]. In the context of using flip-flops as synchronizers, T_w can be defined as the region where metastability may occur when the setup and hold times are violated, see Figure 2.33. It can be said that T_w and the setup plus hold time window are related, and they are to a good approximation of the actual T_w region. Nevertheless, the concept of the metastability window is applied differently in [115], where it is defined for the asymptotic width of the metastability region (*cf.* failure region in Figure 2.33), while the term "metastability window" was referred to as equivalent to $T_w \cdot e^{-t_R/\tau}$, the former meaning will be used throughout this work.

In general, the metastability window is considered narrower than the setup to hold region [114], and using the value of setup plus hold time (t_{S+H}) instead of the metastability window to compute reliability will produce a reserved value of MTBF than with T_w. However, using the reserved value could be considered good practice, based on Equation (2.14); if an acceptable value of $MTBF(t_{S+H})$ is achieved, then the actual value $MTBF(T_w)$ will definitely be longer.

$$MTBF(t_{S+H}) = \frac{e^{t_R/\tau}}{t_{S+H} f_d f_c} \ll MTBF(T_w) = \frac{e^{t_R/\tau}}{T_w f_d f_c} \quad (2.25).$$

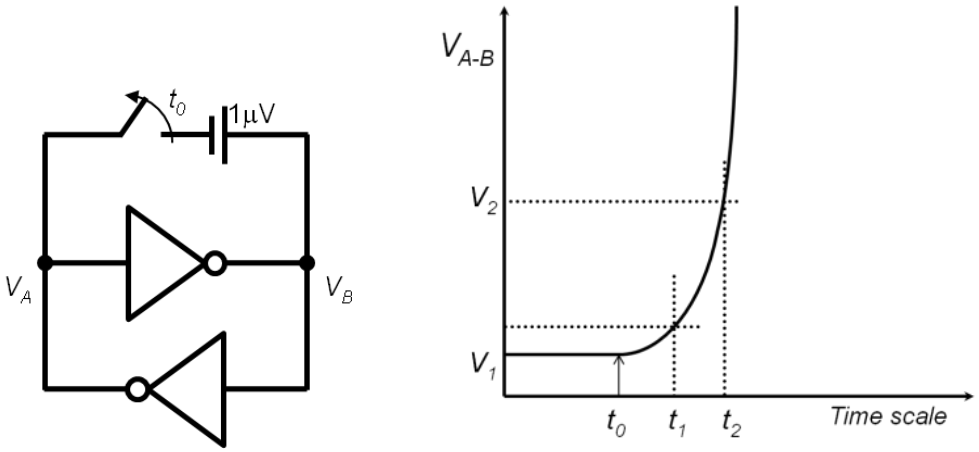


Figure 2.35 Direct measurement of the metastability time constant

2.4.6 Power and Energy

The main sources of power dissipation in digital circuits are short circuit current, switching current and leakage currents. Static power consumption is mainly due to leakage currents in transistors and it is computed as the total leakage current times V_{DD}. The expression for dynamic power dissipation $P_{Dynamic}$, given in Equation (2.26), represents the power consumed during charging and discharging activities, which depends on the voltage supply V_{DD}, frequency of operation f, probability of data switching α and the effective capacitance of the circuit C_{eff}, in addition to the power dissipation during switching caused by a short circuit current I_{SC} over time t_{SC} where both PMOS and NMOS transistors are conducting.

$$P_{Dynamic} = f \alpha C_{eff} V_{DD}^2 + f\, t_{SC} I_{SC}\, V_{DD} \quad (2.26).$$

This equation highlights the trade-offs between speed and power, as they are important in high-performance and low-power applications. To determine the optimum clock frequency and power consumption, the term power-delay product (*PDP*) is used, where $PDP = P_{Dynamic} \times t_{DQ}$, which is the energy spent per switching event in a flip-flop. A rather better term to identify the optimum speed and energy consumption is 'energy-delay product' (*EDP*) which is defined in Equation (2.27). Both *PDP* and *EDP* are quality metrics for digital circuits.

$$EDP = PDP \times t_{DQ} = P_{Dynamic} \times t_{DQ}^2 \quad (2.27).$$

Dynamic power can be simulated using transient analysis and measurements of currents taken during a number of triggering cycles with data signal changing probability of switching α, followed by taking take the average of the total current per cycle and multiplying it by V_{DD}. For example, assuming α = 50% over 4 cycles, then the data signal changes twice.

$$P_{av} = \frac{V_{DD}}{T} \int_0^T i_{DD}(t).dt \quad (2.28).$$

It is important to isolate static power from input driving power, when measuring the required power for clock or data to drive an input terminal. One way of doing this is to place two identical inverters before each input signal, where one of them is only connected to the circuit input terminal capacitance while the other is connected to an open circuit. Then both I_{DD} currents are measured through the inverters and the difference is taken to account for the driving current. The average input terminal power is $(I_{DD\text{-}in} - I_{DD0}) \times V_{DD}$.

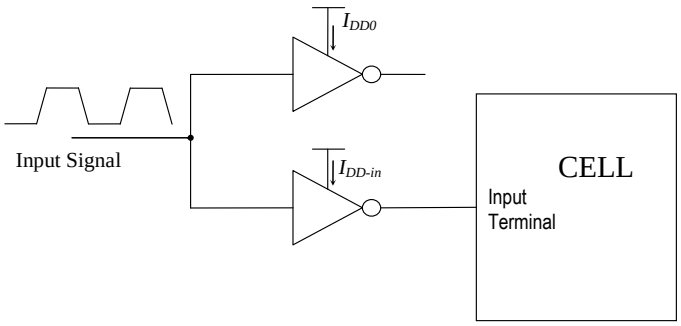


Figure 2.36 Input driving power measurement

2.4.7 Variability Analysis

Variations can be modeled as uniform or normal *Gaussian* statistical distributions, see Figure 2.37. Voltage supply and temperature variations can be modeled using uniform distribution, which specifies an equal probability that all samples will work within specified parameter limits, for example ±10% variation around nominal V_{DD}. The effects of V_{DD} variations and ambient temperature can be simulated using the parametric analysis available in *Cadence Virtuoso Spectre* SPICE-level circuit simulator [56].

A normal distribution is specified around the population mean value (μ) and its

standard deviation (σ). A normal distribution with a variation of one standard deviation (1σ) around the mean incorporates 68.8% of the whole population, whereas 2σ and 3σ include 95.4% and 99.7%, respectively. In general, using process variation of at least three σ must be accounted for, whereas with future process technology challenges at least six-sigma variations analysis will be considered to address the effects of manufacturing process variations, however, it is inappropriate to model the effects of variations in supply voltage and temperature. Process variations are usually modeled as a normal distribution with three standard-deviations around the mean. There are two traditional methods for investigating process-variability tolerance in analog simulation in SPICE-like environments, namely, corner analysis and Monte Carlo analysis.

2.4.7.1 Corner Analysis

Corner analysis is the traditional worst-case model that categorizes all physical and environmental variations into three levels: typical, fast and slow. For process variations, there are five models for combined PMOS and NMOS levels. For voltage supply variations, the three levels refer to nominal V_{DD} for a typical corner, $0.9\times V_{DD}$ for the slow corner and $1.1\times V_{DD}$ for the fast corner. For temperature variations, the three levels refer to room temperature (27°) for a typical corner, low temperatures (0° or −40°) for the slow corner and high temperatures (70° or 125°) for the fast corner. The combination of all these PVT corners creates a total of 45 corners; however, not all of them are needed. Each corner is effective to test a particular condition.

Corner analysis is a straightforward and computationally efficient tool; however, as variations become more significant, it has low accuracy because it does not represent all samples and does not provide an estimation of the yield [4, 9, 10, 45].

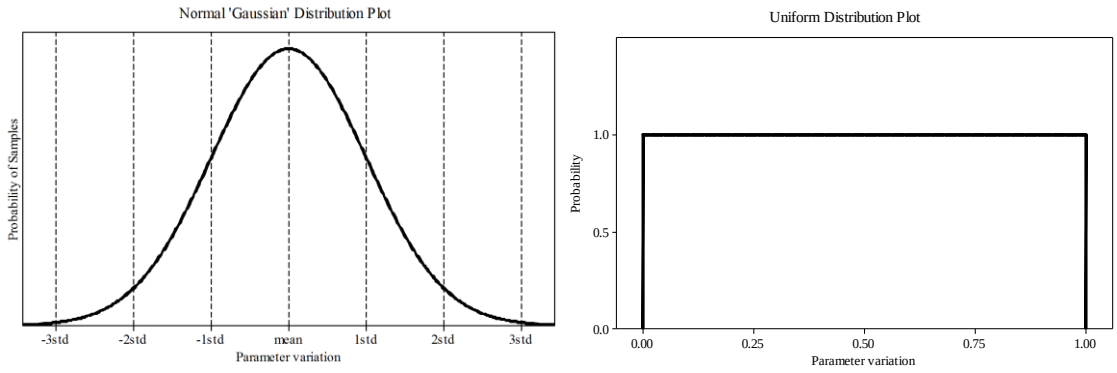


Figure 2.37 Normal and Uniform distribution plots

2.4.7.2 Monte Carlo Analysis

The Monte Carlo statistical analysis provides an accurate and statistical representation of the performance, but its computational efficiency reduces with the increase in circuit size and the number of samples required [4, 9, 10, 45]. It randomizes process parameters according to their offsets in technology model files within a specified number of standard deviations. The *Spectre* simulator [56] is supplied with both tools. In this work, the Monte Carlo method is used when simulating small cells and circuits, and the Corner Analysis method will account for worst case scenarios. Before using either of them, the right technology files must be linked with this analysis. In Monte Carlo, the sigma value, population sample size and type of variation must be defined [56].

2.4.7.3 Cumulative Simulations

Simulating process variability effects on timing and power is straightforward and similar to methods discussed in previous subsections, except in the case of simulating the setup and hold times and the metastability window, because locating them require a lot of simulation runs. Due to the discontinuous failure region shown in Figure 2.33, a large sample fails in the Monte-Carlo analysis, which increases the difficulty of producing enough data about the exponential region to find these parameters. Instead of depending on the normal distribution curve, it is easier to produce a cumulative distribution (CDF) plot, as shown in Figure 2.38, from a series of Monte-Carlo simulation runs at different values of t_{DC}. This is monitored by measuring the output Q voltage after clock-to-Q time to give a stable output. In the Monte-Carlo simulation with total number of samples '*S*', there will be '*N*' simulated samples that are accepted with '*S*–*N*' rejected. Two bins would be sufficient to record this information. Then, the simulation should be repeated with a different t_{DC}, and recording the value of *N*. At the end, a graph is plotted of the predefined D-to-clock time, that is setup time or hold time, against *N*, which gives a CDF plot of normal distributions similar to Figure 2.38, where *N* of 50% is considered the mean, and the positive and negative standard deviation at *N* around 15% and *N* around 85%. Using this method, variation of setup/hold and window times can be measured and presented statistically.

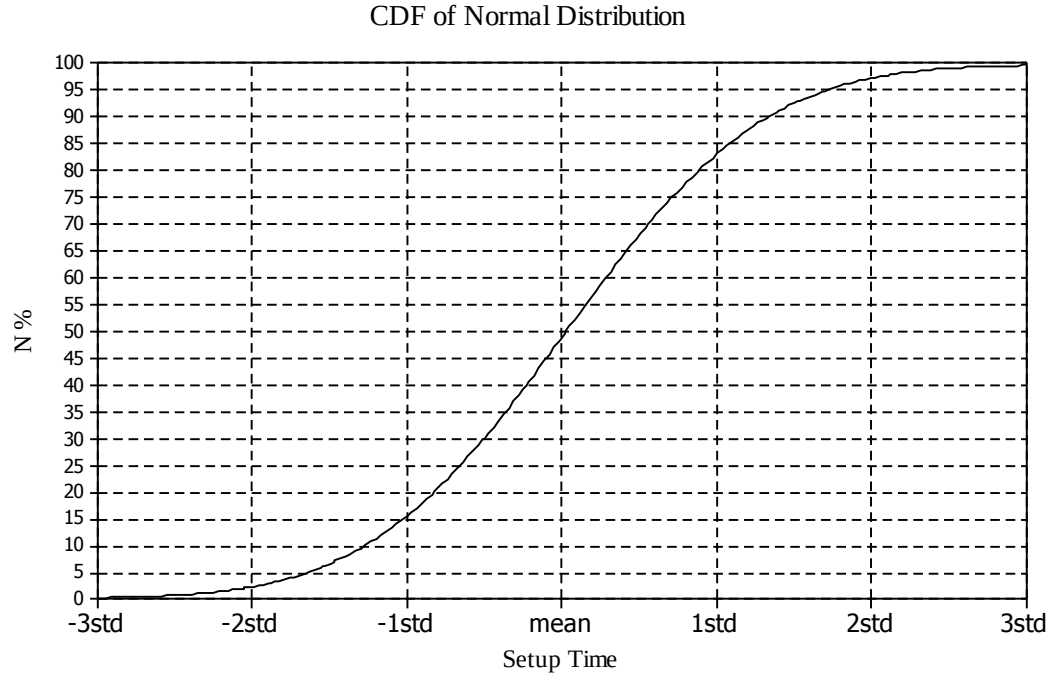


Figure 2.38 Cumulative normal distribution

During mismatch variation analysis, the value of τ should not be simulated using the switch method (Figure 2.35). It is better to produce an input-time difference (t_{DC} or Δt_{in}) against output-time delay (t_{CQ} or t_{out}) curve, similar to Figure 2.38, and then measure τ using Equation (2.23), and time metrics.

2.5 Summary

This chapter reviewed the metastability behavior in synchronizer and arbiters. Then, it discussed the current approaches for synchronizer and arbiter design. Following that, it discussed different synchronization methods in SoC and the synchronization concerns between multiple voltage domains. Finally, the chapter defined the flip-flop metrics and explained their characterization methods.

Having reviewed process, voltage and temperature variations and metastability behavior for synchronizers and arbiters, the subsequent chapters in this thesis will describe the work undertaken to address the issues raised and to advance the state of the art in synchronizer and arbiter design for use in SoC with MCD and MVD.

Chapter 3 The Trade-off between Resolution Time and Delay Time in Flip-Flops

Flip-Flop cells used in an edge triggered register need a minimum time between input D transitions and the rising edge of the clock, usually known as the setup time, and the D input must be held steady after the clock edge called the hold time. In a pipeline based processor, this setup time plus hold time represents a dead period in each pipeline stage during which the D cannot change and therefore prevents useful stage processing taking place in part of the clock cycle. Processing time is reduced in a similar way if the delay from the rising clock to a valid output Q (Clock to Q), is longer than the hold time. Here processing cannot start until the Q value from one stage of a pipeline is available, and must finish before the setup time of the next stage. The relevant times are illustrated in Figure 3.1 below.

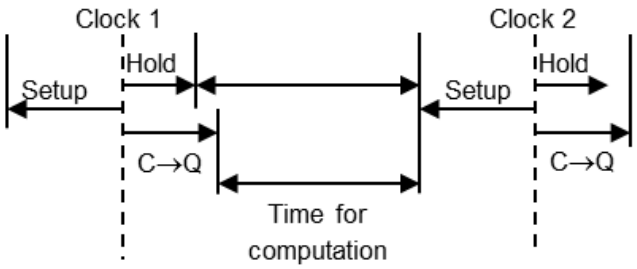


Figure 3.1 Setup-time, hold-time and clock-to-Q time

Edge triggered flip-flops are also used in synchronizers to allow data to pass securely between processors with unrelated clocks. In this application, the relative timing of the two clocks may be unknown, it is possible for input data generated from one clock to violate the setup and hold times of the synchronizer flip-flop. When this occurs the most important parameter is the recovery time of the flip-flop τ, and how much time t, needs to be allowed to reduce the failure rate of the synchronizer to an acceptable level. Thus, the conditions for optimum

performance in a register flip-flop may not be the same as those for a synchronizer, and ideally, the circuit details should not be the same.

The structure of this chapter is as follows. The first section, introduces the basic design of flip-flops in the context of their two main applications, data registers and synchronizers, and shows that the separation of the concerns of delay and recovery can lead to different optimizations. The next section compares these designs with respect to the two main characteristics, D to Q delay and available recovery time. After that, the previous principles are described on other circuits where it can be optimized to give at least as good a performance as conventional designs as either a register or a synchronizer. Thereafter, the concept of 'wagging' is introduced and its effect on synchronizer design is described. Finally, the 'wagging' synchronizer design is compared with the typical flip-flop synchronizer and the results are discussed.

3.1 Flip-Flop Design

Figure 3.2 shows a conventional edge triggered D flip-flop (ETDFF) design, similar to those described as a typical Master Slave Flip-Flop (MSFF) in [85, 92] and as a standard cell flip-flop in [116]. This design was chosen as the base because it is commonly used, and is amongst the best performing circuits in a recent study [92]. A switched buffer, whose circuit is shown on the left in Figure 3.3 drives the master latch input node when the clock is low, and the master latch drives the slave latch through a second switched buffer when the clock is high. The feedback inverters in the keeper latches are clocked to weaken them while writing new data. Figure 3.3 can be simply transformed into an inverter plus a switch as shown on the right by simply adding a connection between the sources of the clocked transistors, and new designs produced. Figure 3.4 shows a Transmission Gate Flip-Flop (TGFF) produced by replacing the switched buffers with switches and eliminating the intervening inverters. This configuration can be faster but consumes more static power than the switched buffer because of the shorter path between V_{DD} and ground in the inverters. The TGFF is considered the fastest flip-flop [85, 92] because there are only two inverters in the path from D to Q. However, in the case of local process variations, data could be lost at the second switch between the master latch and slave latch due to loading of the master by the

slave when the switch is closed. This happens if the clock goes high with the master latch output high and the slave latch input low, because the master output inverter has to drive the slave feedback inverter and the output buffer input capacitances to a high voltage. Simulations show that a weak p-type transistor in the master output inverter combined with a low threshold master feedback inverter prevents this voltage from reaching a high enough level to retain the master state. This disadvantage becomes more serious as process variability and the number of flip-flops used in the design increases.

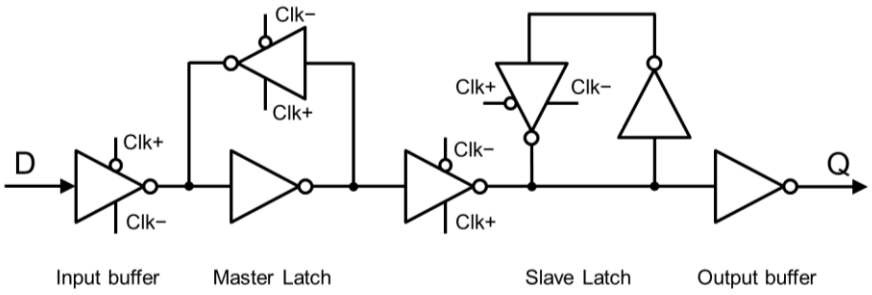


Figure 3.2 Master Slave Edge Triggered D Flip-Flop (ETDFF)

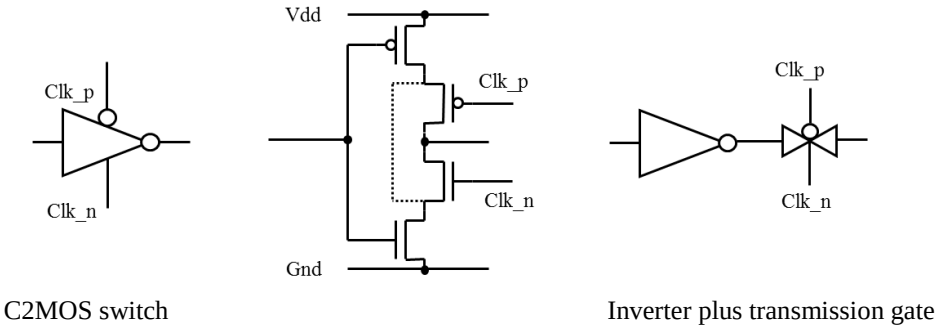


Figure 3.3 Switched inverting buffers

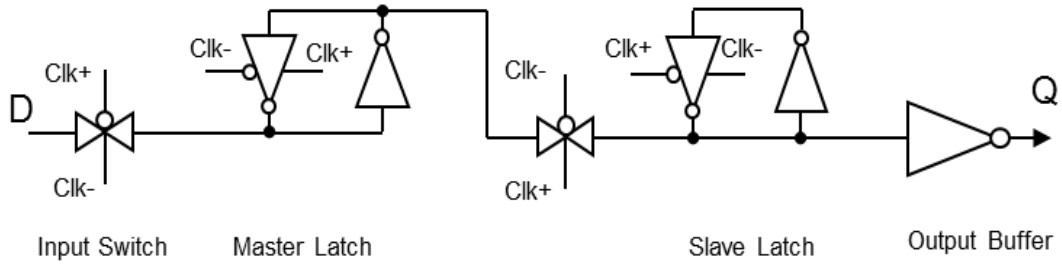


Figure 3.4 Master Slave Transmission Gate Flip-Flop (TGFF)

An alternative structure is shown in Figure 3.5. This arrangement is similar, but not the same as the Capture-Pass latch described by Sutherland [117], and follows the principle of the Double Edge Triggered Flip-Flop (DETFF) [93]. The novel concept of a wagging flip-flop is proposed in which the upper latch is driven when the clock is high and the Q output is taken from the lower latch whose value is held

constant. When the clock goes low the lower latch is driven, and the upper latch value held and passed to the Q output, thus whenever a clock transition occurs a new D value is captured and the previous value outputted. This means that the clock frequency is half that of Figure 3.2 where only the rising edge of the clock produces a new output. There are several advantages to this design, firstly it presents a shorter and potentially faster path through from D to Q than the conventional cascaded master slave design, and secondly the dynamic power consumption is less because although there are more clocked buffers (six rather than four) the clock changes at half the frequency, leading to only six power consuming transitions, rather than eight. Finally, and most importantly, this design separates the concerns of storage of the D value in the latch, from the input/output considerations.

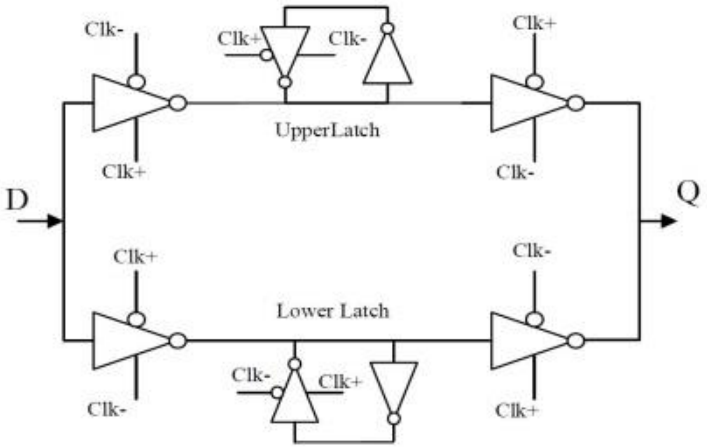


Figure 3.5 Dual Edge Triggered Flip-Flop (DETFF).

This separation of concerns allows the delay path from D to Q to be designed independently from the latch recovery. For a short delay the input buffer must be strong enough to set or reset the latch node quickly, driving the capacitance presented by the latch, which should therefore be weak to allow a fast response. On the other hand, if the input signal changes within the setup and hold time, the latch node may not be fully charged and the function of the latch is to complete the transition to a high or a low as quickly as possible. In this case the latch inverters need to be strong, and the input and output buffers weak to present a low capacitive load to the latch.

Consider two cases where DETFF can be used: storage and synchronization. In the first case, the delay path from D to Q can be considered independently from the latch recovery, which makes it easier to trade off recovery time against

throughput. For a short delay from D to Q the input buffer must be strong enough to set or reset the latch node quickly, this is achieved by strongly driving the capacitance presented by the latch, which should therefore be weak in to allow a fast response. On the other hand, if the input signal changes within the setup and hold time, as in the case of synchronizers, the latch node may not be fully charged and the function of the latch is to complete the transition to a high or a low value as quickly as possible. In this case the latch inverters need to be strong, and the input and output buffers weak to present a low capacitive load to the latch.

In a register, the most important consideration is to reduce the setup and clock to Q time, but the recovery time τ is almost unimportant since the latch is only there to provide static memory. In this case the latch size can be reduced to the minimum. In a synchronizer, the opposite is true, setup and hold times are less important than the time required to recover from an input event, which occurs within the setup plus hold window, which is analogous to T_w.

3.2 Flip-Flop Results

By means of a series of SPICE-level simulations the output Q, setup time, and hold time from 50% values to 50% of the clock was measured as well as the τ time constant in the circuits in Figure 3.2 and Figure 3.5. All circuits were modeled in the UMC 90nm process with all p-type transistors having twice the width of the n-type transistors; so that a buffer width of 1μm means that the n-type transistors were 1μm wide and the p-type transistors 2μm. The ratio of input to output buffer width to latch width was varied from 10:1 to 1:10. In Table 3.1 the total time required from the D setup to clock and then to Q output (D-Q) is shown as the buffer: latch ratio changes. Also included, is a 2μm inverter load on the output of all circuits to ensure a fair comparison, and an estimation of the setup and hold times was obtained by measuring the D to clock time necessary to give a 10% increase in clock to Q time.

The results, in Table 3.1, indicate that this non-useful time reaches a minimum of 52.56ps, when the latch is small at 0.2μm and the buffers large at 2μm, showing that the latch function is simply to hold the flip-flop state, and in this case, where the D input change does not violate the setup and hold conditions, its size should

be minimized. In Table 3.2, the setup time plus hold time is shown for differing buffer to latch width ratios. The figures given are the average of High to Low and Low to High transitions and show that as this ratio decreases, the load presented by the latch increases and consequently increases the time required for setup. Since the hold time is in fact negative, and also increases, the setup plus hold time is the difference between two larger quantities, and is difficult to measure accurately. In practice, it is quite small and amounts to less than 55ps even when the buffer width is minimum and the latch width is maximum. In these circuits the setup plus hold time is always less than the D to Q time so that the time lost from the clock cycle due to the register is dominated by the D to Q time. In a synchronizer application the setup plus hold time is equivalent to the metastability window T_w, and ideally it should be minimized, but the 10:1 variation in Table 3.2 and Table 3.3 shows that it is masked in practice by the much larger effect of τ variation.

Table 3.1 Setup plus Clock to Q time

D to Q Delay Time		Latch Width (μm)			
		0.2	0.4	1.0	2.0
Buffer Width (μm)	0.2	126.93ps	139.61ps	176.02ps	236.0ps
	0.4	88.4ps	95.74ps	117.25ps	153.03ps
	1.0	62.18ps	66.37ps	77.84ps	97.2ps
	2.0	52.56ps	55.26ps	62.52ps	72.92ps

Table 3.2 Setup plus hold time

Setup + Hold Time		Latch Width (μm)			
		0.2	0.4	1.0	2.0
Buffer Width (μm)	0.2	29.00ps	32.90ps	43.35ps	54.85ps
	0.4	32.30ps	36.40ps	47.10ps	58.35ps
	1.0	29.15ps	31.55ps	37.70ps	49.2ps
	2.0	23.90ps	26.10ps	30.90ps	37.90ps

Table 3.3 Resolution time constant τ

Resolution Time Constant		Latch Width (μm)			
		0.2	0.4	1.0	2.0
Buffer Width (μm)	0.2	11.35ps	9.94ps	8.78ps	8.28ps
	0.4	12.61ps	10.98ps	9.21ps	8.44ps
	1.0	15.62ps	12.87ps	10.66ps	9.24ps
	2.0	18.35ps	15.4ps	11.97ps	10.49ps

The effect of buffer to latch width ratio on the resolution time constant τ is shown in Table 3.3. Here the minimum value of τ is 8.28ps and is found where the buffer width is a minimum at 0.2μm and the latch width is a maximum 2μm. According to Equation (2.14), a synchronizer with f_c = f_d = 1GHz, and a T_w of 50ps needs a t of 35τ to give an MTBF of greater than a year. If one clock cycle is used for synchronization, the minimum time for that cycle is given by the time from the change in output Q in the first synchronizer flip-flop to the input setup time of the second flip-flop plus 35τ. Here the minimum value for that time is 420.6ps, found with a latch width of 2μm and buffer widths of 1μm. Clearly increasing both widths by a factor of 2 would further reduce the synchronization time (or increase the MTBF if the cycle time was held constant) though the power dissipation would also increase.

The flip-flop in Figure 3.5 was compared with the standard cell edge triggered design on a like for like basis by using only 1μm buffers and switched buffers for both designs. These results, shown in Table 3.4 indicate that the double edge triggered wagging flip-flop is faster than the conventional circuit both in a register application and in a synchronizer application and also has a lower clock power dissipation because the clock frequency is lower. It is also possible to take advantage of the separation of concerns to produce a significantly faster register flip-flop by reducing the latch to 0.2μm. In this case the D to Q time drops from 78ps to 62ps, and the power dissipation also drops, as indicated in Table 3.4. It is not possible to do this in the conventional circuit of Figure 3.2 because the latch circuit is in the D to Q path, and would slow the Q output rather than speed it up.

In the TGFF circuit, a weak master latch resulting from local process variability could cause stored data to be lost. This happens when the master latch cannot hold the high state as a result of the slave capacitance loading when the middle switch is closed. If the slave latch is in the low state, a high node capacitance may overcome the master and pull the master latch node towards ground. In this case, the TGFF fails. Since the number of flip-flops in a VLSI system is very large and the probability of local variations increases with new technology nodes [16], the probability of having TGFF with weak latches increases. This particular failure mode is absent in the DETFF.

Table 3.4 Flip-Flop Comparison.

Parameter	Flip-Flop Type			
	ETDFF	TGFF	DETFF	
Latch Width	1µm	1µm	1µm	0.2µm
Setup time	49.15ps	29.1ps	55.30ps	40.8ps
Hold time	-31.4ps	-6.57ps	-17.60ps	-15.5ps
Window	17.75ps	22.53ps	37.70ps	25.30ps
Clock to Q time	45.53ps	39.8ps	22.54ps	21.4ps
D to Q time	94.68ps	68.9ps	77.84ps	62.2ps
τ - Master	12.23ps	14.93ps	10.66ps	15.62ps
τ - Slave	10.83ps	11.13ps		
Dynamic Power	16.97µW	16.28µW	20.44µW	14.30µW
Energy	1.6pJ	1.1pJ	1.6pJ	0.89pJ
Static Power	53.7nW	42.5nW	34.3nW	19.2nW

3.3 Other Cell Examples

The principle of separating the recovery time τ from the Input/output time T_w can also be applied to the specially designed Jamb latch synchronizer circuit [74], shown in Figure 3.6 in which the cross-coupled inverters and the set/reset mechanism has been reduced to a minimum to enhance the recovery time constant.

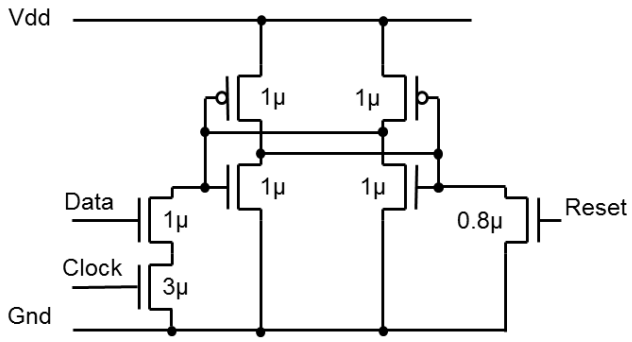


Figure 3.6 Jamb latch Synchronizer.

Using the UMC 90nm process the circuit of Figure 3.6 achieves a τ value of 7.5ps, significantly faster than either of the flip-flops discussed earlier. The barrier to further improvement is that the set and reset transistors cannot be reduced in size because they would then be unable to pull down the latch nodes. Another problem for all synchronizer circuits in future processes is the lower V_{DD} associated with

lower power circuits and processes. Low supply voltages means low transistor currents at metastable levels giving low g_m and high τ. Both of these problems can be alleviated by separating the set/reset function from the metastability recovery, and while Table 3.2 shows there is a penalty in T_w it is more than offset by the improvement in τ.

Figure 3.7 shows a robust synchronizer circuit in which the size of the p-type latch transistors has been reduced to 0.25μm width. This allows the Data, Clock and Reset transistors to be smaller, but the lower feedback gain would normally increase the recovery time constant τ. In this circuit the presence of metastability is detected, and two extra p-type transistors are switched in to increase the current and hence improve g_m . This produces and a τ value of 8.1ps at a nominal V_{DD} (1V) and temperature 27°C. At low voltage (0.7V) and low temperature (-25°C) the jamb latch performance is severely degraded, so the circuit in Figure 3.7 has significantly better performance than the conventional Jamb latch with a τ of 12.5ps compared with 25.7ps for the Jamb latch.

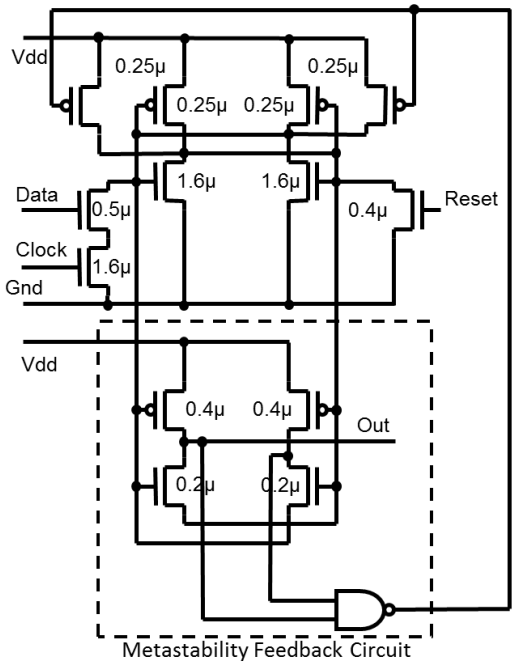


Figure 3.7 Robust Synchronizer with low buffer to latch ratio

The C-element is another bistable circuit commonly used in asynchronous systems. A simple two input C-element is shown on the left of Figure 3.8, in which output goes high after the two inputs X and Y both go high, and after X and Y both go low, the output goes low. If X and Y are different the previous state is be maintained. In

the circuit on the left of Figure 3.8, state holding is carried out by the node capacitance, which provides a dynamic memory. In some cases, dynamic memory is insufficient, and the state must be held by a static latch as shown by the circuit on the right of Figure 3.8. Here the latch is transparent when X and Y states are the same and opaque when they are different. The structure now operates as a switched input buffer followed by a latch and an output buffer in the same way as the DET flip-flop. The same trade-offs between delay and recovery time can therefore be expected.

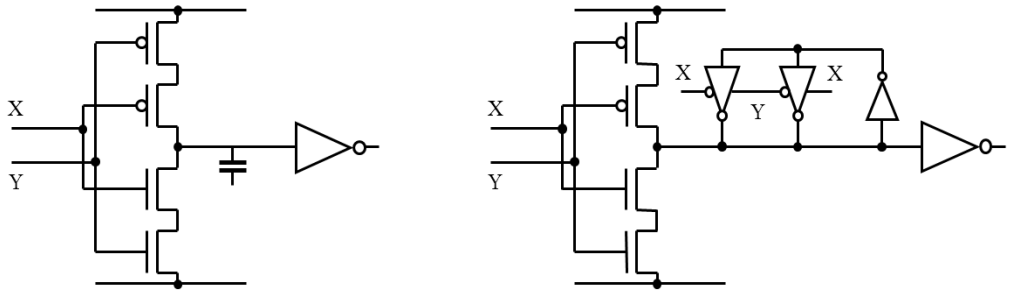


Figure 3.8 C-elements

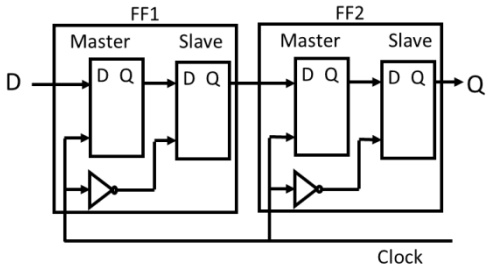
Normally a C-element does not get into a metastable state, and the width of the latch should be minimal, but there are situations, for example when a transient error forces the central node to a half level, which can cause a spurious pulse to appear on the output. This pulse can then propagate through a chain of C-elements causing multiple upsets [118]. In this case recovery time becomes important and the faster the recovery the less likely the pulse is to propagate any significant distance, so a wider latch may be required.

3.4 Reliable Synchronizers

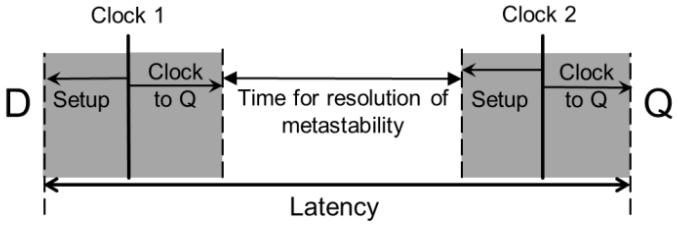
3.4.1 Flip-Flop Synchronizer

A conventional synchronizer typically comprises two flip-flops connected in series, FF1 and FF2, where each flip-flop has a master and a slave latch. Previously discussed latch circuits could be used as the master and slave latches of each flip-flop. This is shown in Figure 3.9(a). This configuration is used to reduce the probability of metastable events occurring in the input flip-flop FF1 from progressing into the system. In this configuration, there is one clock cycle between capturing the state of the input, resolving metastability, and holding the result in the output flip-flop FF2. If the time available to resolve metastability is not

enough, based on Equation (2.14), a synchronizer failure may occur quite frequently. The time available to resolve metastability is less than one clock cycle, due to the clock to Q time delay taken by the master latch, the time taken to pass through the slave latch, and the setup time for the following slave flip-flop, FF2. This time effectively adds up to two D to Q time delays, which can be a significant part of the clock cycle.

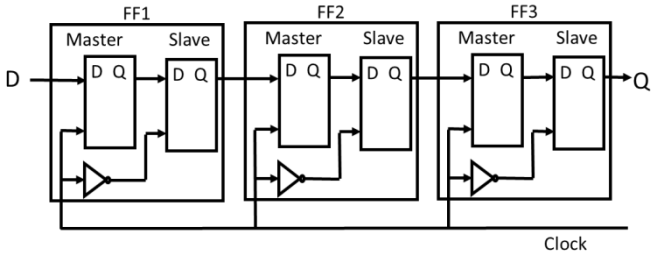


(a) A 2FF circuit

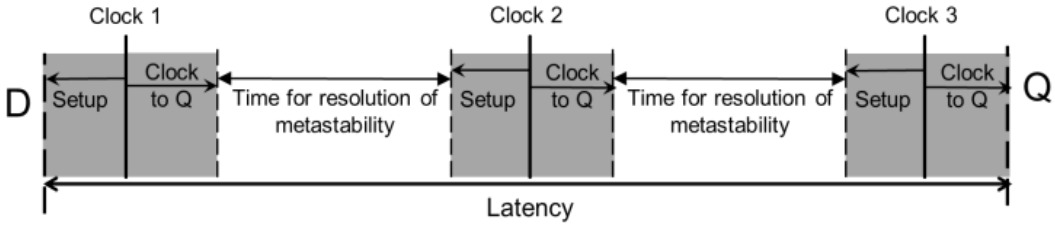


(b) The available resolution time in a 2FF synchronizer

Figure 3.9 A 2FF pipeline synchronizer



(a) A 3FF circuit



(b) The available resolution time in a 3FF synchronizer

Figure 3.10 A 3FF pipeline synchronizer

If the reliability of the two flip-flop synchronizer is insufficient within a single clock cycle, a third flip-flop is often added as in the top of Figure 3.10(a). In this scheme any remaining metastability is passed on from FF2 and FF3 and resolved in the

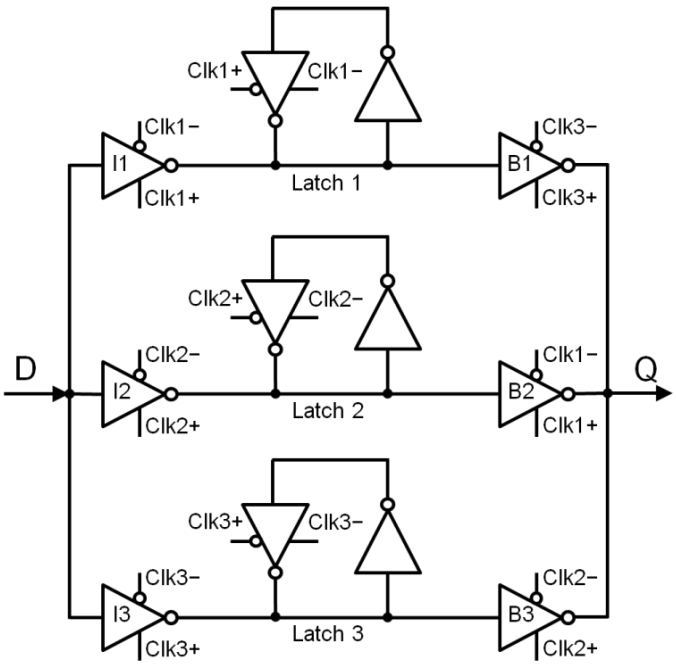
next cycle while another sample in the input is taken by FF1. This maintains the throughput of the synchronizer at the cost of two cycles of latency but has the disadvantage of adding another D to Q time delay.

In the two flip-flop synchronizer, the available resolution time t_R is limited by the clock cycle T_C and lost time in the input to output path. This lost time is equivalent to the clock to Q time in FF1 and the setup time in FF2 as shown in Figure 3.9(b). Whereas, in the three flip-flop synchronizer, the available resolution time is two clock cycles reduced by two D to Q times, as shown in Figure 3.10(b). In general, for a series pipeline synchronizer composed of N flip-flops, the available time for resolution and latency are shown below in Equation (3.1) where (N – 1) is the number of resolution cycles.

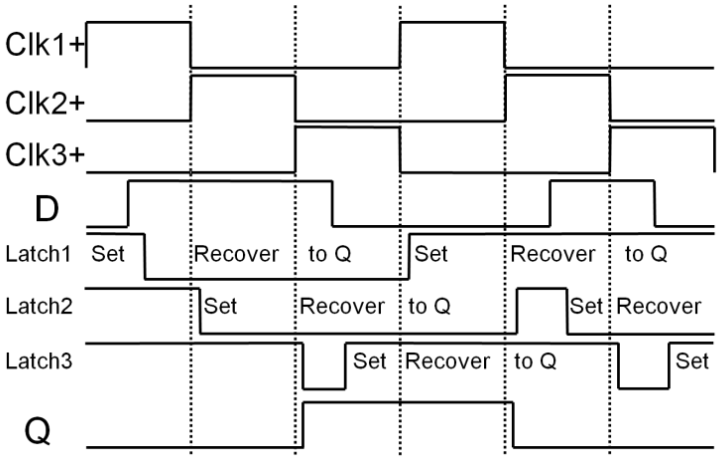
$$\left.\begin{aligned} t_R &= (N-1)\cdot T_C - N\cdot t_{DQ} \\ Latency &= N\cdot t_{DQ} + t_R \end{aligned}\right\} \quad (3.1).$$

3.4.2 Wagging Synchronizer

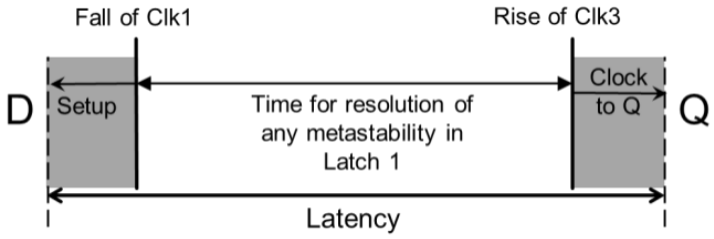
An alternative structure, proposed in [119], is based on the wagging principle, and similar to the approach presented in [120]. This is shown in Figure 3.11 and called the 'wagging' synchronizer. This structure is a three-way wagging synchronizer, which uses three similar paths controlled by three clock phases. Each path has a switched buffer/latch and a switched output buffer, where all buffers drive the output node Q. The input buffer/latch and the output buffer are controlled by two clock phases from the three phases (Clk1, Clk2 and Clk3), as shown in Figure 3.11(a), where each clock phase pulse is equivalent to one clock cycle of the receiver clock frequency and each clock phase is non-overlapping with the others, that is, the rising and falling edges of each clock phase must coincide with edges of the preceding or the succeeding clock phases, in other words, the rising edge of each clock can start to transit after the falling edge of the preceding clock phase has fallen. Each path pair has a different clock signal combination. In Figure 3.11(a), Clk1 drives input buffer I1, latch 1 and buffer B2, whereas Clk2 drives I2, latch 2 and B3, while Clk3 drives I3, latch 3 and B1. All latches are identical and have the same value of τ and setup and delay times. One-micron inverters and 1μm switched inverters have been used to construct the wagging synchronizer.



(a) Circuit structure of the 'wagging' synchronizer



(b) Circuit operation



(c) The available resolution time in the wagging synchronizer

Figure 3.11 Three-way wagging synchronizer

The aim of the wagging synchronizer is to increase the time allowed for metastability to resolve, hence improve the synchronizer reliability. As shown in Figure 3.11(b), when Clk1 is high, latch 1 is set to a new value of input D, while B2 drives the value stored in latch 2 to the output node Q, whereas latch 3 is allowed to recover from any metastability for one clock cycle. Similarly, during

Clk2, latch 1 recovers while latch 2 is set and latch 3 drives Q. In Clk3 phase, latch 2 recovers while latch 3 is set and latch 1 drives Q. The only reduction in the clock cycle time allocated to recover from metastability is the clock to Q time of the latch, and this slightly reduced time is always available in one path, while the D input is stored in another and Q is read in a third.

Figure 3.11(c) indicates the available resolution time t_R for the wagging synchronizer is limited by the clock phase width τ_C and lost time in the input to output path. Following setup, all of the time between the fall of Clk1 and the rise of Clk3 is available for the resolution of metastability. One property of the wagging synchronizer is that it can be expanded to an *N* way wagging synchronizer, (where $N \geq 3$), which expands the available resolution time without the penalty of additional D to Q time delays. The resolution time and latency of the *N*-way wagging synchronizer can be expressed in Equation (3.2), where (N – 2) is the number of resolution cycles. A general schematic of it is exemplified in Figure 3.12.

$$\left.\begin{aligned} t_R &= (N-2)\cdot T_C - t_{DQ} \\ Latency &= t_{DQ} + t_R \end{aligned}\right\} \qquad (3.2).$$

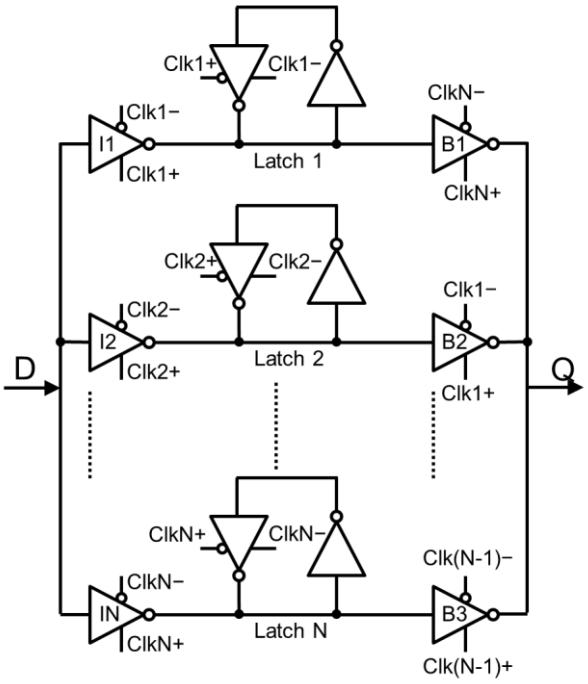


Figure 3.12 *N* way wagging synchronizer

3.4.3 Clocking Control Circuit (CCC)

One requirement of the wagging structure is that clock phases must be ordered

and non-overlapping. To maintain the relationship between the *N* clock phases for an *N* way wagging synchronizer the proposed solution uses the Signal Transition Graph (STG) with the required functionality in Figure 3.13. In this STG, the signal Clk is the input clock signal, which indicates the receiver frequency, whereas signals Clk1, Clk2 and Clk3 are the output clock phases required to drive the 3-way wagging synchronizer. Internal signals S1, S2 and S3 are based on a timing assumption [121] that the negative pulse of the input clock signal is long enough to make two signal transitions before the rising edge of the second clock cycle, i.e. the transitions {Clk−/1 → S1+ → S3− → Clk+/2} must maintain their sequence.

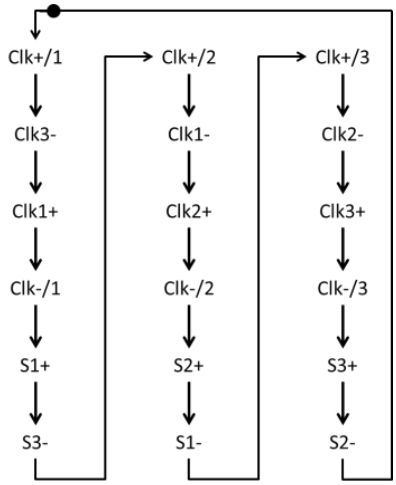


Figure 3.13 STG for CCC

The STG in Figure 3.13 was synthesized and the sequential circuit in Figure 3.14 is proposed to control the clocking of a 3-way wagging synchronizer. This circuit implementation uses symmetric optimized OAI gates and inverters, which have symmetric delays between signal transitions. In other words, the time required from Clk+/1 to Clk1+ is equivalent to the time from Clk+/2 to Clk2+ as well as the time from Clk+/3 to Clk3+. This is also true for the case between transitions from Clk−/1 to S3−, Clk−/2 to S1− and Clk−/3 to S2−. The timing diagram of the control circuit signals with data signal D and output Q in the wagging synchronizer is shown in Figure 3.15. The output clocks of this circuit are buffered to drive the wagging synchronizer.

The circuit has a minimum functional frequency due to the timing assumption in the STG. This timing restriction between Clk−/1 and Clk+/2 has to be at least 130ps at nominal operation. This gives a minimum clock period of 260ps ($f_{CLK(MAX)}$ ≈ 3.8GHz) at 1.0V supply voltage and no process variations. The circuit produces a

delay of 85ps to produce a clock signal from the rising edge of the input clock the rising edge of the next clock phase. A 53ps portion of the 85ps delay is to ensure that the previous clock phase signal has fallen to logic '0' before the rise of the next clock phase signal. This is to maintain the non-overlapping of output clock signals. The pulse width of the output clocks is less than the clock cycle by 53ps, which is the delay between the adjacent clock phases. The period of the clock phase is three times that of the original input clock, that is, 780ps at the minimum clock period in this case.

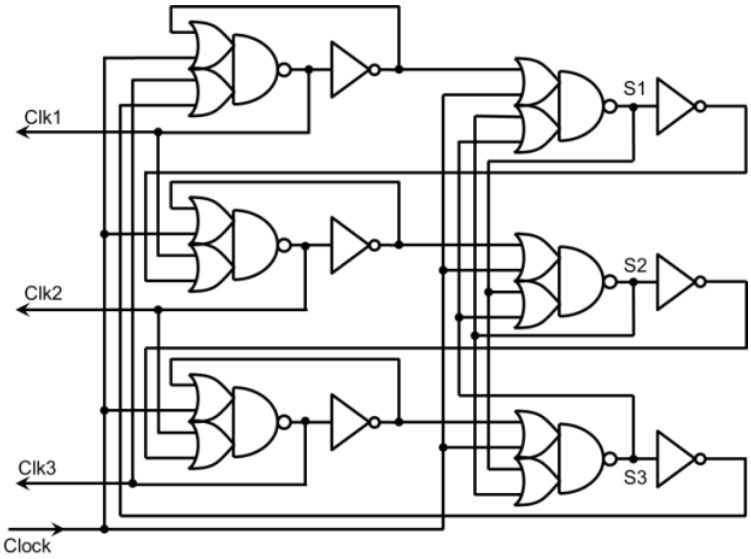


Figure 3.14 The proposed Clocking Control Circuit (CCC)

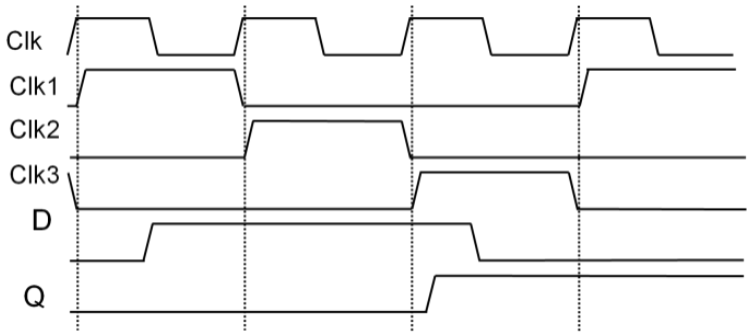


Figure 3.15 Timing diagram of 3 clocking signals with the wagging synchronizer

The design of the control circuit can be expanded for *N* clock phases. This can be done by adding extra sequences in the STG diagram, shown in Figure 3.13, for each additional signal of the clock phases. For example, if it is intended to design a control circuit for a 4-way wagging synchronizer, the sequence {S2– →Clk+/1 → Clk3– } can be replaced in the STG by the following sequence: {S2– →Clk+/4 → Clk3– → Clk4+ → Clk–/4 → S4+ → S3– →Clk+ → Clk4–}. Then, a new circuit can be

synthesized in a similar fashion to the circuit presented in Figure 3.14. The cycle of the clock phases in this case is four times that of the input clock signal.

3.5 Synchronizer Results

Based on Table 3.5, the lost time in the conventional design based on an ETDFF is 95ps. In the wagging synchronizer, no time is lost from the resolution period. If the clock period in each case is 400ps, the available resolution time is 305ps for the ETDFF two flip-flop synchronizer, and 322ps for the DETFF combination. These times are approximately 26.5τ and 30.2τ, respectively. From Equation (2.14), if f_c = f_d =2.5GHz, with t_R =26.5τ gives an MTBF of around 50mins. Replacing the ETDFF with a DETFF would give an MTBF of about 15.5 hours. On the other hand, the wagging synchronizer has a longer time available for recovery. For the same 400ps period, the available resolution time is about 37.5τ, which is equivalent to an MTBF of 2.66 years for the same values. The reliability of the wagging synchronizer is thus significantly better than the conventional structures.

Table 3.5 Comparing two flip-flop and wagging synchronizers

Parameter	Two FF Synchronizer		Wagging
	ETDFF	DETFF	Synchronizer (WS)
Setup time	49.15ps	55.30ps	55.30ps
Clock to Q time	45.53ps	22.54ps	28.97ps
τ	11.5ps [§]	10.66ps	10.66ps
Estimated T_w	17.75ps	37.70ps	37.70ps
Available t_R [§§]	305ps≈ 26.5τ	322ps≈ 30.2τ	400ps≈ 37.5τ
Total D to Q time	189.36ps	155.68ps	84.27ps
MTBF [§§§]	49.6mins	15.5 hrs.	2.66 years
Latency for 40τ	650ps	582ps	511ps

[§] Averaged value. [§§] T_{CLK}= 400ps. [§§§] f_c= f_d=2.5GHz.

Equation (3.1) and Equation (3.2) define the latencies and the resolution time t_R. In a two flip-flop synchronizer, latency is the addition of the resolution time t_R to the total path D to Q delay through both flip-flops, which equals two times D to Q time. On the other hand, the latency of a wagging synchronizer is the resolution time plus only one D to Q time. Table 3.5 shows that the total path delay from D to Q of the new design is much less than that of the old one.

Another measure of the synchronizer's effectiveness is the total latency for a given

resolution time. If the latency of a synchronizer is calculated using Equation (3.1) and Equation (3.2) with a 40τ resolution time, the ETDFF in a two-flip-flop synchronizer requires two D to Q time delays, 95ps each plus the 40τ resolution time of 460ps, a total of 650ps, whereas the wagging synchronizer reduces this to 511ps, a 20% improvement.

The wagging synchronizer can easily be extended from a single cycle resolution time to two cycles by adding a further latch to the three shown in Figure 3.11(a). This then allows one latch to be loaded while two are resolving and the fourth is outputting, thereby improving the reliability of the synchronizer. The effect of this extension on latency is different for the two types of synchronizer considered here. According to Equation (3.1), a three flip-flop synchronizer with a 40τ resolution time, which is split into two periods one between FF1 and FF2 and other between FF2 and FF3, incurs an additional D to Q time delay, leading to 745ps latency. In contrast, from Equation (3.2), a four latch wagging synchronizer only requires an additional 4ps for the extra output buffer fan in, or 515ps latency. Therefore, the relative improvement for the wagging synchronizer is 30%, which relaxes the restriction of operating at a higher clock frequency, this is shown in Figure 3.16 and Figure 3.17, however, the wagging synchronizer is restricted by the maximum clock frequency of the clocking control circuit, that is, 3.8GHz.

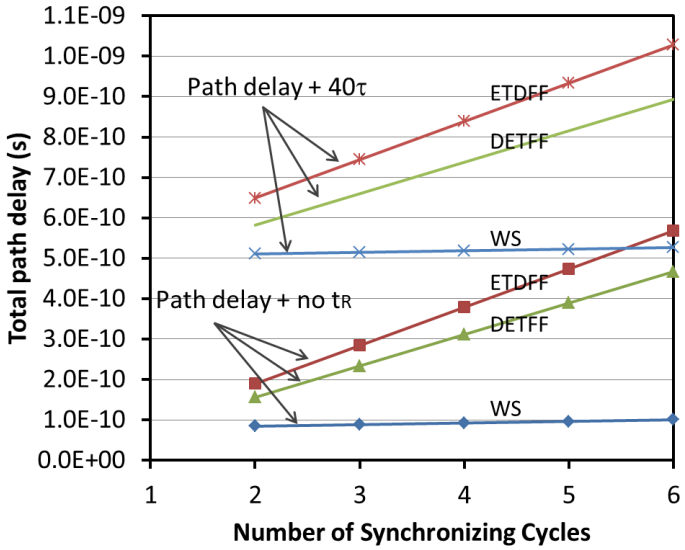


Figure 3.16 Total path delay against the number of cycles

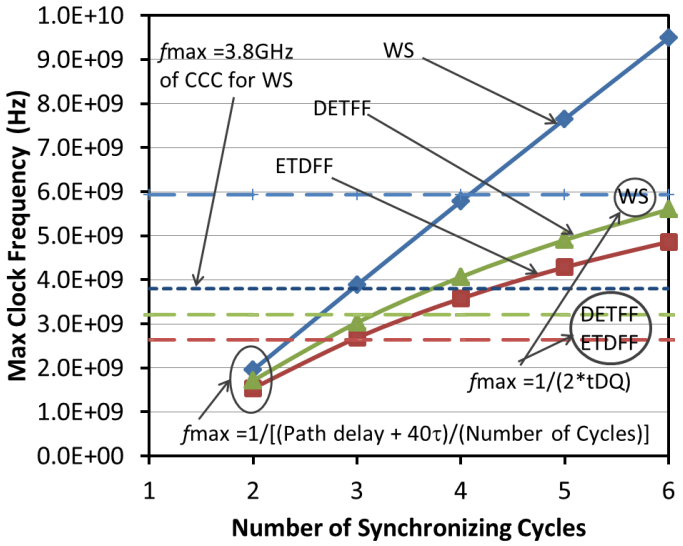


Figure 3.17 Maximum clock frequency against the number of cycles

3.6 Summary

The DETFF can perform at least as well as the common edge triggered master slave configuration in a register application. For similar transistor widths, it is faster and has a lower clock-tree power dissipation because the number of clock transitions in halved, but its main advantage is the 25% shorter D to Q time by reducing the latch size to the minimum, an option not available to the more conventional master slave configuration.

If the concept of wagging is also applied to the synchronizer structure itself, only a single latch is necessary to capture the state of the input. This significantly shortens the path from unsynchronized input to the synchronized output when compared with the conventional two flip-flop synchronizer. The proportion of time available for resolution of metastability is also increased, and the total latency reduced by 20% compared with a two flip-flop synchronizer and 30% for a three flip-flop synchronizer. This allows a reliable wagging synchronizer to be built with significantly lower latency than more conventional designs.

In a synchronizer application, the DETFF circuit does not suffer either from the additional delay brought about by the back edge of the clock [77, 114], or the additional complication of two different τ values for the master and the slave, as both clock transitions are used to output a new Q value and there is no back edge.

Chapter 4 Variation-Tolerant Arbiter Design

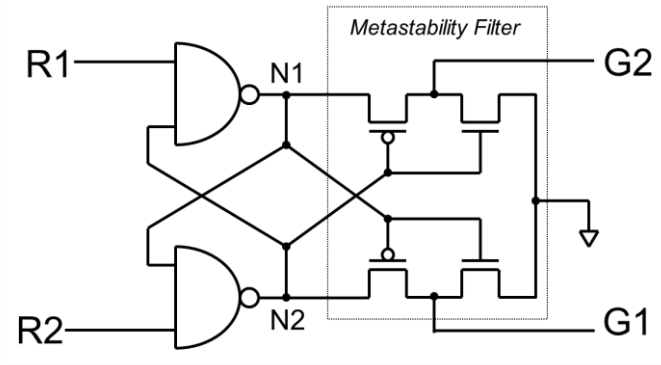
Arbiters are necessary to distribute data between processing elements and different resources within a system. For example, in a bus system with two CPUs and memory without an arbiter, both processors might need to read or write to the memory at the same time; this leads to conflict and data loss if both CPUs access at the same time. However, if an arbiter was used between the CPUs and memory, it will arrange their access to the memory one after the other. As process parameter variations will have a considerable impact on metastability resolution time. Therefore, any increase of variability of PVT within a chip will increase the variations in the metastability behavior in arbiters and synchronizers, and eventually put the system at risk of critical delays and increases in power dissipation.

In this chapter, the work is focused on the MUTEX, the fundamental cell within the arbiter, concentrating in general terms on performance and in particular in relation to metastability. As mentioned in Chapter 2, the metastability behavior in bistable circuits is highly dependent on technology and environment, for example, it worsens significantly with lower supply voltages [80, 86] and increased loading [50, 74]. That is why the study also concentrates on tolerance against Process, Voltage, and Temperature (PVT) variations supported by simulation results. The study covers the typical MUTEX and a number of modified MUTEX circuits, some of which showed little or no improvement, while others showed significant enhancement.

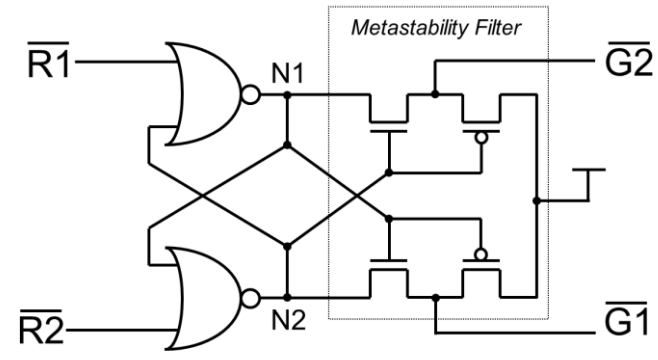
4.1 A Typical MUTEX

A MUTEX is normally used as an arbiter circuit, as explained in Chapter 2, which grants a request to a client to access a particular resource. A typical MUTEX circuit is composed of a cross-coupled NAND gate latch driven by two input signals R1

and R2 followed by a mutual-exclusive metastability filter delivering two output signals G1 and G2, as shown on the left of Figure 4.1. This circuit may fall into metastability if both input signals R1 and R2 arrive simultaneously; subsequently the outputs of the latch will be locked into a metastable state, turning-on all latch transistors and conducting short circuit currents. This holds the rest of system waiting until there is enough noise at the inputs. Metastability could resolve unnoticeably, but if this noise is quite small, it will probably last for a much longer time providing there are no disturbances [50]. The MUTEX may also be composed of a cross-coupled NOR gate driven by two input signals $\overline{R1}$ and $\overline{R2}$ followed by a mutual-exclusive metastability filter delivering two output signals $\overline{G1}$ and $\overline{G2}$, as shown on the right of Figure 4.1, however in this thesis only the NAND based MUTEX is considered as being typical.



(a) NAND-based MUTEX circuit



(b) NOR-based MUTEX circuit

Figure 4.1 Typical MUTEX circuits

4.1.1 Analytical Approximation Model for Metastability in a MUTEX

To analyze metastability behavior in a MUTEX, the bistable NAND gate latch needs to be analyzed and reduced down to its small-signal parameters, during metastability, in terms of the latch transconductance g_m, output conductance g_{out},

and output and Miller capacitance (C_{out} and C_M). This gives an estimate of the MUTEX metastability resolution time constant product based on Equation (2.10) introduced earlier in Chapter 2 for a cross-coupled inverter.

With reference to Figure 4.2, the analysis is undertaken with respect to the NAND gate on left, which produces an output N1 and receives feedback as input from N2, under the assumption of a symmetric MUTEX. To reduce complexity, each NAND gate analysis is split into a PMOS network and a NMOS network.

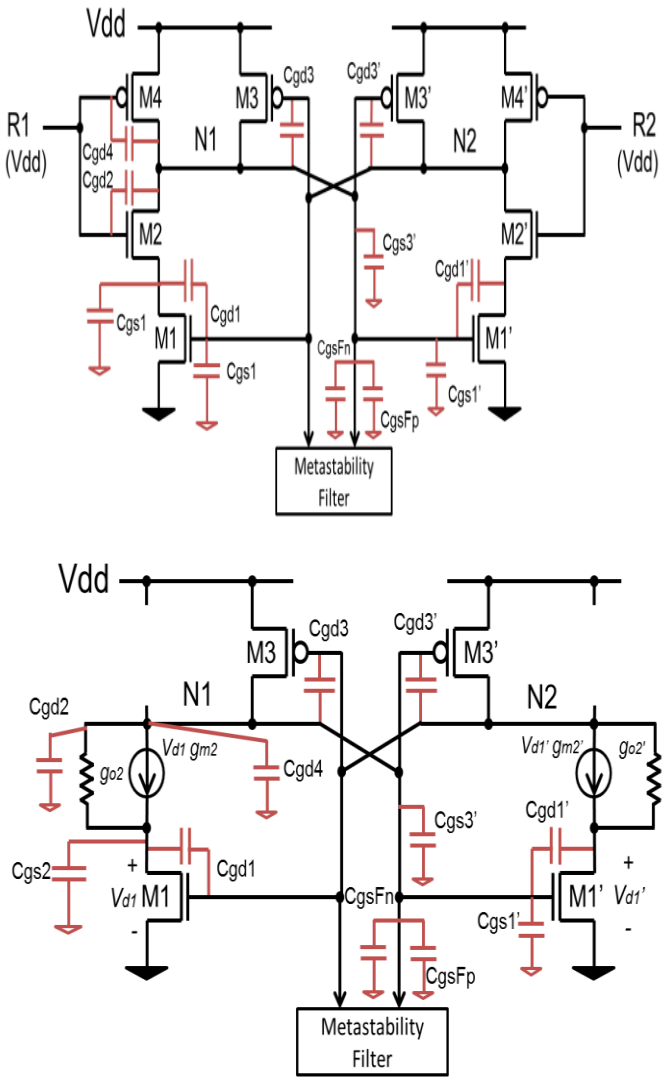


Figure 4.2 Modelling of NAND gates during metastability

At first, it is necessary to find the metastable level, which determines the region of operation of transistors M1 and M3. In a similar manner for the inverter, the metastable level voltage is derived as shown in Equation (4.1). In a symmetric MUTEX, the metastable level is the same for both nodes N1 and N2. This leads to the realization that all feedback transistors are in saturation during metastability,

hence transistor M1 in saturation as ($V_{DS1}>V_{GS1}-V_{THn1}$), and transistor M3 is also in saturation as ($V_{DS3}<V_{GS3}-V_{THp3}$).

$$V_m = \frac{\sqrt{\frac{W_{p3}\times\mu_p}{W_{n1}\times\mu_n}}(V_{DD}+V_{THp3})+V_{THn1}}{1+\sqrt{\frac{W_{p3}\times\mu_p}{W_{n1}\times\mu_n}}} \qquad (4.1).$$

First, it is necessary to obtain a formula for the MUTEX transconductance. The PMOS network has a transconductance equivalent to that of transistor M3 as in Equation (4.2) below. Whereas, the NMOS network transconductance is equivalent to that of a simple cascode common-source amplifier. This cascode configuration reduces the effect of the Miller capacitance C_{gd1} on the input signal at the input of transistor M1 and increases the overall output resistance to increase gain. However, it may reduce the overall transconductance if transistor M1 has a comparably large conductance g_{o1}, which is dependent on the drain current available governed by the transistor model and channel length modulation. However, since it is known that M1 is definitely in saturation, the value of g_o typically in saturation is almost a hundred times smaller than g_m [73], and therefore the transconductance of the NMOS network g_{mn} is approximately g_{m1}, and the overall transconductance is the sum of g_{mn} and g_{mp}, as expressed in Equation (4.2) below.

$$\left.\begin{aligned} g_{mp} = g_{m3} &= \frac{\mu_p C_{ox} W_3}{L}\left(V_{N2} - V_{DD} - V_{Tp}\right) = \sqrt{\frac{2\mu_p C_{ox} W_3 I_D}{L}} \\ g_{mn} &= \frac{(g_{o2}+g_{m2})g_{m1}}{(g_{o1}+g_{o2}+g_{m2})} \\ \because M1\ in\ sat \quad &\Rightarrow \quad g_{m1} \approx 100 \times g_{o1} \\ \therefore g_{mn} \approx g_{m1} &= \frac{\mu_n C_{ox} W_1}{L}(V_{N2} - V_{Tn}) = \sqrt{\frac{2\mu_n C_{ox} W_1 I_D}{L}} \\ g_m = g_{mn} + g_{mp} &= \sqrt{\frac{2C_{ox} I_D}{L}\left(\mu_n W_1 + \mu_p W_3\right)} \end{aligned}\right\} \qquad (4.2).$$

In a similar manner, the output conductance of both networks can be derived as in Equation (4.3).

$$
\left.
\begin{aligned}
g_{outp} &= g_{o3} = \lambda I_D \\
R_{outn} &= r_{o1} + r_{o2} + r_{o1} r_{o2} g_{m2} \approx r_{o1} r_{o2} g_{m2} \\
&\Rightarrow\ g_{outn} \approx \frac{g_{o1} \cdot g_{o2}}{g_{m2}} = \frac{L\,(\lambda I_D)^2}{\mu_n C_{ox} W_2 (V_{DD} - V_{Tn2})} \\
&\Rightarrow\ g_{outn} \approx \lambda^2 \mu_n C_{ox} \frac{W_2}{L} (V_{DD} - V_{Tn2})^3 \\
g_{out} &= g_{outn} + g_{outp}
\end{aligned}
\right\} \qquad (4.3).
$$

The output capacitances and Miller capacitance in Equation (4.4) are obtained assuming that the dominant capacitances around each transistor are those associated with its gate terminal.

$$
\left.
\begin{aligned}
C_{outp} &= C_{gd3} + C_{gd4} = (L_D W_4 + L_D W_3) C_{ox} \\
C_{outn} &= C_{gd2} = L_D W_2 C_{ox} \\
C_L &= C_{gs1'} + C_{gd1'} + C_{gs3'} + C_{gd3'} + C_{gsFp} + C_{gsFn} \\
C_{out} &= C_{outp} + C_{outn} + C_L \\
C_M &= C_{gd3} = L_D W_3 C_{ox}
\end{aligned}
\right\} \qquad (4.4).
$$

From these derived parameters, the metastability resolution time constant τ of the MUTEX is determined in Equation (4.5).

$$
\left.
\begin{aligned}
\tau &= \frac{C_{out} + 4C_M}{g_m - g_{out}} \\
\Rightarrow \tau &= \frac{C_{gd2} + C_{gd4} + 5C_{gd3} + C_{gs1'} + C_{gd1'} + C_{gs3'} + C_{gd3'} + C_{gsFp} + C_{gsFn}}{g_{m1} + g_{m3} - \frac{g_{o1} \cdot g_{o2}}{g_{m2}} - g_{o3}} \\
&\because \textit{M1 and M3 in sat.} \qquad \Rightarrow g_m \approx 100 \times g_o \\
\therefore \tau &\approx \frac{C_{gd2} + C_{gd4} + 5C_{gd3} + C_{gs1'} + C_{gd1'} + C_{gs3'} + C_{gd3'} + C_{gsFp} + C_{gsFn}}{g_{m1} + g_{m3}} \\
\tau &\approx \frac{C_{ox}\left[L_D(W_2 + 5W_3 + W_4 + 2W_{1'} + 2W_{3'} + W_{Fn} + W_{Fp}) + \left(\frac{2L}{3}\right)(W_{1'} + W_{3'} + W_{Fn} + W_{Fp})\right]}{\sqrt{\frac{2C_{ox} I_D}{L}(\mu_n W_1 + \mu_p W_3)}}
\end{aligned}
\right\} \qquad (4.5).
$$

If it is assumed that the NMOS transistors of the latch gates have a width equivalent to twice that of PMOS devices (W), and the filter PMOS transistors have

a width equivalent to twice that of NMOS devices (W_F), then the τ in Equation (4.5) and V_m in Equation (4.1) can be rewritten as in Equation (4.6).

$$\left.\begin{aligned} \tau &\approx \frac{C_{ox}[L_D(14W+3W_F)+2L(W+W_F)]}{\sqrt{\frac{2C_{ox}I_D}{L}(2\mu_n+\mu_p)W}} \\ V_m &= \frac{\sqrt{2\mu_p/\mu_n}(V_{DD}+V_{THp3})+V_{THn1}}{1+\sqrt{2\mu_p/\mu_n}} \end{aligned}\right\} \quad (4.6).$$

Equation (4.6) shows that to reduce τ by resizing the MUTEX depends on reducing the MUTEX size as a whole together with increasing the latch size to the load size ratio. The latter must be balanced so that the metastable level, in Equation (4.1), remains within a region that would maintain both transistors M1 and M3 in saturation. Increasing the size of the latch increases g_m, however it also increases the capacitances, which limits the minimization of τ, because g_m depends on the amount of current I_D and the size of transistors M1 and M3, which are a major contributor to C_L and C_M. Using transistors with a low-V_{TH} increases the current that increases g_m without adding capacitance. There are two main problems with this design choice; first, the gates have small noise margins, and second, its leakage and switching power is very large compared to designs with normal-V_{TH} devices.

The sensitivity of the value of τ towards W, W_F and V_{DD} can be derived is in Equation (4.7). The sensitivity of τ to W and W_F are similar but negative for W and positive for W_F, in other words as W increases τ decreases, however increasing W_F increases τ. The sensitivity of τ to V_{DD} has a negative sign also, which will increase with reduced voltage supply. It depends mainly on its value and the difference between threshold voltages of PMOS and NMOS transistors.

$$\left.\begin{aligned} S_W^\tau &= \frac{W}{\tau}\frac{\partial\tau}{\partial W} = \frac{[14WL_D+2WL]}{[L_D(14W+3W_F)+2L(W+W_F)]} - 1 \\ \Rightarrow\ S_W^\tau &= \frac{-W_F[3L_D+2L]}{[L_D(14W+3W_F)+2L(W+W_F)]} \\ S_{W_F}^\tau &= \frac{W_F}{\tau}\frac{\partial\tau}{\partial W_F} = \frac{W_F[3L_D+2L]}{[L_D(14W+3W_F)+2L(W+W_F)]} \\ S_{V_{DD}}^\tau &= \frac{V_{DD}}{\tau}\frac{\partial\tau}{\partial V_{DD}} = \frac{-V_{DD}}{V_{DD}+V_{THp}-V_{THn}} \end{aligned}\right\} \quad (4.7).$$

4.1.2 Simulating a MUTEX

To estimate the performance of a MUTEX using a circuit simulator, first, a schematic test setup is constructed, as shown in Figure 4.3. The Device-Under-Test (DUT) receives two input request signals R1 and R2 buffered through 4X large gates, and produces two output grant signals G1 and G2 driving a load of 4X large gates. The internal node signals N1 and N2 are not attached to any loading, but are only used for observation. The test starts by sending signals R1 and R2 whose rising edges arrival times are separated by an input time difference Δt_{in}. Then, the output delay time t_{out} is measured against Δt_{in}, which is the propagation delay between a request signal to the rising edge of its corresponding grant signal. After taking measurements, the test is repeated with a narrower or wider input time, until there are sufficient results to plot a curve showing some constant t_{out} and an exponentially rising slope from right to left, similar to Figure 4.4. Then, the MUTEX timing characteristics can be extracted. The delay time t_D is the minimum output time, the metastability time constant is extracted by taking the absolute semi-natural-log slope near the metastable region where t_{out} measurements draw an exponential increase from t_D, which defines the metastability window T_w.

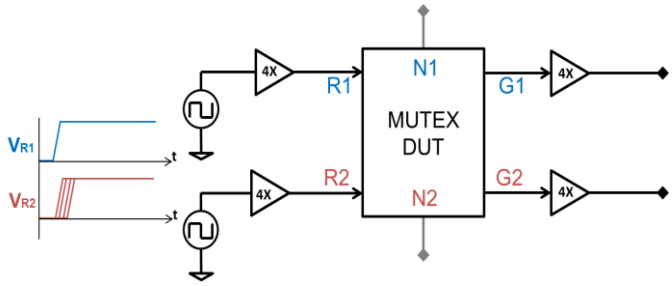


Figure 4.3 Testing setup for simulating MUTEX performance

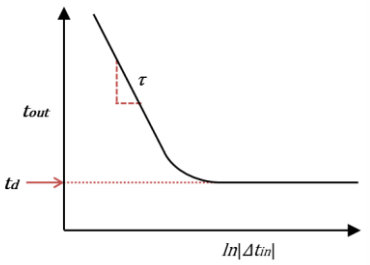


Figure 4.4 Input-Output time curve

The impact of varying the supply voltage and temperature was determined by running a number of parametric simulations to find the value of τ and t_d. The impact of varying the process parameters was determined by running a number of Monte Carlo Statistical simulations to find the value of τ and t_d. The process parameters of the input and output buffers were not varied; only those for the DUT. To reduce the number of Monte Carlo simulation samples, the request signals R1 and R2 were initiated as two periodic signals: one with period of T_{R1} and the

other with period T_{R2} that equals T_{R1} plus an offset δt. They are initially separated by a small input difference Δt_{in1}, the next cycle it becomes wider Δt_{in2} and then wider Δt_{in3}, as shown in Figure 4.5. This method computes the value of τ in a single run.

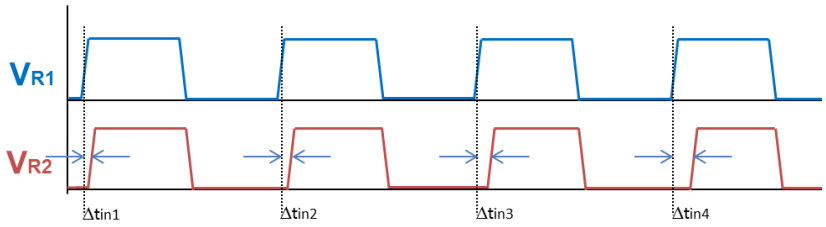
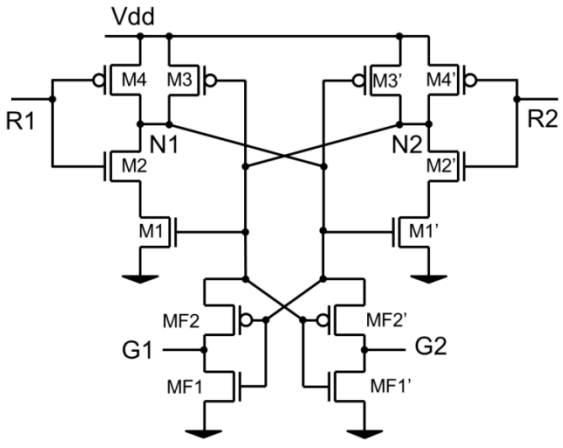


Figure 4.5 Input signals R1 and R2 to measure τ in one run

4.1.3 Simulation Results of the Typical MUTEX

The typical MUTEX (ME0) in Figure 4.6, was realized in UMC90nm CMOS process technology. In this instance, the size of the NAND gates, based on the minimum width (W) of the PMOS transistors, was changed between 0.2µm and 1µm and from 1V down to 0.5V supply at W_F = 0.24µm. According to the simulation results shown in Figure 4.7, increasing the size of the NAND gates reduces the metastability time constant τ as well as the delay time; furthermore, they both decrease with larger size of NAND gates at lower voltages. At a 1V supply, the sensitivity of the value τ to the MUTEX size is –7.91ps/µm and t_d is –28.0ps/µm, whereas at 0.5V, the sensitivity of τ to size becomes higher at –63.2ps/µm and the same for t_d at –93.5ps/µm.



Transistor	Size
M1, M2	$2W$
M3, M4	W
MF1	W_F
MF2	$2W_F$

Figure 4.6 Typical MUTEX transistor level under test

In the second simulation, the size of the loading filters on nodes N1 and N2 based on the minimum width (W_F) of the NMOS transistors was changed between 0.2µm

and 1µm and from 1V down to 0.5V supply at W=0.8µm. The results illustrated in Figure 4.8 show that by adding more load the value of τ increases while t_d reduces. At 1V, τ shows increments of 6.24ps/µm and t_d reduces to –11.7ps/µm, whereas at 0.5V, τ shows increments of 105ps/µm and t_d remains almost unchanged at –0.8ps/µm.

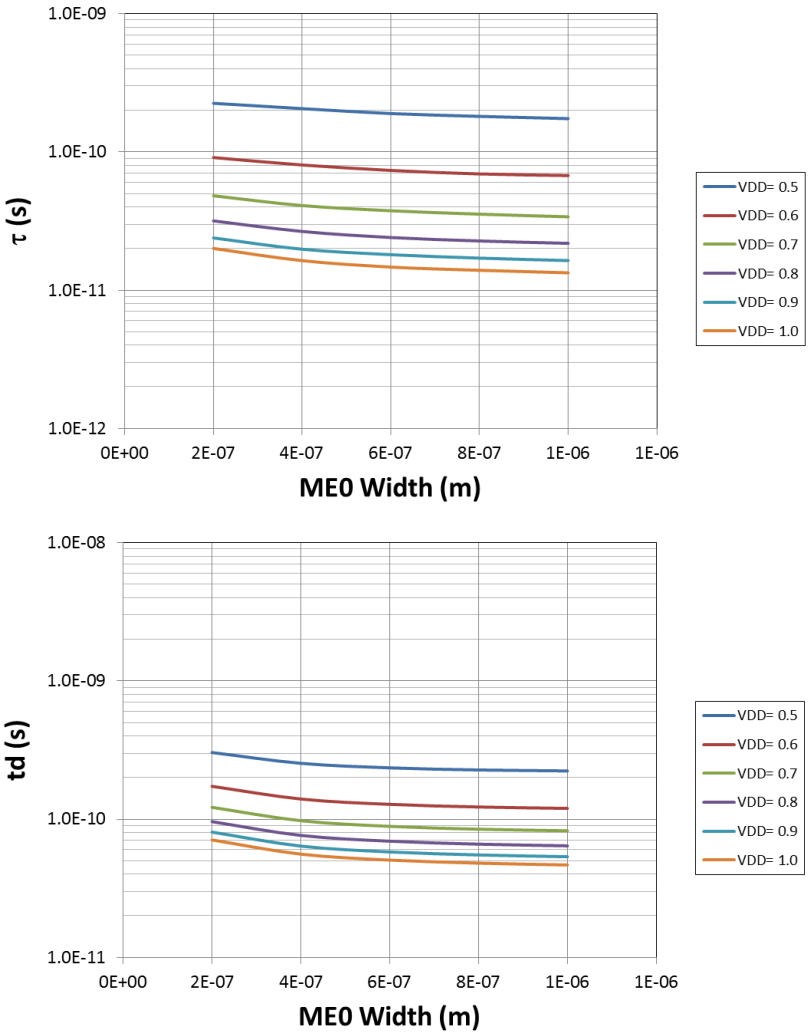


Figure 4.7 Impact of Latch size (W) on τ and t_d (ME0)

From the above results, minimization of the load and increasing the size of the NAND gates is essential to enhance the nominal performance of a MUTEX. The choice of sizes for the MUTEX depends on the points where sensitivity to a change of width is minimal. Based on Figure 4.7, this is for W sizes between 0.6µm and 1µm. On the other hand, Figure 4.8 shows a persistent increase in τ with increased size of the load W_F, which suggests keeping the load size to a minimum. From this point onwards, the presented results reflect sizes of W=0.8µm and W_F = 0.24µm. Accordingly, a MUTEX design can be optimized with large NAND NMOS transistors

to enhance the speed, and with small latch PMOS transistors and small transistors in the metastability filter to improve the metastability response.

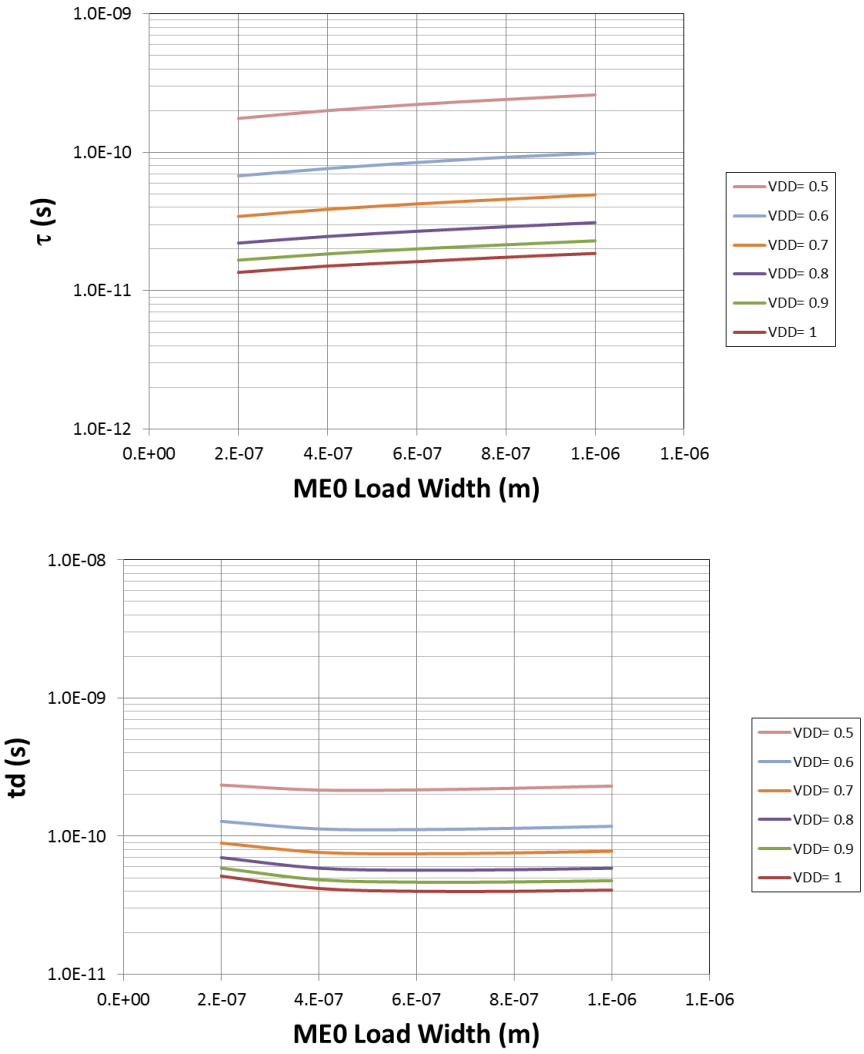


Figure 4.8 Impact of load size (W_F) on τ and t_d (ME0)

The next set of results shown in Figure 4.9 clearly indicates the impact of reducing V_{DD} from 1V down to 0.5V and temperatures between –125°C and +125°C on τ and t_d of the MUTEX. Reducing the voltage supply produces a higher change in both values of τ and t_d, and increases at low temperatures. For instance, at room temperature, τ changes by –287ps/V and t_d of –319ps/V, and at 75°C τ changes by –186ps/V and t_d by –272ps/V, whereas at –50°C, τ changes by –1.13ns/V and t_d by –549ps/V. With changes of temperature, τ decreases at V_{DD} below 0.9V, while it increases at higher voltages. For instance, at 1V, the change in τ is very subtle around 26.5fs/°C, whereas at 0.5V it sharply decreases at –2.91ps/°C. This is because at high V_{GS} drain currents are lowered with increased temperatures due to mobility reduction, while at low V_{GS} drain currents are lowered with increased

temperatures due to threshold-voltage reduction. This is the same for t_d responding differently towards temperature and voltage supply changes.

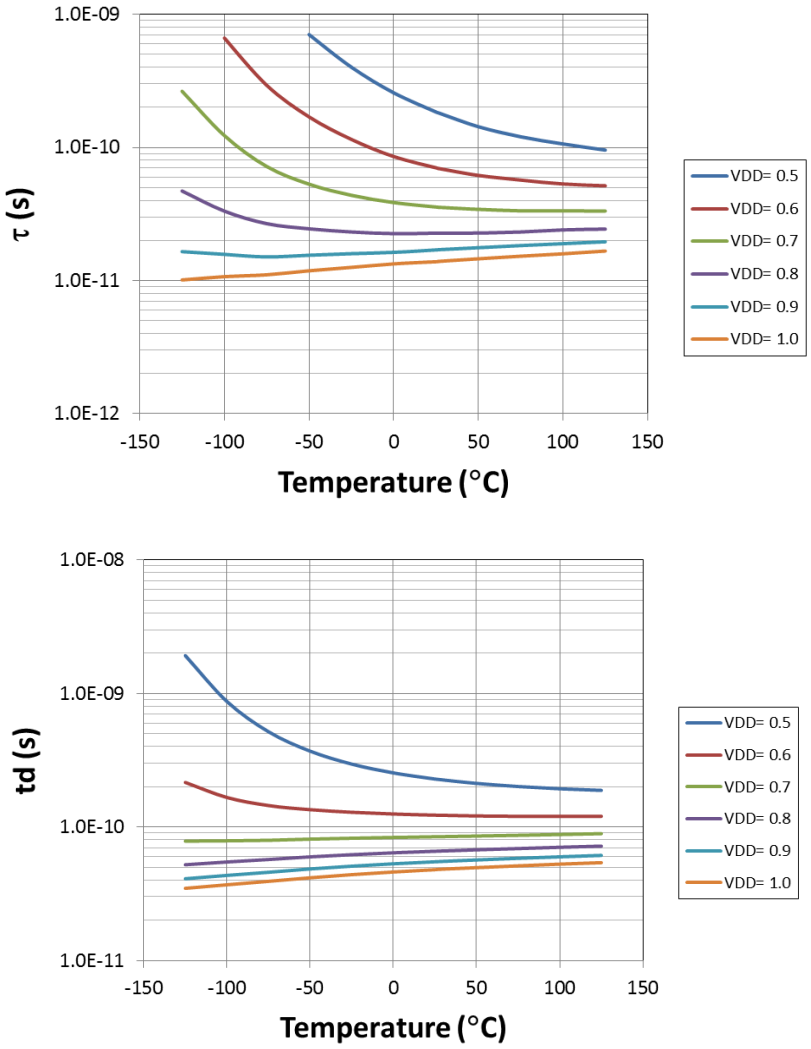


Figure 4.9 Impact Voltage and Temperature on τ and t_d (ME0)

Statistical distribution results of τ and t_d in response to a ±3σ 1000 runs process variation and voltage supply reduction from 1V down to 0.5V are shown in Figure 4.10, and their mean and mean plus 3 standard-deviations against the voltage supply are plotted in Figure 4.11. Process variations at 1V could influence a target τ of 14ps to increase to 15.9ps, while at 0.5V that deviation could push the value of τ from 188ps to reach over 336ps. In the contrary, the delay time variation seems to be subtle; it may increase from 48ps at 1V to 54ps and from 229ps at 0.5V to 293ps. This is because none of transistors composing the MUTEX used the minimum channel-width at 120nm. The increase of variability of τ becomes very critical if the time required to resolve metastability is time bounded in the MUTEX, which may increase its MTBF.

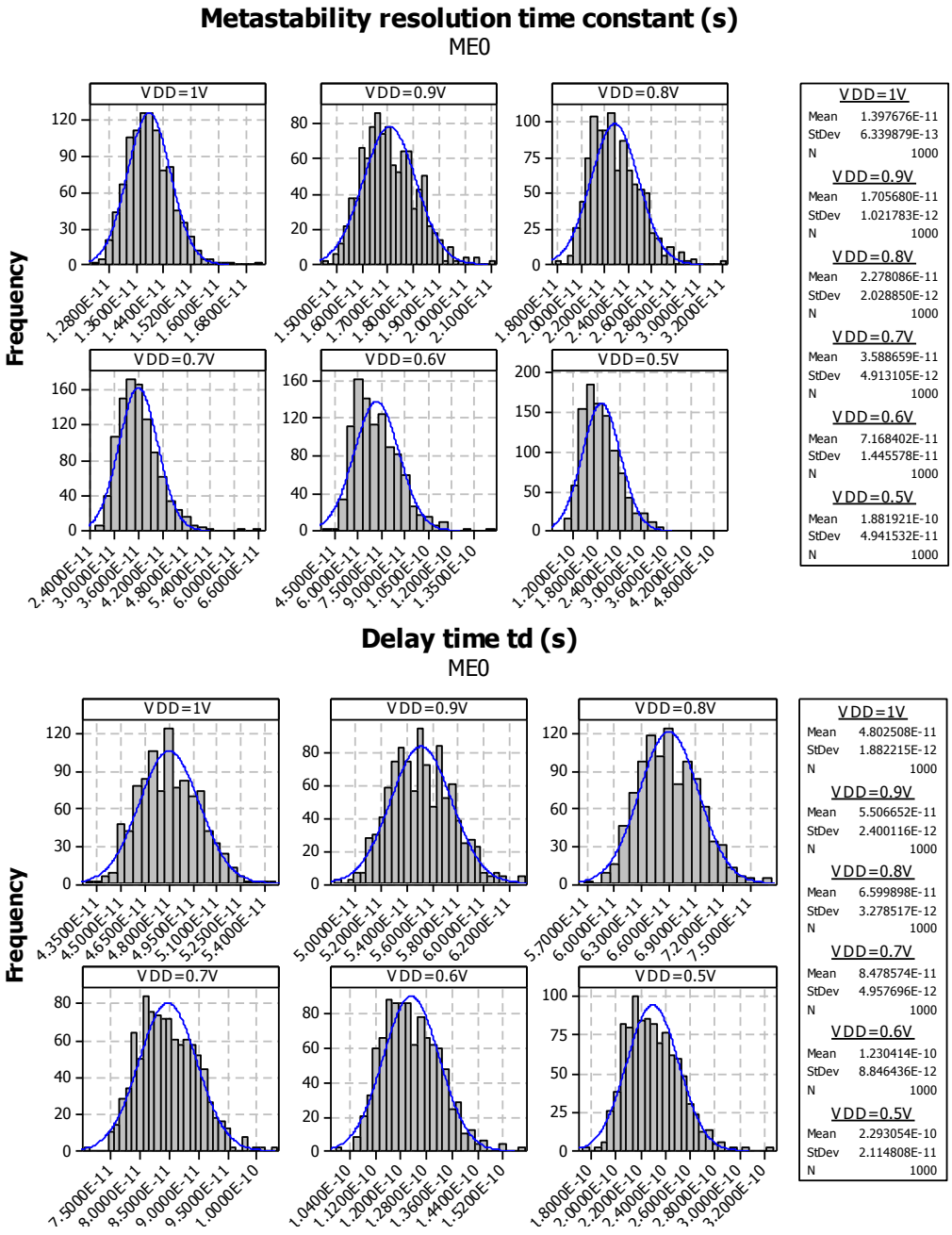


Figure 4.10 Histograms of the impact Process Variation and Voltage on τ and t_d (ME0)

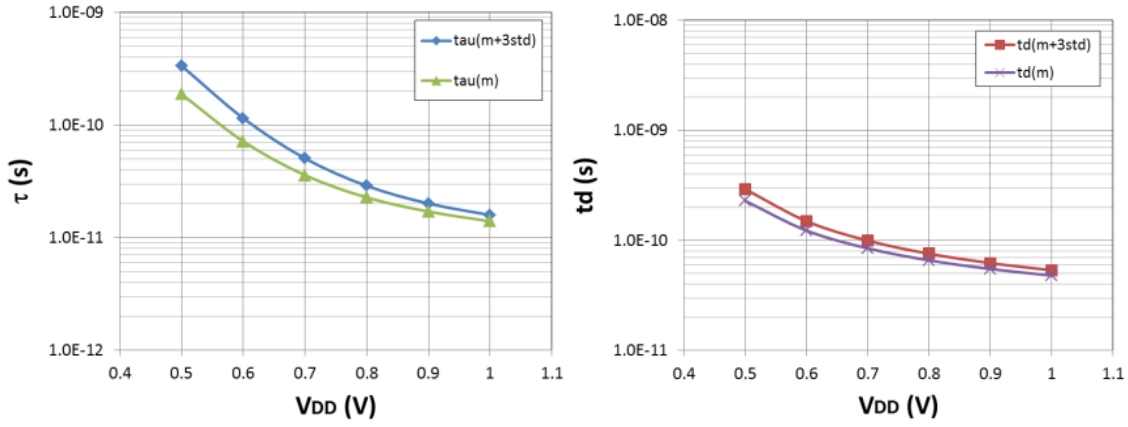


Figure 4.11 Impact of Process Variation and Voltage reduction on τ and t_d (ME0)

4.2 Improving the MUTEX

This work proposes a number of modifications around a typical MUTEX in order to improve its decision time particularly during metastability events as well as its tolerance towards PVT variations and variable environments. These approaches are based on Equation (4.5) by either increasing the denominator or by reducing the numerator using circuit techniques other than resizing. These approaches focus on adding fixed current sources to provide more current flowing down the latch during metastability, reducing the capacitive loading, boosting the transconductance of the latch during metastability, compensating for the existing Miller capacitance, and by adding an imbalanced transconductance.

In this section, nine different Modified MUTEXes (M-MUTEXes) are proposed. They utilize circuit techniques to improve τ tolerance towards PVT variations.

4.2.1 M-MUTEX with current source (ME1A)

The first approach focuses on increasing the transconductance by adding current sources to inject more current from the voltage rail into nodes N1 and N2 and down through the NMOS transistors. During metastability, a current source I_{SS} shifts the metastable voltage level up and enhances the transconductance of the NMOS branch by the amount of $\sqrt{g_{m1} I_{SS}/I_{D1}}$, but it adds to the capacitive loading, as expressed in Equation (4.8) and Equation (4.9). The current source can be implemented by adding two PMOS transistors of width W to the typical MUTEX, with zero volts applied to their gate terminals to give excess current in each NAND gate; $I_{SS} = \frac{1}{2}\mu_p C_{ox} W/L \left(-V_{DD} - V_{THp}\right)^2$. This technique is similar in manner to that used in the Robust synchronizer [82].

$$V_{m(new)} = V_{m(ME0)} + \sqrt{2LI_{SS}/\mu_n C_{ox} W_{n1}} + V_{THn1} \tag{4.8}.$$

$$\left.\begin{aligned} \tau &\approx \frac{C_{gd2}+C_{gd4}+C_{gd5}+5C_{gd3}+C_{gs1'}+C_{gd1'}+C_{gs3'}+C_{gd3'}+C_{gsFp}+C_{gsFn}}{g_{m1}+g_{m3}} \\ \tau &\approx \frac{C_{ox}[L_D(15W+3W_F)+2L(W+W_F)]}{\sqrt{\frac{2C_{ox}}{L}(\mu_n 2(I_{D3}+I_{SS})+\mu_p I_{D3})W}} \end{aligned}\right\} \tag{4.9}.$$

This approach is applied in the first M-MUTEX, ME1A, which is a MUTEX with two extra PMOS transistors M5 and M5' between V_{DD} and nodes N1 and N2, as shown in Figure 4.12. These extra transistors operate as current sources during normal

operation and during metastability. Their gate terminals are controlled by the corresponding MUTEX grant output, in other words for the PMOS connected to N1, its gate is controlled by G1 signal, as shown in Figure 4.12. This way, if both request signals rise at exactly the same time, the internal nodes will go into a metastable state. During this time, the grant signals, being kept low by the metastability filter, will keep the additional PMOS transistors in saturation, injecting constant currents through the NMOS branch, which increase their transconductance g_{mn} and their gain to recover faster from metastability. In the case of low V_{DD}, ME1A has greater g_{mn} compared to the typical MUTEX, and that is why it is expected to overcome metastability quicker.

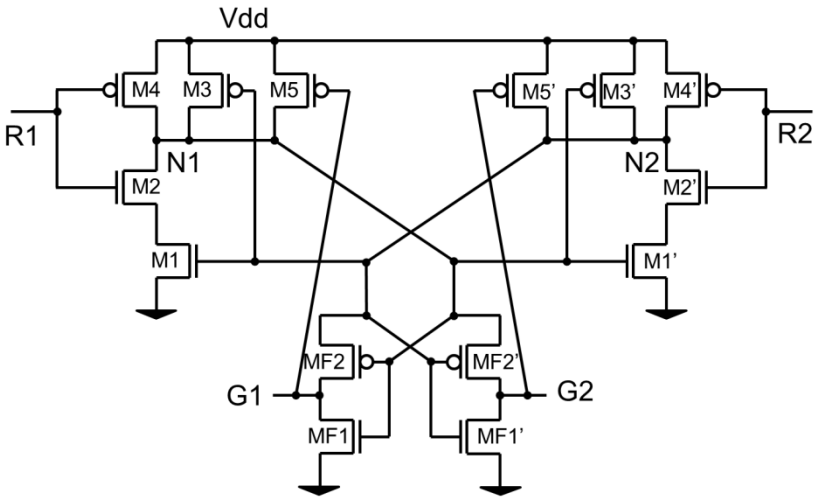


Figure 4.12 M-MUTEX ME1A with current source biased by output feedback (ME1A)

4.2.2 M-MUTEX with current source and reduced load (ME1B)

The second approach focuses, on top of increasing g_m, reducing the capacitive loading and removing Miller capacitance. The capacitive loading can be reduced by using feedback only to NMOS transistors M1 and M1' and not to PMOS transistors M3and M3', while using transistors M3 and M3' as current sources. This way the load is reduced and the Miller effect is cancelled. Although the transconductance is reduced by g_{mp}, the value of g_{mn} is increased due to the current source. This is shown in Equation (4.10).

$$\tau \approx \frac{C_{gd2}+C_{gd4}+C_{gd3}+C_{gs1'}+C_{gd1'}+C_{gsFp}+C_{gsFn}}{g_{m1}}$$
$$\Rightarrow \quad \tau \approx \frac{C_{ox}[L_D(6W+3W_F)+2L(2W/3+W_F)]}{\sqrt{2\mu_n C_{ox} 2W I_D/L}} \qquad (4.10).$$

This approach is implemented the same way as ME1A, in addition, to removing the

two PMOS transistors of the NAND gates, whose gate-terminal was connected to nodes N1 and N2. The additional PMOS ones are left biased by the grant signals G1 and G2 and operating as current sources. This circuit is called ME1B and is shown in Figure 4.13.

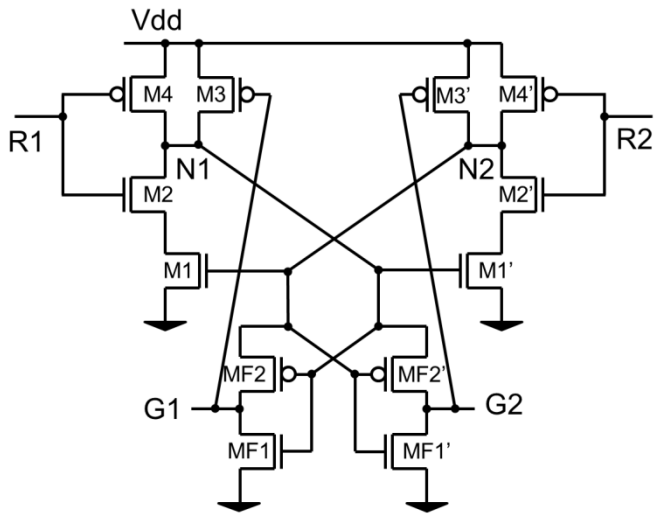


Figure 4.13 M-MUTEX ME1B with current source and reduced load (ME1B)

The internal feedback from N1 and N2 only controls the NMOS branch, which blocks the other request from being granted and the internal nodes are pulled high to V_{DD} by the current sources during normal operations as well as the metastable state. This technique helps to reduce the total capacitive loading on the internal nodes and remove the Miller capacitance associated with M3 in ME0 and ME1A. Although, removing the PMOS transistors removes g_{m3} from the Equation (4.5), reducing the overall transconductance, the transconductance of the NMOS transistor g_{m1} is higher because, due to the effect of the inserted current source, the metastable level V_m and the drain current are higher, compared to that of ME0, which increases the transconductance in Equation (4.2).

4.2.3 M-MUTEX with g_m Boosting

The next approach is similar to the first approach implemented in ME1A, except the additional PMOS devices are not biased all the time. Otherwise, these transistors receive a feedback signal from a metastability error checking circuit that checks for possible metastability based on the logic state of both input requests and both output grants and then will turn on or off the additional two PMOS transistors as current sources. This circuit is shown in Figure 4.14, and is called ME2A. Specifically, in Figure 4.14, if both input requests are ones and both

output grants are zeroes, that is indicating possible metastability, then both current sources M5 and M5' are enabled together, by the output ME of the 4 input NAND gate, to inject excess currents down the NAND gates and enhance their transconductance. After that, once one of the outputs changes to a logic '1', the error circuit recognizes that and disables both current sources.

Another approach combines two past approaches ME1B and ME2A as M-MUTEX ME2B. This is shown in Figure 4.15.

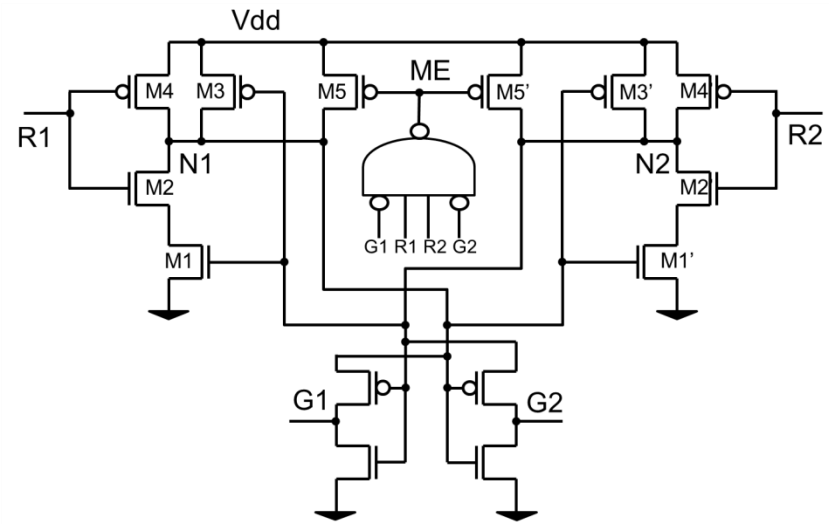


Figure 4.14 M-MUTEX with g_m boosting during metastability (ME2A)

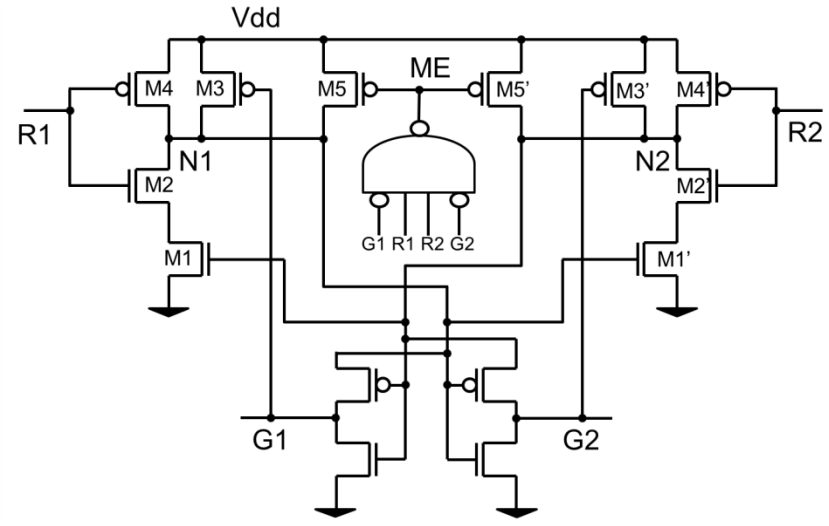


Figure 4.15 M-MUTEX with g_m boosting, current-source and reduced-load (ME2B)

4.2.4 M-MUTEX with Cascode Feedback

The next approach focuses on only compensating for the Miller capacitance effect C_{gd3} in the typical MUTEX by cascoding PMOS feedback transistor, by adding a PMOS transistor M5 between M3 and the output node that is biased by the

associated grant signal, as shown in Figure 4.16. In this way, the value of τ is reduced by $4C_{gd3}$ as expressed in Equation (4.11). This implementation is called ME3A.

$$\tau \approx \frac{C_{ox}[L_D(10W+3W_F)+2L(W+W_F)]}{g_{m1}+g_{m3}} \quad (4.11).$$

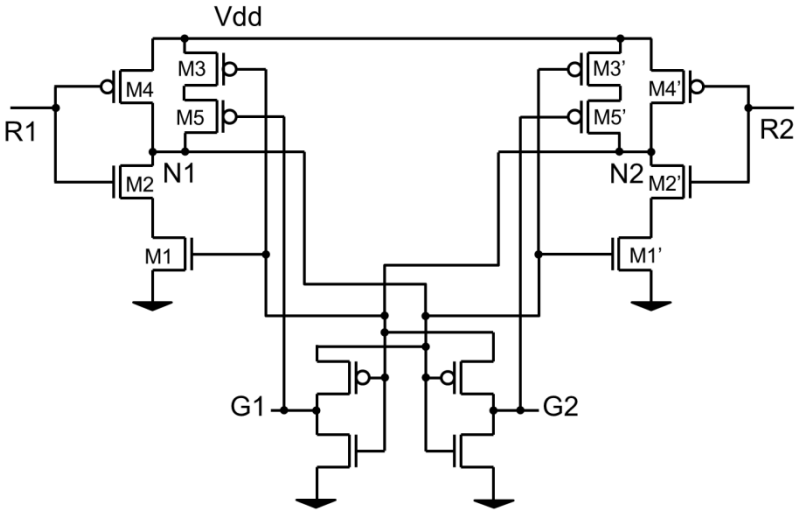


Figure 4.16 M-MUTEX with Cascode Feedback with feedback current source (ME3A)

This approach can be improved by adding a metastability error checking circuit, similar to ME2A, to boost g_m during metastability. This circuit is shown in Figure 4.17, and is called ME3B. In this way, the value of τ benefits from reducing the Miller capacitance and boosted g_m as expressed in Equation (4.12).

$$\tau \approx \frac{C_{ox}[L_D(11W+3W_F)+2L(W+W_F)]}{\sqrt{\frac{2C_{ox}}{L}(\mu_n 2(I_{D3}+I_{SS})+\mu_p I_{D3})W}} \quad (4.12).$$

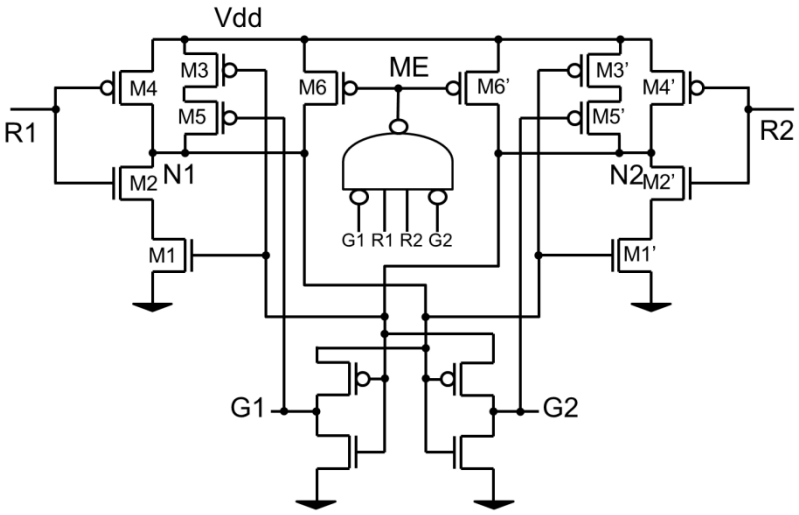


Figure 4.17 M-MUTEX with Cascode and g_m boosting (ME3B)

This approach can also be implemented using standard cell OAI circuits and is referred to as ME4A, as shown in Figure 4.18. This could be implemented as full standard-cell as shown in Figure 4.19.

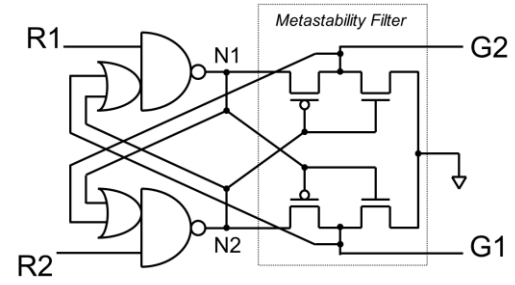


Figure 4.18 OAI based MUTEX utilizing Cascode current source feedback (ME4A)

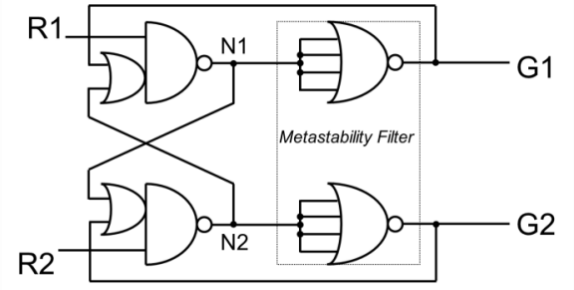


Figure 4.19 ME4A comprised of standard-cell OAI based

4.2.5 M-MUTEX with Biased Choice

This approach focuses on designing an imbalance of MUTEX to promote one request over the other. The technique is based on maneuvering the metastable point and transconductance of both internal nodes. This is achieved by adding two more PMOS devices to ME1A except that the biasing is based on a memory cell to record the last granted request. This approach is shown in Figure 4.20 and is called ME5A. One of the added PMOS transistors is turned 'ON' corresponding to the recent granted request while the other PMOS transistor is turned 'OFF'. Once the other request is granted, then the PMOS transistor will switch their states, that is, the former will turn 'OFF' and the latter will turn 'ON'. The additional two transistors are cascoded through another two PMOS transistors biased by the related output grant signal, to reduce the short circuit current if one request is granted. The ME5A circuit requires a C-element gate receiving two signals G1 and inverted G2; if G1 is granted then X1 will switch to a logic '1' and X2 to a logic '0', so

moving the metastable voltage of N1 up and that of N2 down to the unbalanced state. Once G2 is granted, the signals X1 and X2 will switch to move V_m of N1 back down and V_m of N2 up. The τ and V_m can be modelled for both nodes N1 and N2, as in equations (4.13) and (4.14), where V_{X1} is either 0V or V_{DD}, and V_{X2} is the inverse of V_{X1}.

$$\left.\begin{aligned}\tau_1 &\approx \frac{C_{ox}[L_D(15W+3W_F)+2L(W+W_F)]}{\sqrt{2C_{ox}/L\left(\mu_n 2(I_{D3}+I_{SS}(V_{DD}-V_{X2})/V_{DD})+\mu_p I_{D3}\right)W}}\\ \tau_2 &\approx \frac{C_{ox}[L_D(15W+3W_F)+2L(W+W_F)]}{\sqrt{2C_{ox}/L\left(\mu_n 2(I_{D3}+I_{SS}(V_{DD}-V_{X1})/V_{DD})+\mu_p I_{D3}\right)W}}\end{aligned}\right\} \quad (4.13).$$

$$\left.\begin{aligned}V_{m1(new)} &= V_{m(ME0)} + \left(\sqrt{2LI_{SS}/\mu_n C_{ox}W_{n1}} + V_{THn1}\right)\frac{(V_{DD}-V_{X2})}{V_{DD}}\\ V_{m2(new)} &= V_{m(ME0)} + \left(\sqrt{2LI_{SS}/\mu_n C_{ox}W_{n1}} + V_{THn1}\right)\frac{(V_{DD}-V_{X1})}{V_{DD}}\end{aligned}\right\} \quad (4.14).$$

The ME5A circuit still operates as a MUTEX based on first-come first-granted, unless the request signal arrival times are very close, then it depends on the previously granted request. For example, initially R1 was granted and followed by R1 deassertion, after that, R1 and R2 were asserted and arrived to ME5A inputs very close to each other, then ME5A will decide to grant R2, unless their arrival time leads to the maneuvered metastable point.

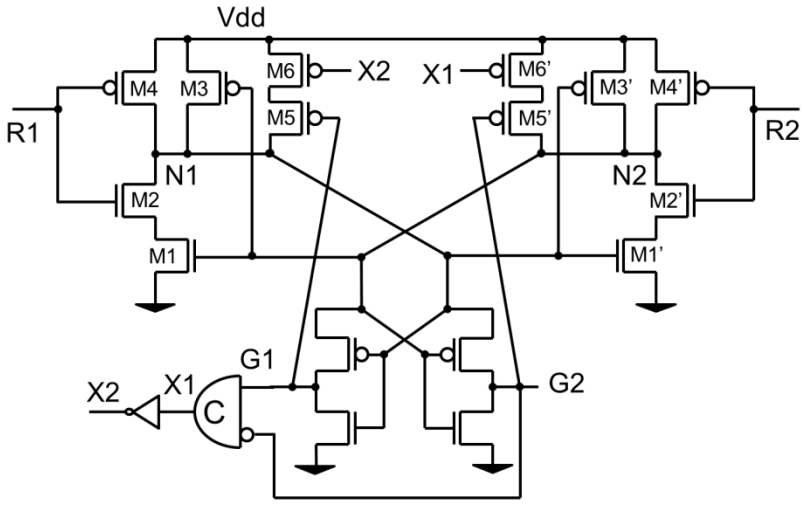


Figure 4.20 M-MUTEX ME5A

The above method can be also implemented using the metastability error checking

circuit instead of using the feedback from the grant signals. This circuit is shown in Figure 4.21 and is called ME5B. In this circuit, any occurring metastability will be detected at first, then, the imbalanced point is activated during metastability by enabling one branch over the other, based on the recently granted request.

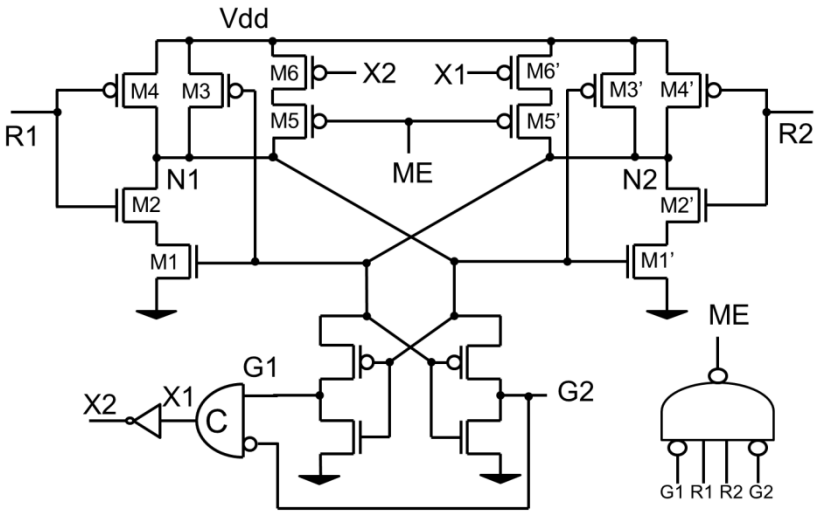


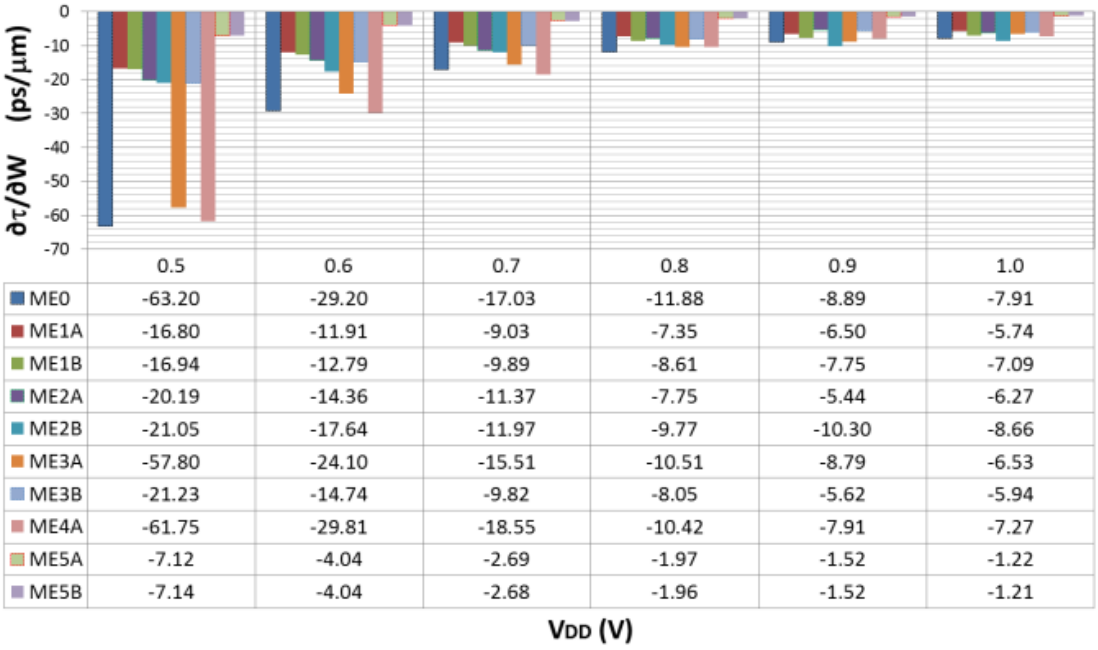
Figure 4.21 M-MUTEX ME5B

4.3 Simulation Results

4.3.1 Latch Sizing

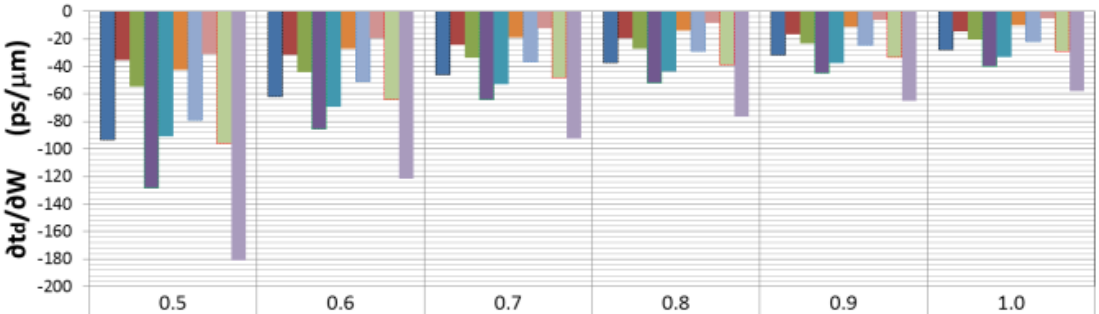
All the MUTEXes described were realized using UMC90nm CMOS process technology and tested using the test setup discussed in Section 4.1.2. First, the size of latch which is based on the minimum width (W) of the PMOS transistors was changed between 0.2µm and 1µm and from 1V down to 0.5V supply at filter size W_F = 0.24µm. The main simulation results are available in Appendix D. Figure 4.22 shows the value of τ and t_d for latch width against V_{DD}. According to these results, increasing the size of the latch reduces the metastability time constant as well as the delay time; furthermore, the amount of that reduction is increased at lower voltages. At 1V, the sensitivity of τ is best for ME5A and ME5B at about –2ps/µm and increases to –7ps/µm at 0.5V, while the worst cases were for ME0, ME3A and ME4A at around –8ps/µm at 1V and worsens to around –60ps/µm at 0.5V. Although, the cascoded ME3A and ME4A circuits have practically no improvement over ME0 in terms of τ, the sensitivity of t_d to latch size is among the best with

around –5ps/μm at 1V, and around –40ps/μm at 0.5V. Figure 4.23 shows the improvement of the sensitivity of τ and t_d to Latch width against V_{DD}. It appears to be that M-MUTEXes ME1A, ME1B and M3B are less sensitive to latch size change in terms of τ and t_d in comparison to ME0.



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	-63.20	-29.20	-17.03	-11.88	-8.89	-7.91
ME1A	-16.80	-11.91	-9.03	-7.35	-6.50	-5.74
ME1B	-16.94	-12.79	-9.89	-8.61	-7.75	-7.09
ME2A	-20.19	-14.36	-11.37	-7.75	-5.44	-6.27
ME2B	-21.05	-17.64	-11.97	-9.77	-10.30	-8.66
ME3A	-57.80	-24.10	-15.51	-10.51	-8.79	-6.53
ME3B	-21.23	-14.74	-9.82	-8.05	-5.62	-5.94
ME4A	-61.75	-29.81	-18.55	-10.42	-7.91	-7.27
ME5A	-7.12	-4.04	-2.69	-1.97	-1.52	-1.22
ME5B	-7.14	-4.04	-2.68	-1.96	-1.52	-1.21

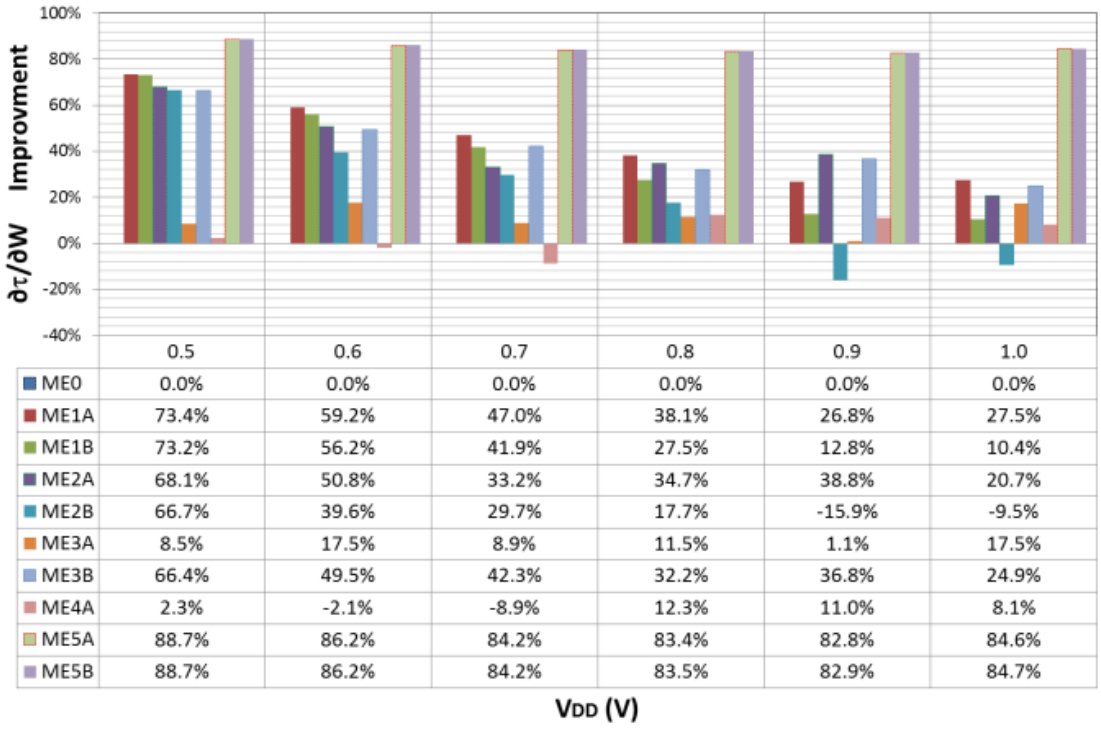
VDD (V)



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	-93.50	-62.00	-45.96	-37.02	-31.54	-28.00
ME1A	-35.10	-31.25	-24.15	-19.39	-16.49	-14.67
ME1B	-54.55	-44.30	-33.90	-27.33	-23.25	-20.69
ME2A	-128.70	-85.35	-64.00	-52.09	-44.95	-40.17
ME2B	-90.65	-69.15	-53.00	-43.35	-37.33	-33.37
ME3A	-42.50	-27.05	-18.60	-14.02	-11.37	-9.86
ME3B	-78.80	-50.80	-36.65	-29.00	-24.50	-21.73
ME4A	-31.00	-19.15	-12.20	-8.45	-6.35	-5.16
ME5A	-96.20	-64.10	-48.05	-39.10	-33.20	-28.95
ME5B	-181.15	-121.60	-92.70	-76.40	-65.23	-57.93

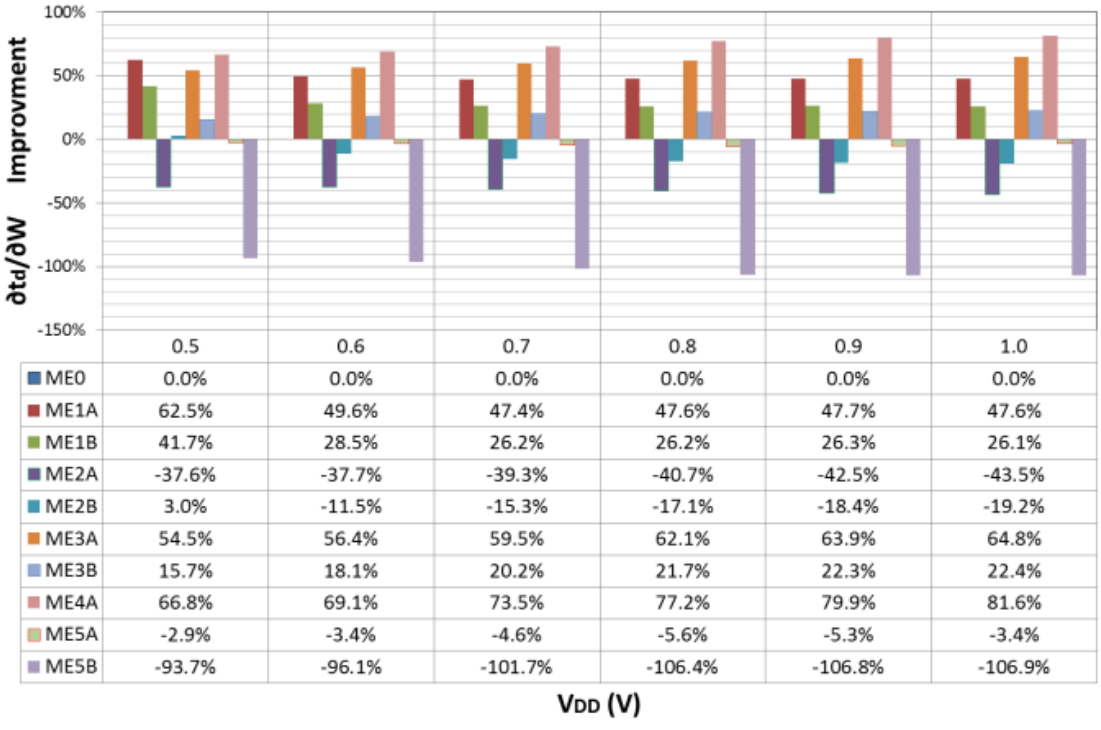
VDD (V)

Figure 4.22 Sensitivity of τ and t_d to Latch width against V_{DD}



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
ME1A	73.4%	59.2%	47.0%	38.1%	26.8%	27.5%
ME1B	73.2%	56.2%	41.9%	27.5%	12.8%	10.4%
ME2A	68.1%	50.8%	33.2%	34.7%	38.8%	20.7%
ME2B	66.7%	39.6%	29.7%	17.7%	-15.9%	-9.5%
ME3A	8.5%	17.5%	8.9%	11.5%	1.1%	17.5%
ME3B	66.4%	49.5%	42.3%	32.2%	36.8%	24.9%
ME4A	2.3%	-2.1%	-8.9%	12.3%	11.0%	8.1%
ME5A	88.7%	86.2%	84.2%	83.4%	82.8%	84.6%
ME5B	88.7%	86.2%	84.2%	83.5%	82.9%	84.7%

VDD (V)



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
ME1A	62.5%	49.6%	47.4%	47.6%	47.7%	47.6%
ME1B	41.7%	28.5%	26.2%	26.2%	26.3%	26.1%
ME2A	-37.6%	-37.7%	-39.3%	-40.7%	-42.5%	-43.5%
ME2B	3.0%	-11.5%	-15.3%	-17.1%	-18.4%	-19.2%
ME3A	54.5%	56.4%	59.5%	62.1%	63.9%	64.8%
ME3B	15.7%	18.1%	20.2%	21.7%	22.3%	22.4%
ME4A	66.8%	69.1%	73.5%	77.2%	79.9%	81.6%
ME5A	-2.9%	-3.4%	-4.6%	-5.6%	-5.3%	-3.4%
ME5B	-93.7%	-96.1%	-101.7%	-106.4%	-106.8%	-106.9%

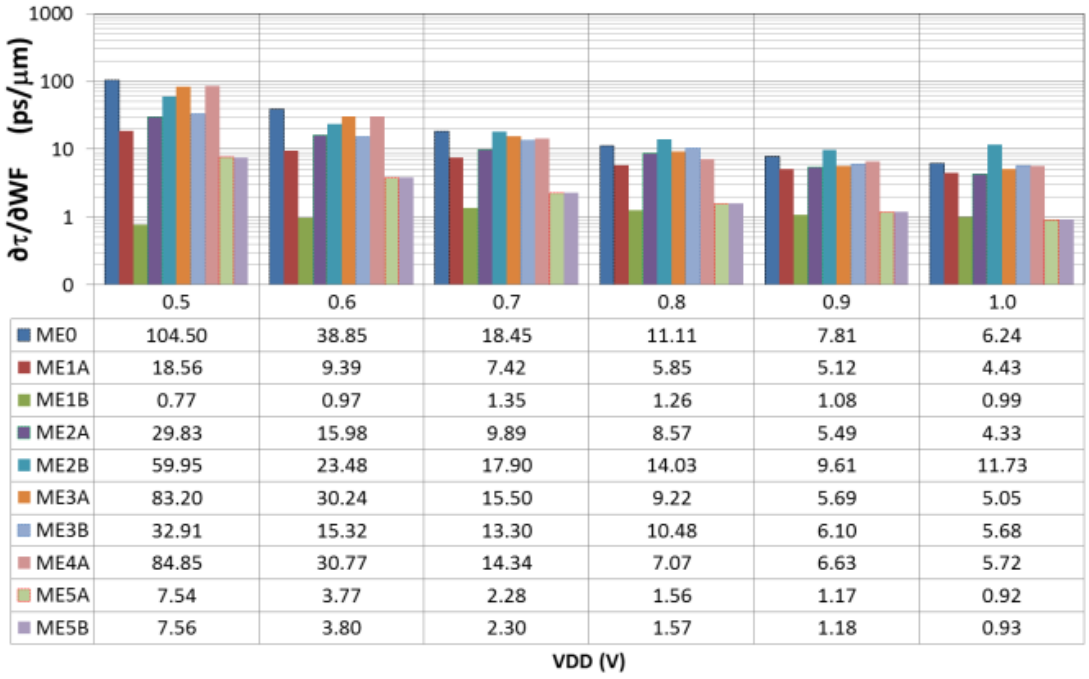
VDD (V)

Figure 4.23 Improvement of sensitivity of τ and t_d to latch width against V_{DD}

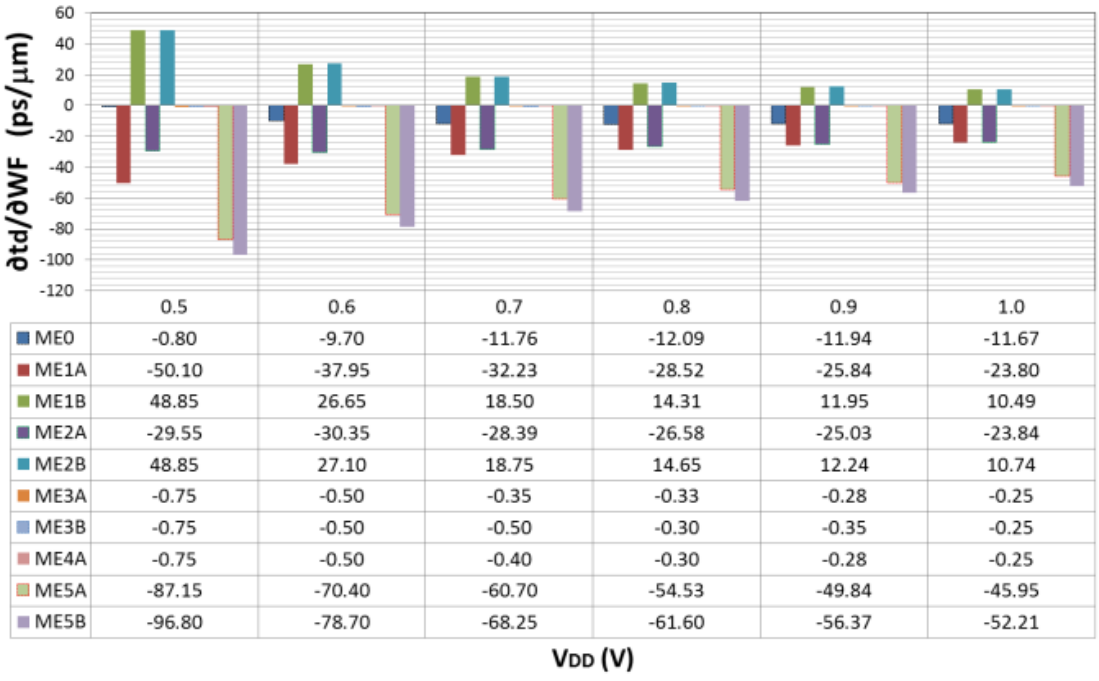
4.3.2 Load Sizing

The second set of simulation results focuses on the size of the filters that are loading the latch nodes N1 and N2, which were designed using the minimum width (W_F) of NMOS transistors and were changed between 0.2µm and 1µm and from 1V down to 0.5V supply at W=0.8µm. The results illustrated in Figure 4.24 shows the

sensitivity of τ and t_d to load width against V_{DD}, and reveal that in most designs when adding more load τ increases whereas t_d does not, a general trend for all. Figure 4.25 shows the improvement of sensitivity of τ and t_d to latch width against V_{DD}. At 1V, the ME1B, ME5A and ME5B showed the least sensitivity of τ to load size at around 1ps/µm, which is an improvement of 85% over ME0 and exceeds 90% at 0.6V. On the other hand, ME3A, ME3B and ME4A showed the least sensitivity of t_d to load size at around −1ps/µm.

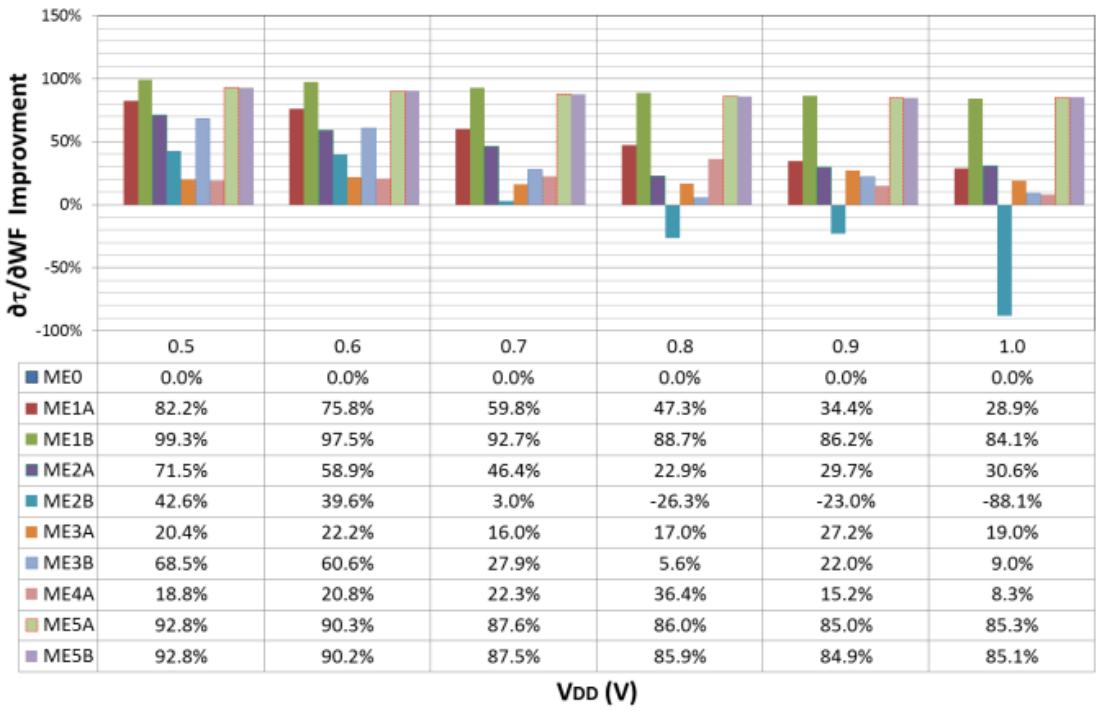


	0.5	0.6	0.7	0.8	0.9	1.0
ME0	104.50	38.85	18.45	11.11	7.81	6.24
ME1A	18.56	9.39	7.42	5.85	5.12	4.43
ME1B	0.77	0.97	1.35	1.26	1.08	0.99
ME2A	29.83	15.98	9.89	8.57	5.49	4.33
ME2B	59.95	23.48	17.90	14.03	9.61	11.73
ME3A	83.20	30.24	15.50	9.22	5.69	5.05
ME3B	32.91	15.32	13.30	10.48	6.10	5.68
ME4A	84.85	30.77	14.34	7.07	6.63	5.72
ME5A	7.54	3.77	2.28	1.56	1.17	0.92
ME5B	7.56	3.80	2.30	1.57	1.18	0.93



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	-0.80	-9.70	-11.76	-12.09	-11.94	-11.67
ME1A	-50.10	-37.95	-32.23	-28.52	-25.84	-23.80
ME1B	48.85	26.65	18.50	14.31	11.95	10.49
ME2A	-29.55	-30.35	-28.39	-26.58	-25.03	-23.84
ME2B	48.85	27.10	18.75	14.65	12.24	10.74
ME3A	-0.75	-0.50	-0.35	-0.33	-0.28	-0.25
ME3B	-0.75	-0.50	-0.50	-0.30	-0.35	-0.25
ME4A	-0.75	-0.50	-0.40	-0.30	-0.28	-0.25
ME5A	-87.15	-70.40	-60.70	-54.53	-49.84	-45.95
ME5B	-96.80	-78.70	-68.25	-61.60	-56.37	-52.21

Figure 4.24 Sensitivity of τ and t_d to load width against V_{DD}



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
ME1A	82.2%	75.8%	59.8%	47.3%	34.4%	28.9%
ME1B	99.3%	97.5%	92.7%	88.7%	86.2%	84.1%
ME2A	71.5%	58.9%	46.4%	22.9%	29.7%	30.6%
ME2B	42.6%	39.6%	3.0%	-26.3%	-23.0%	-88.1%
ME3A	20.4%	22.2%	16.0%	17.0%	27.2%	19.0%
ME3B	68.5%	60.6%	27.9%	5.6%	22.0%	9.0%
ME4A	18.8%	20.8%	22.3%	36.4%	15.2%	8.3%
ME5A	92.8%	90.3%	87.6%	86.0%	85.0%	85.3%
ME5B	92.8%	90.2%	87.5%	85.9%	84.9%	85.1%

VDD (V)

	0.5	0.6	0.7	0.8	0.9	1.0
ME0	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
ME1A	-6162.5%	-291.2%	-174.0%	-136.0%	-116.4%	-103.9%
ME1B	6206.3%	374.7%	257.3%	218.4%	200.1%	189.9%
ME2A	-3593.8%	-212.9%	-141.4%	-119.9%	-109.6%	-104.3%
ME2B	6206.3%	379.4%	259.4%	221.2%	202.5%	192.0%
ME3A	6.2%	94.8%	97.0%	97.2%	97.7%	97.9%
ME3B	6.2%	94.8%	95.7%	97.5%	97.1%	97.9%
ME4A	6.2%	94.8%	96.6%	97.5%	97.7%	97.9%
ME5A	-10793.8%	-625.8%	-416.2%	-351.2%	-317.4%	-293.7%
ME5B	-12000.0%	-711.3%	-480.4%	-409.7%	-372.1%	-347.4%

VDD (V)

Figure 4.25 Improvement of sensitivity of τ and t_d to load width against V_{DD}

4.3.3 Voltage Supply Effect

For the purpose of comparison, the results for ME0, with W=0.8µm and W_F = 0.24µm, were taken as a benchmark. The results are shown in Figure 4.26 for the impact of reducing V_{DD} from 1.2V down to 0.5V on τ and t_d, which displays an exponential increase of both values with lowered voltage supply. In terms of best performance, ME5A and ME5B can resolve metastability faster than other designs

by one to ten times compared to ME0, however, ME0 is the fastest at when there is no metastability present by a factor of two to 3.5. In terms of least sensitive to voltage supply reduction, Figure 4.27 shows the sensitivity of τ and t_d to V_{DD} and its improvement, again ME5A and ME5B can sustain more voltage supply change than other designs at sensitivity of 14ps/V, but still the delay time of ME0 varies less than the modified designs. In general, the modified designs had a penalty of increased delay time for improving τ. This is because of the overload at the inputs, outputs or internal nodes, as $delay \propto RC$. This is illustrated by comparison with inverter FO4 delay in Figure 4.28, which shows the ratio of τ and t_d of all designs to a FO4 delay of an inverter against V_{DD}. For instance, the worst delay of 3 times FO4 is that of ME5B, whereas ME0 is only 1.5 times, because the inputs R1 and R2 drive the MUTEX plus a 4 input NAND gate, outputs G1 and G2 drive load plus 2 inverters plus C-element, and the latch internal nodes N1 and N2 have an extra PMOS branch.

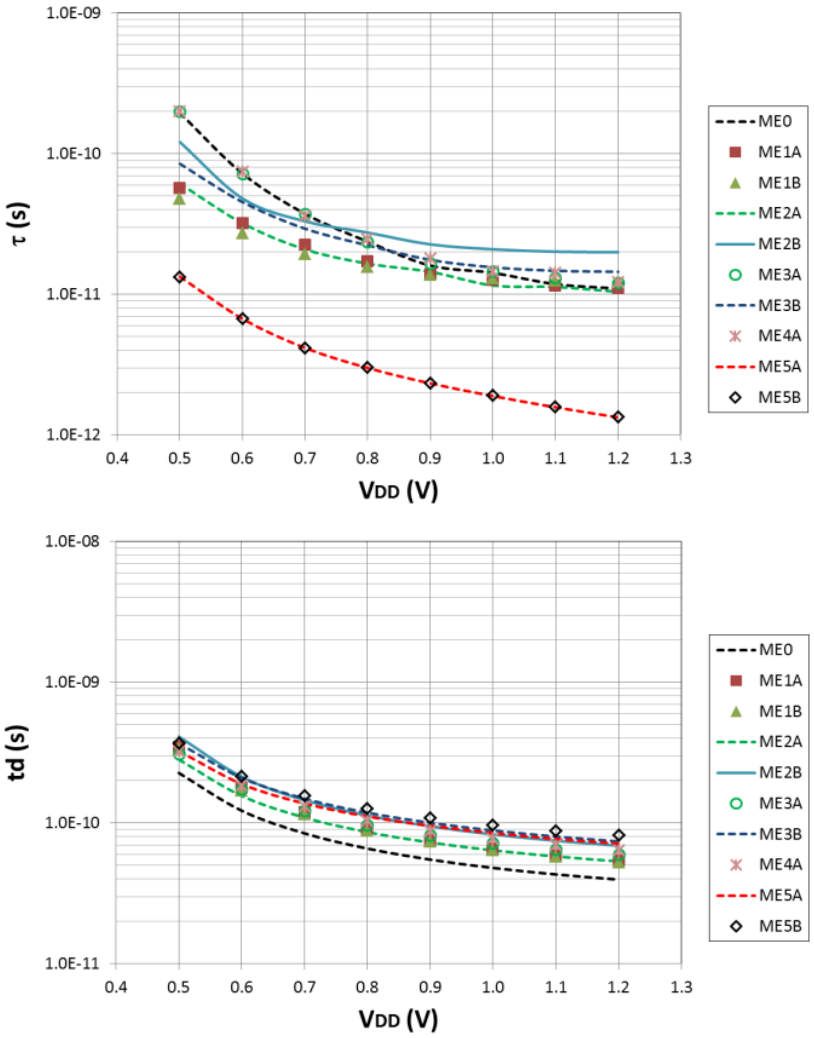
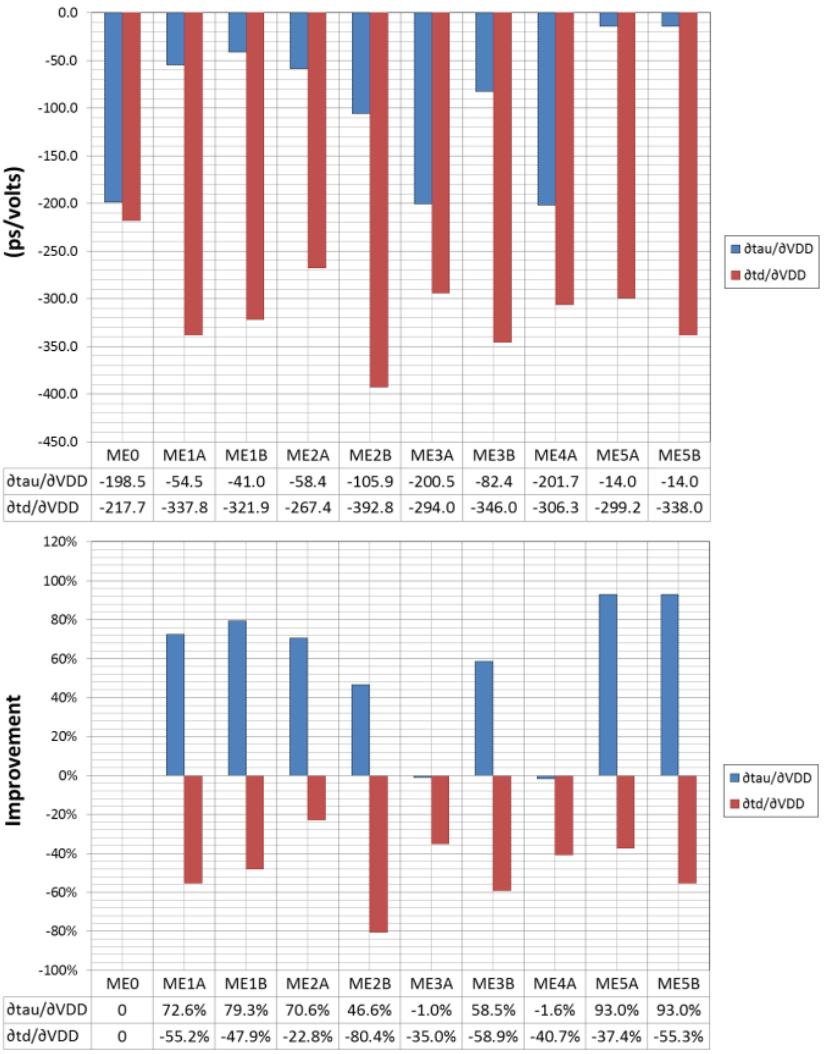


Figure 4.26 Metastability time constant τ and delay time t_d against V_{DD}



	ME0	ME1A	ME1B	ME2A	ME2B	ME3A	ME3B	ME4A	ME5A	ME5B
∂tau/∂VDD	-198.5	-54.5	-41.0	-58.4	-105.9	-200.5	-82.4	-201.7	-14.0	-14.0
∂td/∂VDD	-217.7	-337.8	-321.9	-267.4	-392.8	-294.0	-346.0	-306.3	-299.2	-338.0

	ME0	ME1A	ME1B	ME2A	ME2B	ME3A	ME3B	ME4A	ME5A	ME5B
∂tau/∂VDD	0	72.6%	79.3%	70.6%	46.6%	-1.0%	58.5%	-1.6%	93.0%	93.0%
∂td/∂VDD	0	-55.2%	-47.9%	-22.8%	-80.4%	-35.0%	-58.9%	-40.7%	-37.4%	-55.3%

Figure 4.27 Sensitivity τ and t_d to V_{DD} and Sensitivity Improvement

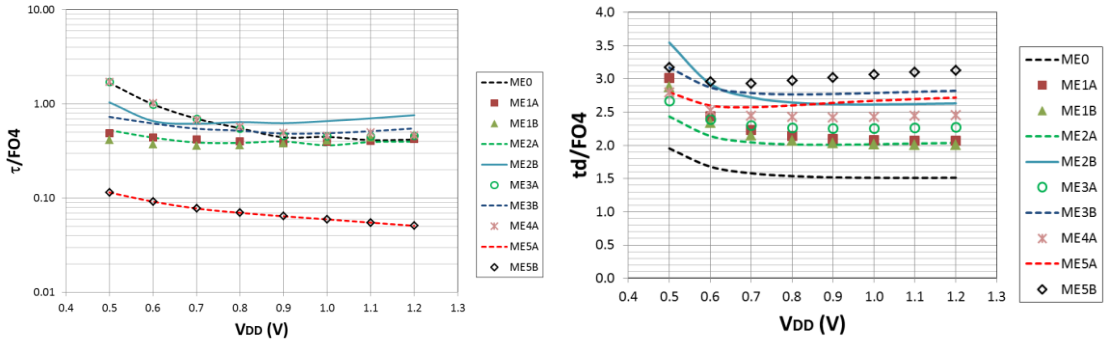
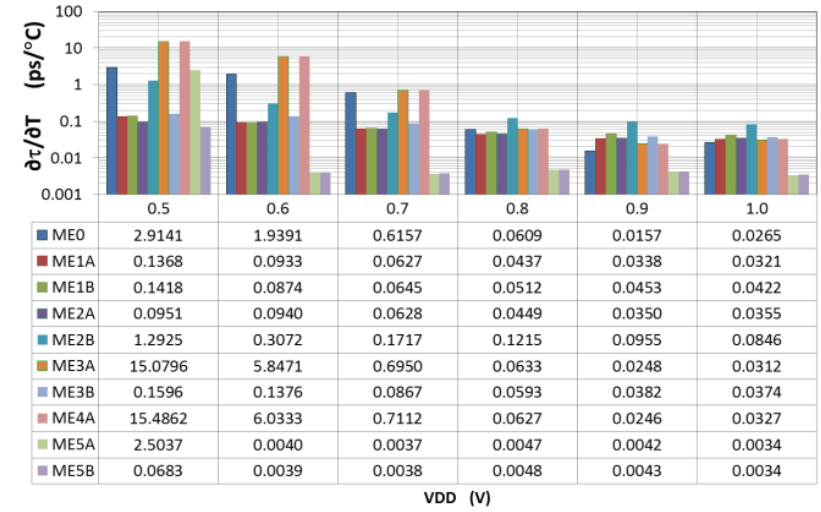


Figure 4.28 Ratio of τ and t_d of all MUTEXes to FO4 delay of inverter against V_{DD}

4.3.4 Temperature and Voltage Supply Effect

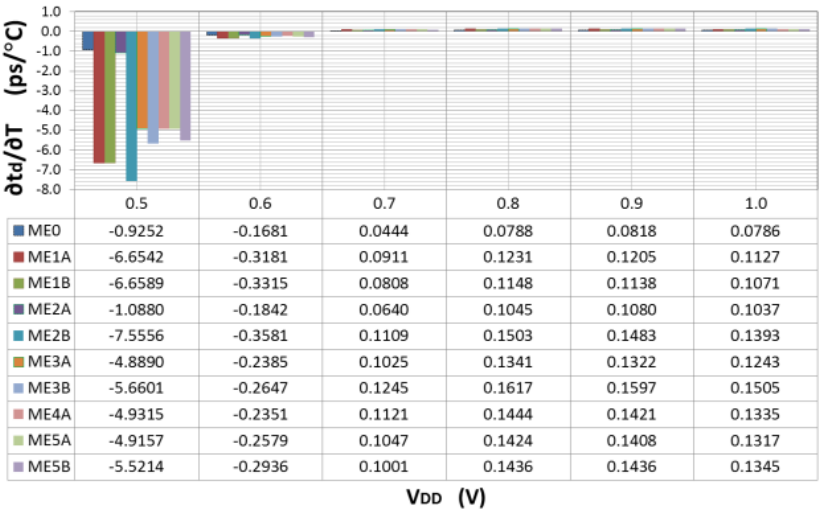
The results of the sensitivity of τ and t_d to temperature are shown in Figure 4.29 for a temperature sweep between –125°C and +125°C with changes in V_{DD} from 1V down to 0.5V. Overall, the sensitivity of τ to temperature increases with reducing the voltage supply. For example, at 1V τ is sensitive to temperature at less than

90fs/°C, which appears very small, whereas at 0.5V some circuits are tolerant temperatures changes, namely ME1A, ME2B, ME2A, ME3B and ME5B at sensitivities less than 0.16ps/°C. Other circuits that became very sensitive to temperature include ME0, ME2B, ME3A, ME4A, and ME5A. On the other hand, t_d responded differently towards temperature and voltage supply variations, and showed less tolerance compared to that of ME0. The sensitivity improvement of τ shown in Figure 4.30 indicate that only ME5A and ME5B can tolerate temperature variations over the full range of voltage supply with considerable improvement over ME0. Other designs start to show a better tolerance from 0.8V downwards, except for ME3A and ME4A. While the delay times of MUTEXes show less tolerance compared to that of ME0, but the second best design here is ME2A, which can tolerate a wide range of temperatures and lower voltage supply variations in terms of both τ and t_d.



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	2.9141	1.9391	0.6157	0.0609	0.0157	0.0265
ME1A	0.1368	0.0933	0.0627	0.0437	0.0338	0.0321
ME1B	0.1418	0.0874	0.0645	0.0512	0.0453	0.0422
ME2A	0.0951	0.0940	0.0628	0.0449	0.0350	0.0355
ME2B	1.2925	0.3072	0.1717	0.1215	0.0955	0.0846
ME3A	15.0796	5.8471	0.6950	0.0633	0.0248	0.0312
ME3B	0.1596	0.1376	0.0867	0.0593	0.0382	0.0374
ME4A	15.4862	6.0333	0.7112	0.0627	0.0246	0.0327
ME5A	2.5037	0.0040	0.0037	0.0047	0.0042	0.0034
ME5B	0.0683	0.0039	0.0038	0.0048	0.0043	0.0034

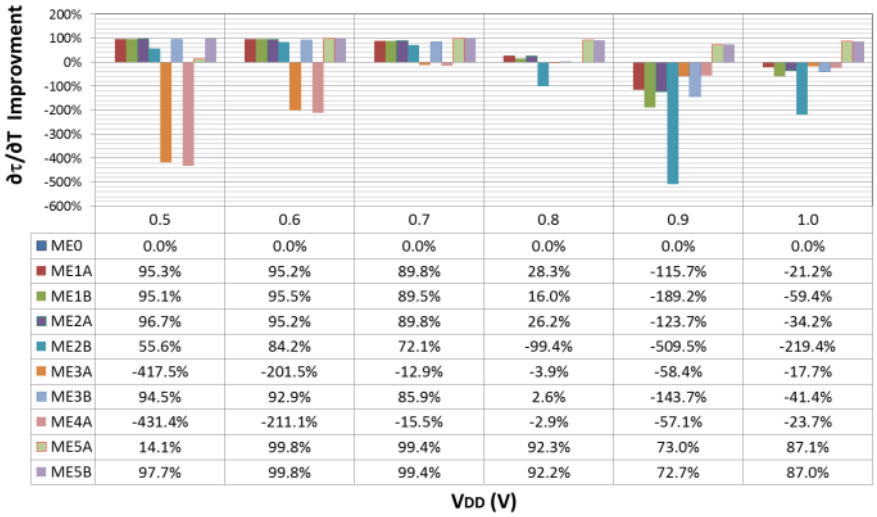
VDD (V)



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	-0.9252	-0.1681	0.0444	0.0788	0.0818	0.0786
ME1A	-6.6542	-0.3181	0.0911	0.1231	0.1205	0.1127
ME1B	-6.6589	-0.3315	0.0808	0.1148	0.1138	0.1071
ME2A	-1.0880	-0.1842	0.0640	0.1045	0.1080	0.1037
ME2B	-7.5556	-0.3581	0.1109	0.1503	0.1483	0.1393
ME3A	-4.8890	-0.2385	0.1025	0.1341	0.1322	0.1243
ME3B	-5.6601	-0.2647	0.1245	0.1617	0.1597	0.1505
ME4A	-4.9315	-0.2351	0.1121	0.1444	0.1421	0.1335
ME5A	-4.9157	-0.2579	0.1047	0.1424	0.1408	0.1317
ME5B	-5.5214	-0.2936	0.1001	0.1436	0.1436	0.1345

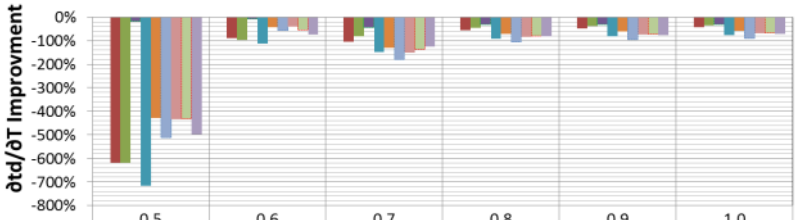
VDD (V)

Figure 4.29 Sensitivity τ and t_d to Temperature against V_{DD}



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
ME1A	95.3%	95.2%	89.8%	28.3%	-115.7%	-21.2%
ME1B	95.1%	95.5%	89.5%	16.0%	-189.2%	-59.4%
ME2A	96.7%	95.2%	89.8%	26.2%	-123.7%	-34.2%
ME2B	55.6%	84.2%	72.1%	-99.4%	-509.5%	-219.4%
ME3A	-417.5%	-201.5%	-12.9%	-3.9%	-58.4%	-17.7%
ME3B	94.5%	92.9%	85.9%	2.6%	-143.7%	-41.4%
ME4A	-431.4%	-211.1%	-15.5%	-2.9%	-57.1%	-23.7%
ME5A	14.1%	99.8%	99.4%	92.3%	73.0%	87.1%
ME5B	97.7%	99.8%	99.4%	92.2%	72.7%	87.0%

VDD (V)



	0.5	0.6	0.7	0.8	0.9	1.0
ME0	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
ME1A	-619.2%	-89.2%	-105.1%	-56.1%	-47.3%	-43.4%
ME1B	-619.7%	-97.2%	-81.8%	-45.6%	-39.1%	-36.3%
ME2A	-17.6%	-9.6%	-44.1%	-32.6%	-32.0%	-31.9%
ME2B	-716.7%	-113.0%	-149.8%	-90.7%	-81.3%	-77.2%
ME3A	-428.4%	-41.9%	-130.7%	-70.1%	-61.7%	-58.2%
ME3B	-511.8%	-57.5%	-180.2%	-105.0%	-95.3%	-91.5%
ME4A	-433.0%	-39.8%	-152.3%	-83.1%	-73.7%	-69.8%
ME5A	-431.3%	-53.4%	-135.7%	-80.6%	-72.1%	-67.6%
ME5B	-496.8%	-74.6%	-125.4%	-82.1%	-75.6%	-71.1%

VDD (V)

Figure 4.30 Sensitivity Improvement of τ and t_d to temperature against V_{DD}

4.3.5 Process Variations and Voltage Effect

The statistical analysis of τ and t_d in response to a ±3σ 1000 runs process variation and voltage supply reduction from 1V down to 0.5V appears in Table 4.1 and Table 4.2, showing their mean (μ), standard-deviations (σ) and percentage variability ($3\sigma/\mu$) against the voltage supply. The least variability percentages of τ over all voltage ranges is for ME5B and ME5A, for instance 12% at 1V and 36% at 0.5V, whereas the highest variability of τ is for ME2B, for instance 39% at 1V and 369% at 0.5V. On the other hand, the best variability percentages for t_d over all voltage ranges is for most MUTEXes, for instance around 28% at 0.5V, whereas the worst cases are for ME1A, ME1B and ME2B, over 38% at 0.5V. The optimum performance here is demonstrated by ME5A and ME5B, as both can tolerate ±3σ process variations with reduced voltage supply.

As process variations can influence the targeted τ to deviate beyond 3σ, it is

important to minimize its value as well as least deviation. In this part, the deviation from the mean is illustrated in Figure 4.31 to assess the deviation as a magnitude of variation as rated to the standard-deviation of the FO4 delay, in terms of τ and t_d. The graphs show ME0 has σ between 0.6 to 4 times that of FO4, whereas ME5A and ME5B has σ less than 0.15 of σ_{FO4}, and ME1A, M1B and ME2A do not exceed one standard deviation of FO4 delay. Figure 4.32 shows the clear improvement of σ for most M-MUTEXes and particularly for lower supply voltages.

Table 4.1 Response of τ to ±3σ 1000 runs Monte Carlo process variation

	τ	1V	0.9V	0.8V	0.7V	0.6V	0.5V
ME0	μ	14.0ps	17.1ps	22.8ps	35.9ps	71.7ps	188.2ps
	σ	0.6ps	1.0ps	2.03ps	4.9ps	14.5ps	49.4ps
	$3\sigma/\mu$	13.6%	18.0%	26.7%	41.1%	60.5%	78.8%
ME1A	μ	12.6ps	14.2ps	17.3ps	22.5ps	32.5ps	58.1ps
	σ	0.3ps	0.6ps	0.9ps	1.4ps	2.7ps	8.4ps
	$3\sigma/\mu$	7.1%	12.6%	15.4%	18.4%	24.6%	43.1%
ME1B	μ	13.1ps	13.9ps	15.7ps	19.4ps	27.4ps	49.4ps
	σ	0.18ps	0.3ps	0.6ps	1.0ps	2.2ps	7.5ps
	$3\sigma/\mu$	4.0%	6.6%	10.8%	15.9%	24.2%	45.7%
ME2A	μ	12.1ps	14.4ps	16.6ps	22.3ps	32.5ps	60.8ps
	σ	0.7ps	0.3ps	0.97ps	1.35ps	3.2ps	9.8ps
	$3\sigma/\mu$	17.3%	6.1%	17.5%	18.1%	29.9%	48.2%
ME2B	μ	21.1ps	23.0ps	26.7ps	34.1ps	55.1ps	171.7ps
	σ	2.75ps	3.1ps	4.2ps	7.2ps	23.8ps	211.1ps
	$3\sigma/\mu$	39.0%	40.2%	47.3%	63.3%	129.5%	368.8%
ME3A	μ	14.6ps	17.4ps	23.2ps	36.6ps	75.0ps	205.3ps
	σ	0.25ps	1.05ps	1.9ps	5.0ps	15.7ps	51.7ps
	$3\sigma/\mu$	5.1%	18.0%	25.1%	41.0%	62.7%	75.6%
ME3B	μ	15.2ps	17.7ps	22.4ps	30.1ps	45.7ps	85.7ps
	σ	0.53ps	0.96ps	1.55ps	2.14ps	4.5ps	15.5ps
	$3\sigma/\mu$	10.4%	16.2%	20.7%	21.4%	29.5%	54.3%
ME4A	μ	15.0ps	18.1ps	23.9ps	37.7ps	77.6ps	211.5ps
	σ	0.54ps	1.3ps	2.2ps	5.2ps	15.7ps	54.1ps
	$3\sigma/\mu$	10.9%	21.7%	27.6%	41.5%	60.8%	76.8%
ME5A	μ	1.89ps	2.33ps	3.01ps	4.2ps	6.77ps	13.6ps
	σ	76.8fs	0.11ps	0.16ps	0.27ps	0.57ps	1.64ps
	$3\sigma/\mu$	12.2%	14.1%	16.0%	19.3%	25.3%	36.3%
ME5B	μ	1.9ps	2.34ps	3.02ps	4.21ps	6.79ps	13.6ps
	σ	76.7fs	0.11ps	0.16ps	0.27ps	0.57ps	1.64ps
	$3\sigma/\mu$	12.1%	14.0%	15.8%	19.2%	25.1%	36.1%

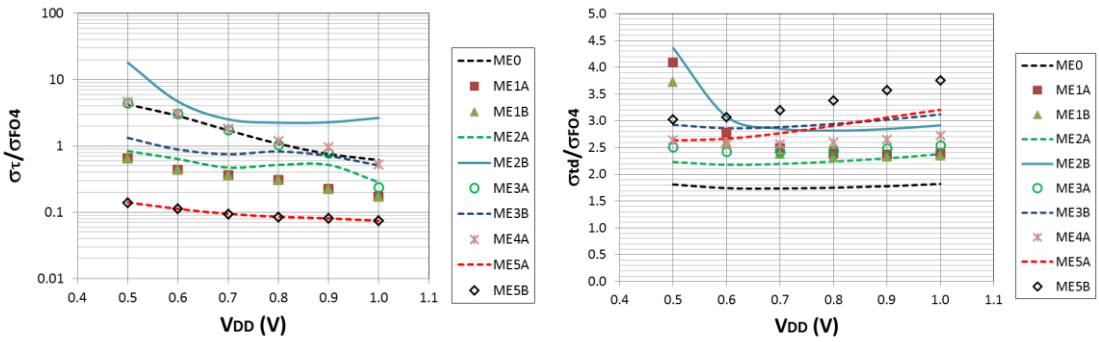
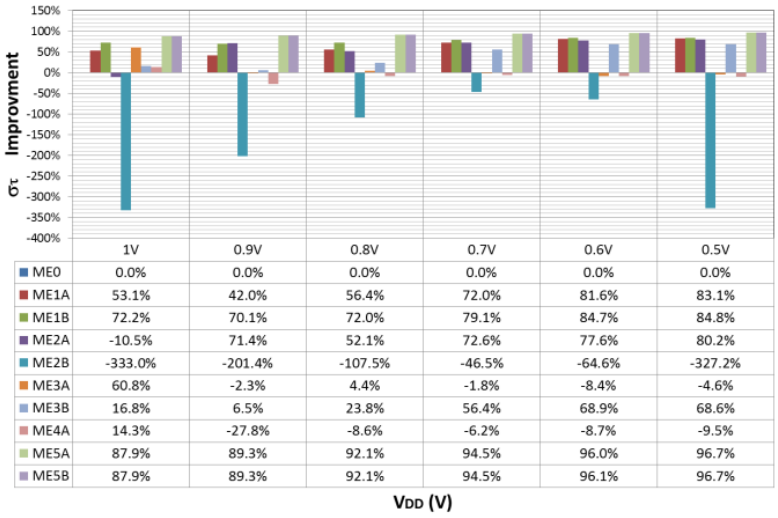


Figure 4.31 Ratio of standard deviation of τ and t_d to that of FO4 delay against V_{DD}

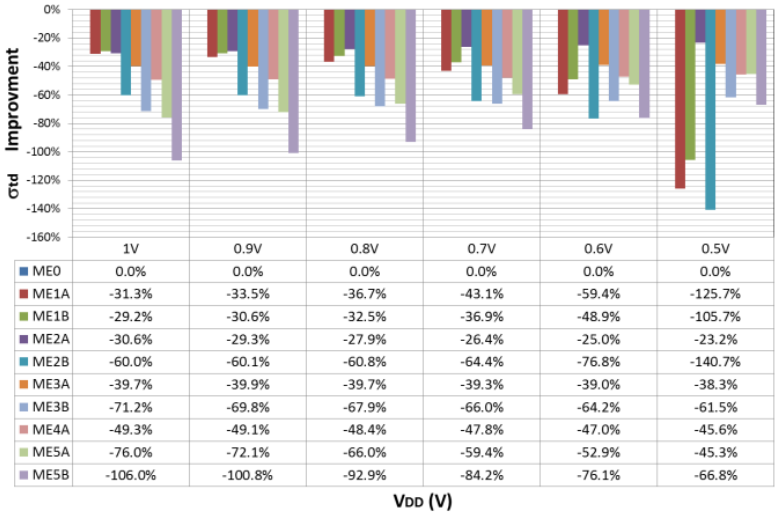
Table 4.2 Response of t_d to ±3σ 1000 runs Monte Carlo process variation

	t_d	1V	0.9V	0.8V	0.7V	0.6V	0.5V
ME0	μ	48.0ps	55.1ps	66.0ps	84.8ps	123.0ps	229.3ps
	σ	1.9ps	2.4ps	3.3ps	5.0ps	8.9ps	21.2ps
	$3\sigma/\mu$	11.8%	13.1%	14.9%	17.5%	21.6%	27.7%
ME1A	μ	66.0ps	76.0ps	91.9ps	119.7ps	179.0ps	358.7ps
	σ	2.5ps	3.2ps	4.5ps	7.1ps	14.1ps	47.7ps
	$3\sigma/\mu$	11.2%	12.6%	14.6%	17.8%	23.6%	39.9%
ME1B	μ	63.9ps	73.6ps	887.2ps	115.3ps	171.6ps	342.6ps
	σ	2.4ps	3.1ps	4.3ps	6.8ps	13.2ps	43.5ps
	$3\sigma/\mu$	11.4%	12.8%	14.7%	17.7%	23.0%	38.1%
ME2A	μ	64.0ps	72.8ps	86.5ps	109.8ps	156.9ps	285.8ps
	σ	2.5ps	3.1ps	4.2ps	6.3ps	11.1ps	26.0ps
	$3\sigma/\mu$	11.5%	12.8%	14.5%	17.1%	21.2%	27.3%
ME2B	μ	83.0ps	94.9ps	113.5ps	146.2ps	215.0ps	421.8ps
	σ	3.0ps	3.8ps	5.3ps	8.2ps	15.6ps	50.9ps
	$3\sigma/\mu$	10.9%	12.1%	13.9%	16.7%	21.8%	36.2%
ME3A	μ	71.8ps	81.9ps	97.3ps	123.4ps	175.2ps	313.3ps
	σ	2.6ps	3.4ps	4.6ps	6.9ps	12.3ps	29.3ps
	$3\sigma/\mu$	11.0%	12.3%	14.1%	16.8%	21.1%	28.0%
ME3B	μ	88.5ps	100.5ps	118.7ps	149.5ps	210.5ps	372.3ps
	σ	3.2ps	4.1ps	5.5ps	8.2ps	14.5ps	34.2ps
	$3\sigma/\mu$	10.9%	12.2%	13.9%	16.5%	20.7%	27.5%
ME4A	μ	77.2ps	87.8ps	104.0ps	131.3ps	185.3ps	328.5ps
	σ	2.8ps	3.6ps	4.9ps	7.3ps	13.0ps	30.8ps
	$3\sigma/\mu$	10.9%	12.2%	14.0%	16.7%	21.1%	28.1%
ME5A	μ	84.8ps	95.6ps	111.6ps	138.1ps	190.5ps	328.6ps
	σ	3.3ps	4.1ps	5.4ps	7.9ps	13.5ps	30.7ps
	$3\sigma/\mu$	11.7%	13.0%	14.6%	17.2%	21.3%	28.1%
ME5B	μ	97.3ps	109.5ps	127.6ps	157.5ps	216.8ps	373.0ps
	σ	3.9ps	4.8ps	6.3ps	9.1ps	15.6ps	35.3ps
	$3\sigma/\mu$	12.0%	13.2%	14.9%	17.4%	21.6%	28.4%



	1V	0.9V	0.8V	0.7V	0.6V	0.5V
ME0	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
ME1A	53.1%	42.0%	56.4%	72.0%	81.6%	83.1%
ME1B	72.2%	70.1%	72.0%	79.1%	84.7%	84.8%
ME2A	-10.5%	71.4%	52.1%	72.6%	77.6%	80.2%
ME2B	-333.0%	-201.4%	-107.5%	-46.5%	-64.6%	-327.2%
ME3A	60.8%	-2.3%	4.4%	-1.8%	-8.4%	-4.6%
ME3B	16.8%	6.5%	23.8%	56.4%	68.9%	68.6%
ME4A	14.3%	-27.8%	-8.6%	-6.2%	-8.7%	-9.5%
ME5A	87.9%	89.3%	92.1%	94.5%	96.0%	96.7%
ME5B	87.9%	89.3%	92.1%	94.5%	96.1%	96.7%

VDD (V)



	1V	0.9V	0.8V	0.7V	0.6V	0.5V
ME0	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%
ME1A	-31.3%	-33.5%	-36.7%	-43.1%	-59.4%	-125.7%
ME1B	-29.2%	-30.6%	-32.5%	-36.9%	-48.9%	-105.7%
ME2A	-30.6%	-29.3%	-27.9%	-26.4%	-25.0%	-23.2%
ME2B	-60.0%	-60.1%	-60.8%	-64.4%	-76.8%	-140.7%
ME3A	-39.7%	-39.9%	-39.7%	-39.3%	-39.0%	-38.3%
ME3B	-71.2%	-69.8%	-67.9%	-66.0%	-64.2%	-61.5%
ME4A	-49.3%	-49.1%	-48.4%	-47.8%	-47.0%	-45.6%
ME5A	-76.0%	-72.1%	-66.0%	-59.4%	-52.9%	-45.3%
ME5B	-106.0%	-100.8%	-92.9%	-84.2%	-76.1%	-66.8%

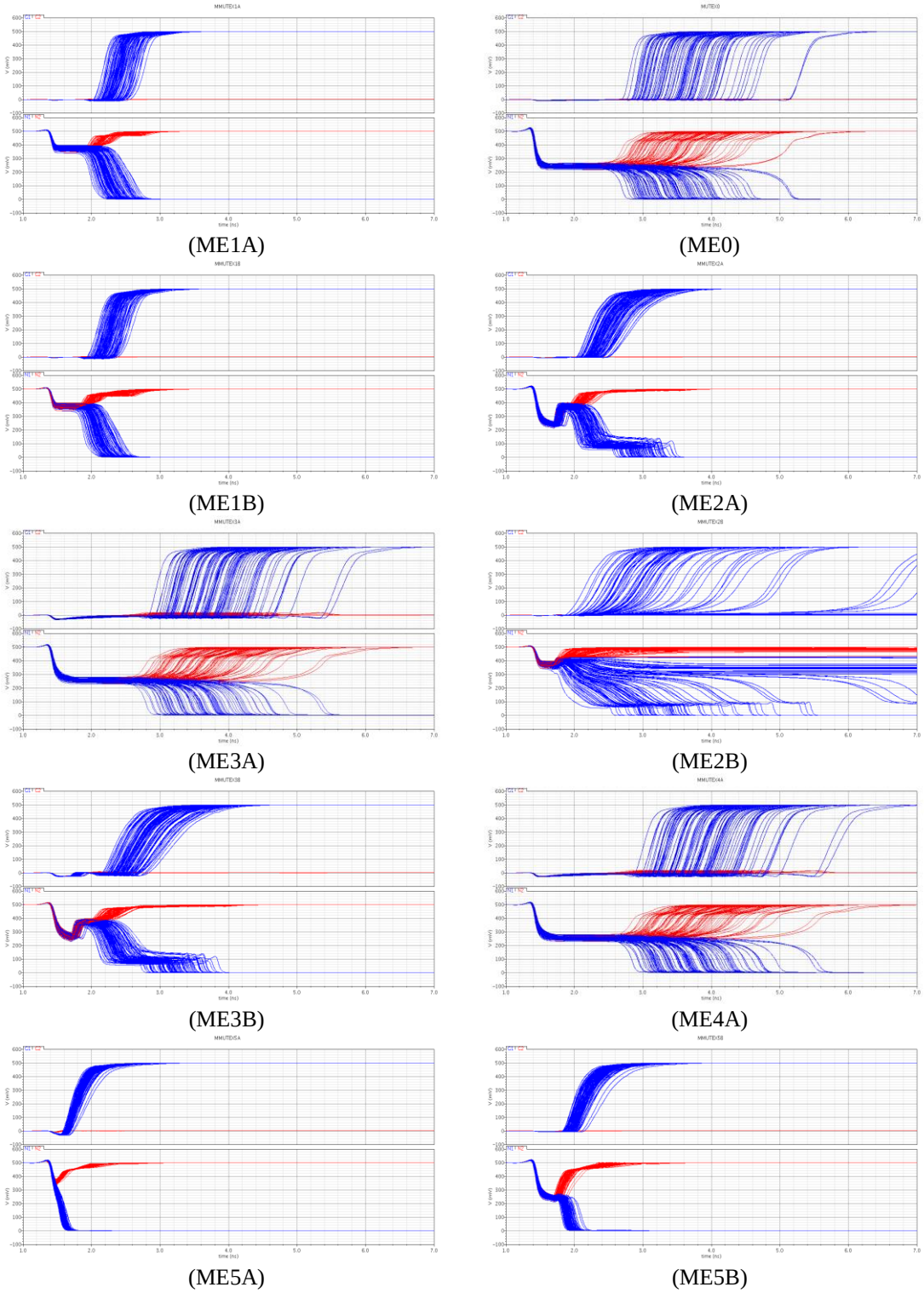
VDD (V)

Figure 4.32 Improvement of standard deviation of τ and t_d to that of ME0 against V_{DD}

The waveforms at the outputs (G1, G2) and internal nodes (N1, N2) in response to process variations are shown in Figure 4.33. They clearly show the metastability resolution time taken by ME5A and ME5B are the best over all the runs, in comparison to others. It also presents the worst responses, experienced in ME0, ME2B, ME3A and ME4A. The other M-MUTEXes show significant improvement over the typical case.

In Figure 4.33, the waveforms of ME2A, ME2B and ME3B show that when metastability resolves, the node voltage resolving to zero-volts is leveled at about 100mV for a period of time before completely falling to 0V. This is caused by the delay time taken the metastability error detection circuit to disable the current sources. In Figure 4.33, the metastability resolution of ME2B is taking a longer time than other circuits, this is because ME2B uses two τ increasing methods that may result in a conflict; one method is active all the time and the other is only activated

after metastability is detected. For instance, the latter method may reinitiate metastability while the former method started resolving the first one.



(ME1A) (ME0)

(ME1B) (ME2A)

(ME3A) (ME2B)

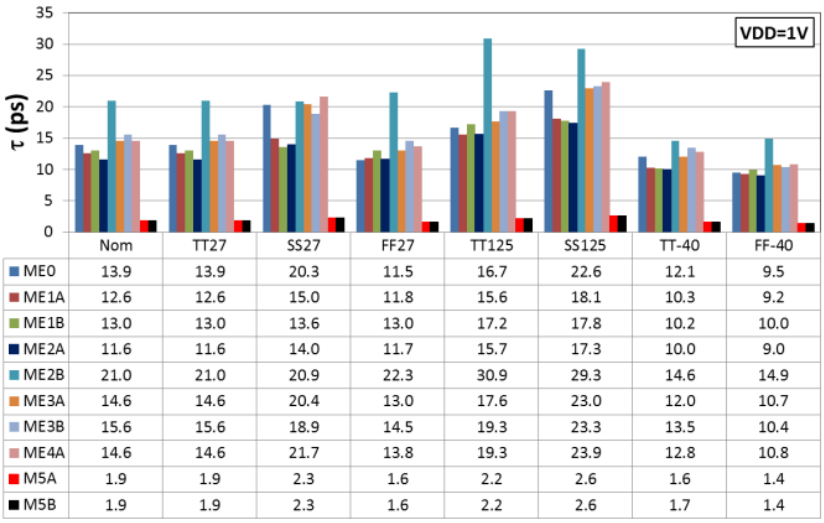
(ME3B) (ME4A)

(ME5A) (ME5B)

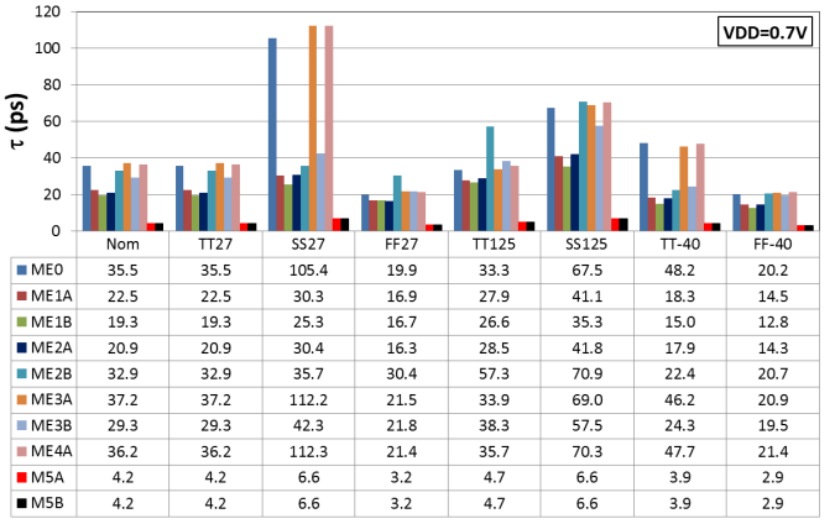
Figure 4.33 Monte Carlo waveforms of metastability resolution at 0.5V

4.3.6 Process, Voltage and Temperature Corners Effect

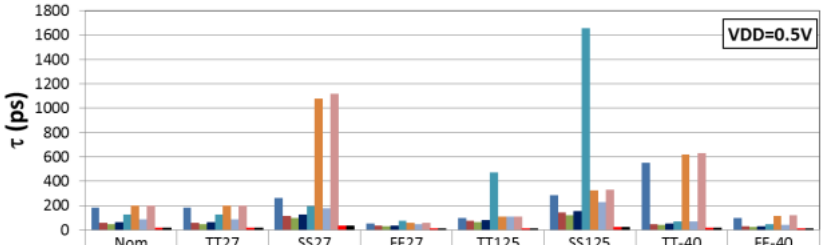
The results in Figure 4.34 and Figure 4.35, demonstrate the response of PVT deviation impact on τ and t_d using process corners TT, SS and FF at corner temperature 27°C, 125°C and –40°C and supply voltages 1V, 0.7V and 0.5V.



	Nom	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
ME0	13.9	13.9	20.3	11.5	16.7	22.6	12.1	9.5
ME1A	12.6	12.6	15.0	11.8	15.6	18.1	10.3	9.2
ME1B	13.0	13.0	13.6	13.0	17.2	17.8	10.2	10.0
ME2A	11.6	11.6	14.0	11.7	15.7	17.3	10.0	9.0
ME2B	21.0	21.0	20.9	22.3	30.9	29.3	14.6	14.9
ME3A	14.6	14.6	20.4	13.0	17.6	23.0	12.0	10.7
ME3B	15.6	15.6	18.9	14.5	19.3	23.3	13.5	10.4
ME4A	14.6	14.6	21.7	13.8	19.3	23.9	12.8	10.8
M5A	1.9	1.9	2.3	1.6	2.2	2.6	1.6	1.4
M5B	1.9	1.9	2.3	1.6	2.2	2.6	1.7	1.4

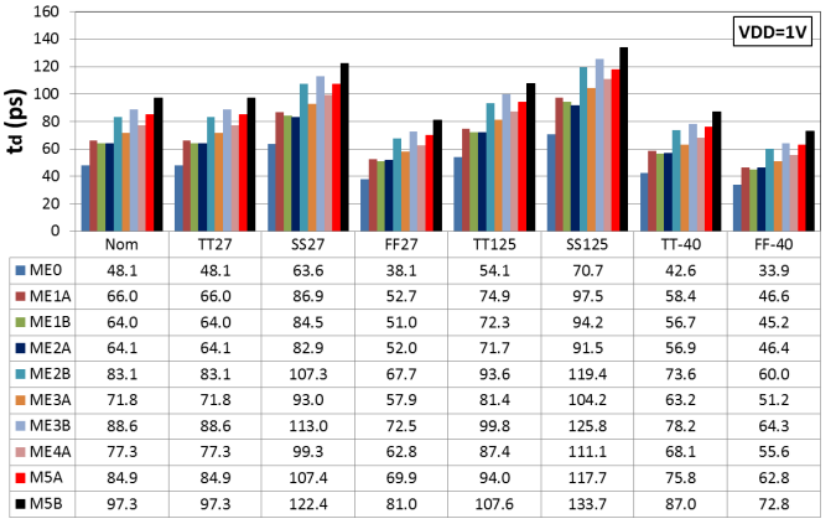


	Nom	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
ME0	35.5	35.5	105.4	19.9	33.3	67.5	48.2	20.2
ME1A	22.5	22.5	30.3	16.9	27.9	41.1	18.3	14.5
ME1B	19.3	19.3	25.3	16.7	26.6	35.3	15.0	12.8
ME2A	20.9	20.9	30.4	16.3	28.5	41.8	17.9	14.3
ME2B	32.9	32.9	35.7	30.4	57.3	70.9	22.4	20.7
ME3A	37.2	37.2	112.2	21.5	33.9	69.0	46.2	20.9
ME3B	29.3	29.3	42.3	21.8	38.3	57.5	24.3	19.5
ME4A	36.2	36.2	112.3	21.4	35.7	70.3	47.7	21.4
M5A	4.2	4.2	6.6	3.2	4.7	6.6	3.9	2.9
M5B	4.2	4.2	6.6	3.2	4.7	6.6	3.9	2.9

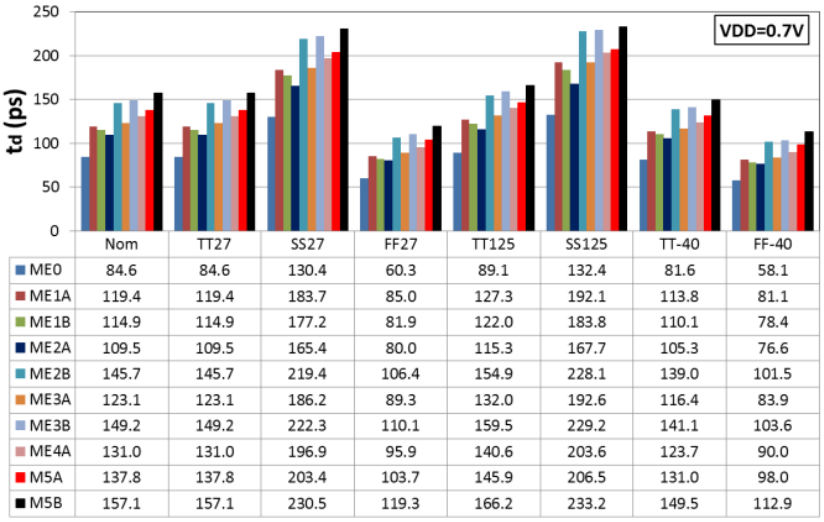


	Nom	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
ME0	181.0	181.0	260.0	51.1	95.6	288.0	551.0	100.8
ME1A	57.1	57.1	117.8	34.3	73.6	142.9	49.9	30.3
ME1B	48.1	48.1	99.5	30.0	64.5	121.9	42.2	25.6
ME2A	61.4	61.4	125.5	34.1	77.9	151.7	51.8	28.7
ME2B	124.0	124.0	193.8	75.7	469.9	1658.0	68.4	44.4
ME3A	200.0	200.0	1078.0	55.9	106.9	323.8	619.9	117.7
ME3B	84.9	84.9	176.7	49.9	110.4	225.7	72.0	41.5
ME4A	200.7	200.7	1120.0	58.2	109.7	332.9	632.3	118.7
M5A	13.4	13.4	32.6	7.5	12.2	23.5	16.3	8.2
M5B	13.4	13.4	32.6	7.5	12.3	23.4	16.3	8.2

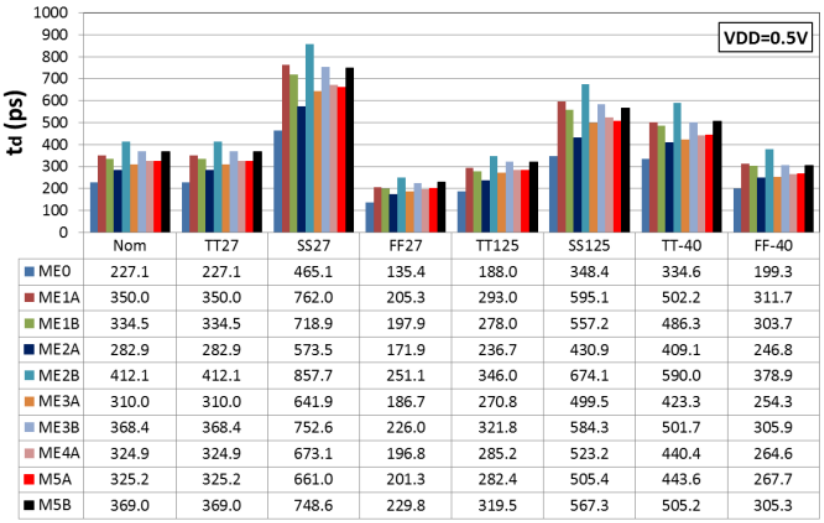
Figure 4.34 PVT corners effect on τ



	Nom	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
ME0	48.1	48.1	63.6	38.1	54.1	70.7	42.6	33.9
ME1A	66.0	66.0	86.9	52.7	74.9	97.5	58.4	46.6
ME1B	64.0	64.0	84.5	51.0	72.3	94.2	56.7	45.2
ME2A	64.1	64.1	82.9	52.0	71.7	91.5	56.9	46.4
ME2B	83.1	83.1	107.3	67.7	93.6	119.4	73.6	60.0
ME3A	71.8	71.8	93.0	57.9	81.4	104.2	63.2	51.2
ME3B	88.6	88.6	113.0	72.5	99.8	125.8	78.2	64.3
ME4A	77.3	77.3	99.3	62.8	87.4	111.1	68.1	55.6
M5A	84.9	84.9	107.4	69.9	94.0	117.7	75.8	62.8
M5B	97.3	97.3	122.4	81.0	107.6	133.7	87.0	72.8



	Nom	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
ME0	84.6	84.6	130.4	60.3	89.1	132.4	81.6	58.1
ME1A	119.4	119.4	183.7	85.0	127.3	192.1	113.8	81.1
ME1B	114.9	114.9	177.2	81.9	122.0	183.8	110.1	78.4
ME2A	109.5	109.5	165.4	80.0	115.3	167.7	105.3	76.6
ME2B	145.7	145.7	219.4	106.4	154.9	228.1	139.0	101.5
ME3A	123.1	123.1	186.2	89.3	132.0	192.6	116.4	83.9
ME3B	149.2	149.2	222.3	110.1	159.5	229.2	141.1	103.6
ME4A	131.0	131.0	196.9	95.9	140.6	203.6	123.7	90.0
M5A	137.8	137.8	203.4	103.7	145.9	206.5	131.0	98.0
M5B	157.1	157.1	230.5	119.3	166.2	233.2	149.5	112.9



	Nom	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
ME0	227.1	227.1	465.1	135.4	188.0	348.4	334.6	199.3
ME1A	350.0	350.0	762.0	205.3	293.0	595.1	502.2	311.7
ME1B	334.5	334.5	718.9	197.9	278.0	557.2	486.3	303.7
ME2A	282.9	282.9	573.5	171.9	236.7	430.9	409.1	246.8
ME2B	412.1	412.1	857.7	251.1	346.0	674.1	590.0	378.9
ME3A	310.0	310.0	641.9	186.7	270.8	499.5	423.3	254.3
ME3B	368.4	368.4	752.6	226.0	321.8	584.3	501.7	305.9
ME4A	324.9	324.9	673.1	196.8	285.2	523.2	440.4	264.6
M5A	325.2	325.2	661.0	201.3	282.4	505.4	443.6	267.7
M5B	369.0	369.0	748.6	229.8	319.5	567.3	505.2	305.3

Figure 4.35 PVT corners effect on t_d

The delay time of the M-MUTEXes, as was indicated by the previously discussed results, shows a consistent increase among all conditions from that of ME0, from around 1.4X to nearly two times. However, using some feedback techniques such as ME1A, ME1B and ME2A, provides significant control over PVT variations and to

overcome an anticipated metastability, so providing a reliable design.

4.4 Summary

This chapter focused on the classical MUTEX circuit (ME0) as an arbiter. It was discussed and analyzed. Although, the ME0 has an optimum speed with minimum deviation, the presented results indicate that its metastability resolution time constant τ is very sensitive to voltage supply reduction, process variations and temperature increase as well as high sensitivity to latch and load size changes. This problem leads to an investigation of how to minimize the value of τ and how to provide a controlling mechanism to enhance its tolerance to common deviations. Therefore, a number of modifications at the circuit and transistor levels were proposed and investigated, to adjust the value τ and its tolerance to PVT. The modifying techniques are based on increasing the internal gain A by adding active current sources, reducing the capacitive loading C_{Load}, boosting the transconductance g_m of the latch via a metastability error detection circuit, compensating the existing Miller capacitance C_M via cascoding, and adding asymmetry between the NAND gates to maneuver the metastable point. Nine Modified MUTEX circuits are proposed, where each one had either one or two modifications. The results showed that four circuits had little or almost no improvements, namely ME2B, ME3A, ME3B and ME4A, while five techniques, namely ME1A, ME1B, ME2A, ME5A and ME5B, showed significant improvements by reducing τ and maintaining high tolerance towards process variations, lower V_{DD} and temperature variations as well as latch and load size change. In other words, compensating the existing Miller capacitance, as in ME3A and ME4A, has negligible effect on reducing τ or its variability because C_M is small compared to the overall load capacitance, which minimizes it effect on τ. In addition, adding current sources along with g_m boosting technique, as in ME2B, increases τ and its variability because of the response time of the current-source method is quicker than that of the g_m booster due to longer delay in the detection circuit. For instance, in a ME2B circuit, if metastability starts to resolve by the current-source technique and shortly after the g_m booster is activated, then the resolution of metastability is disturbed and prolonged. On the other hand, M-MUTEX circuits with current sources ME1A, with current sources and reduced C_{Load} ME1B or with

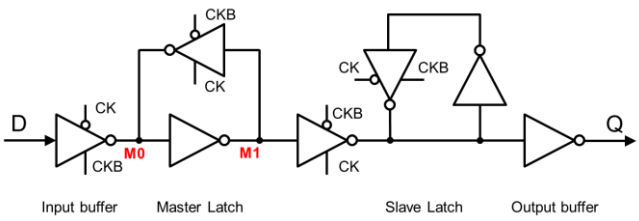
the g_m booster ME2A significantly reduces τ and improves its tolerance. The maneuvered asymmetry between the NAND gates, as in ME5A and ME5B, significantly reduces τ and enhances its tolerance by almost 90% compared to that of ME0. However, the delay time is increased in all modified circuits, because of the increased loading at the inputs, the outputs and the internal nodes by the additional devices.

Chapter 5 Variation-Tolerant Synchronizer Design

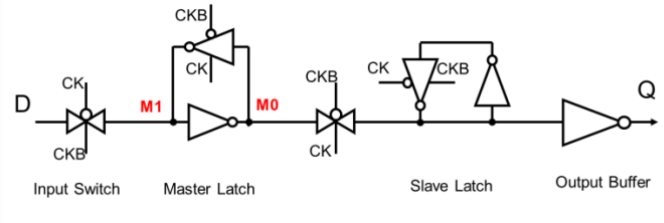
In this chapter, a number of techniques are presented, to further reduce the metastability failures in synchronizer circuits, particularly in the presence of process variations, reduced supply voltage and temperature corners. The proposed techniques are based on two methods. The first method utilizes the wagging structure, introduced in Chapter 3, to extend the available resolution time in conjunction with improved τ latches. The second method exploits additional circuitry to detect metastable events during normal operation and enforces a correction process to cut the resolution time down from uncertainty to certain time, which produces a very low value of τ.

5.1 Typical Synchronizer Circuits

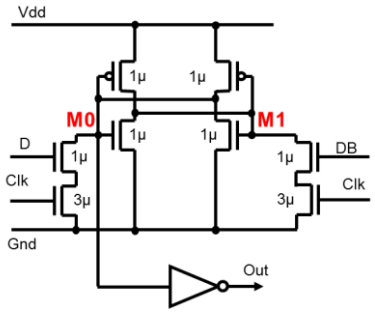
For the purpose of evaluating the proposed technique in this chapter, four synchronizer circuits are taken into consideration. They are based on edge-triggered D flip-flops with a master-latch and slave-latch, and are shown in Figure 5.1, which were discussed earlier in Chapter 3. The latches are composed of one of the following circuits; switched-inverter latches (or Clocked CMOS C2MOS), Transmission-Gate (TG) latches, Jamb Latch (JL) and Robust Latch (RL). The C2MOSFF and TGFF symbolize the common flip-flop circuits used in digital circuits. The Jamb Latch is particularly designed as a synchronizer circuit with a small τ, and the Robust Latch is a special synchronizer circuit that provides a small value of τ at a low voltage supply. In simulations, the former two, shown in Figure 5.1(a) and Figure 5.1(b), are constructed with 1μm n-type transistors and 2μm p-type transistors, and the latter two are composed without the reset part; instead, the inversion of data signal is used, and their transistor sizes are shown in Figure 5.1(c) and Figure 5.1(d).



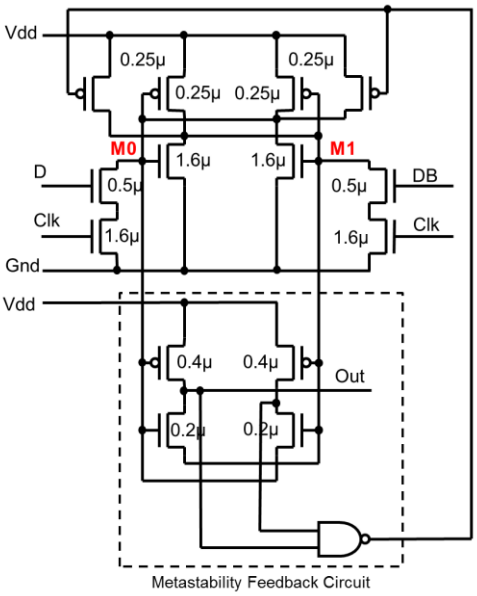
(a) C2MOS Flip-Flop



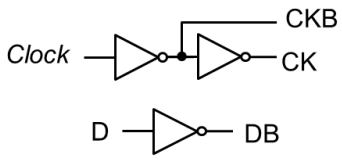
(b) TG Flip-Flop



(c) Jamb Latch with differential input



(d) Robust Latch with differential input



Buffered clock driver and inverted data

Figure 5.1 Latch and flip-flop circuits

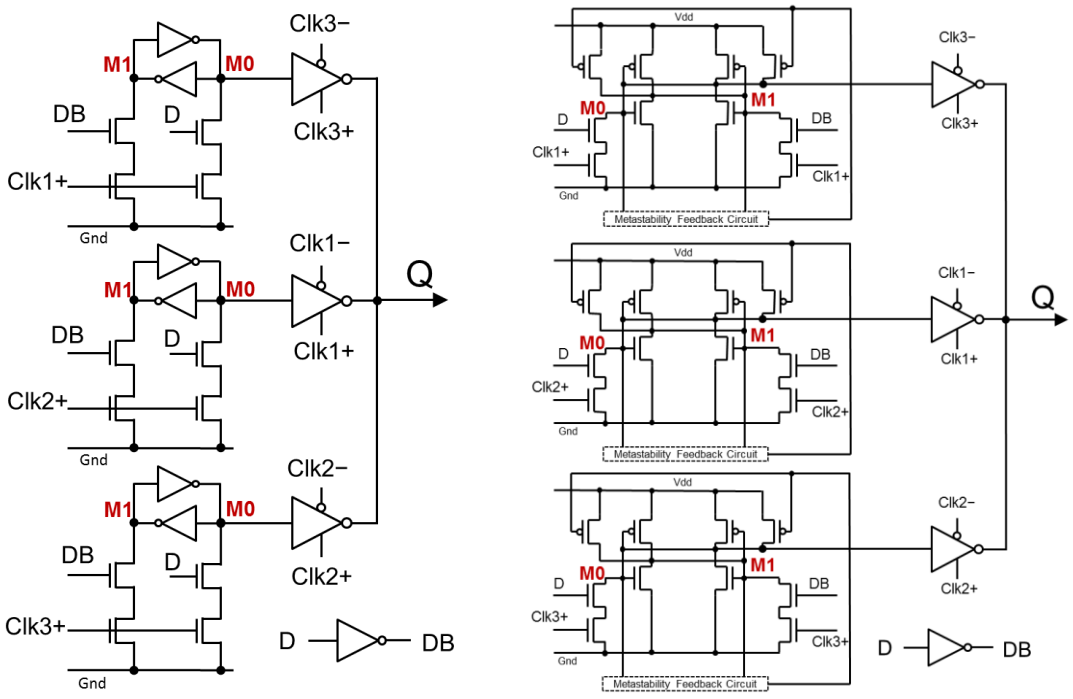
5.2 Variation-Tolerant Wagging Synchronizer

The concept of wagging was introduced earlier in Chapter 3. This section proposes two scenarios to improve the robustness of the wagging synchronizer. The first method concentrates on the type of latch used in the wagging synchronizer, and the second on extending the number of synchronizing cycles during operation of the wagging synchronizer.

5.2.1 Fast and Robust Wagging Synchronizer

In a similar manner to flip-flops, the wagging synchronizer can be further improved by either adjusting the latches to provide a lower value of τ, or inserting an additional path to increase t_R. The wagging synchronizer can be constructed using the fast τ Jamb Latches instead of a typical switched-inverter (C2MOS) latch. This arrangement, illustrated in Figure 5.2(a), provides the synchronizer with a better performance in terms of latency and failure rate, because it will have the faster resolution time constant of the Jamb Latch and the longer resolution time of the wagging structure.

In order to improve the reliability of the wagging synchronizer at lower V_{DD} operating points, Robust Latches can be used. This arrangement is illustrated in Figure 5.2(b). The output of the latch is taken from either node of the cross-coupled inverters, which will drive out either Q or its inverse. The connection shown in Figure 5.2 drives the output buffer with the inverted store value to drive out Q which follows D.



(a) Fast τ latch

(b) Robust τ latch

Figure 5.2 Improved τ wagging synchronizer circuits

5.2.2 Wagging Synchronizer with Reconfigurable Resolution Time

In this section, a reconfigurable wagging synchronizer is presented, which is suitable for operating at lower supply voltages and high variability environments. In multiple voltage designs, synchronizers require careful characterization for each supply voltage operating point, in terms of reliability; the use of a two-clock cycle synchronizer may be enough at nominal V_{DD} points but not enough at low-V_{DD} points. While extending the resolution time, by extending the chain of flip-flops, boosts the reliability and increases the overall latency. The wagging synchronizer can be used with a reconfigurable clocking circuit to increase the reliability at low-V_{DD} and reduce the latency at high-V_{DD} points.

A reconfigurable Clocking Control Circuit (CCC) to operate a six-way wagging synchronizer is proposed. This circuit can enable three-paths and up to six-paths of the wagging synchronizer, as presented in Table 5.1. The design process of this CCC is in a similar manner to that described in Chapter 3. The STG for each case in Table 5.1 was drawn to generate a set of logic functions that control the clocking phases, and then optimize them into matching stages. The circuit cell to control a clock signal is optimized and constructed as in Figure 5.3, where each cell is reliant on its preceding clock and control signals (C_{i-1} and $\overline{S_{i-1}}$) to set its clock signal (C_i) and on the succeeding control signal (S_{i+1}) to reset its control signal (S_i), where each transition is coordinated with the main clock signal. The clocking signal (Clk_i) is produced from the inverse of $\overline{C_i}$ and forward directly to the input latches of the wagging synchronizer, to reduce the load at the internal node C_i. A timing diagram of these signals in one CCC circuit is depicted in Figure 5.4.

To add re-configurability to the wagging synchronizer as described Table 5.1, the connecting paths between the cells in the clocking circuit to the synchronizer are multiplexed, as shown in Figure 5.5(a), based on the configuration signals X0 and X1, which are decoded into a 4-bit code (y0, y1, y2 and y3), as shown in Figure 5.5(b). New values of X0 and X1 can only be set during Clk1 phase, not to create any conflicts. A set signal is used to establish the clocking signals by setting signal C_1 to one and resetting other signals from C_2 to C_6 to zero.

The circuit design, in Figure 5.5, was realized using UMC90nm CMOS process technology. Simulation waveforms of the generated clock signals are shown in Figure 5.6 at 1V and 0.4V to demonstrate operational re-configurability of the

CCC. Initially, it shows the clocking signals were reset to zero, and only C_1 is set to one, to establish the CCC operation. It also shows the configuration code (X0 and X1) changes in the sequence of (00→01→10→11), which is allowed to transit during the C_1 phase. When the configuration code was is to 00, only three clocking signals (C_1 to C_3) are produced. Then, when it is set to 01, four clocking signals (C_1 to C_4) are produced. After that, when it is set 10, five clocking signals (C_1 to C_5) are produced. Finally, when it is set 11, six clocking signals (C_1 to C_6) are produced.

Table 5.1 6 way reconfigurable wagging synchronizer

Structure	X0	X1	Enabled Clocks	Reliability	Latency
3 way	0	0	Clk1 to Clk3	Better than 2FF	2 Cycles
4 way	0	1	Clk1 to Clk4	Better than 3FF	3 Cycles
5 way	1	0	Clk1 to Clk5	Better than 4FF	4 Cycles
6 way	1	1	Clk1 to Clk6	Better than 5FF	5 Cycles

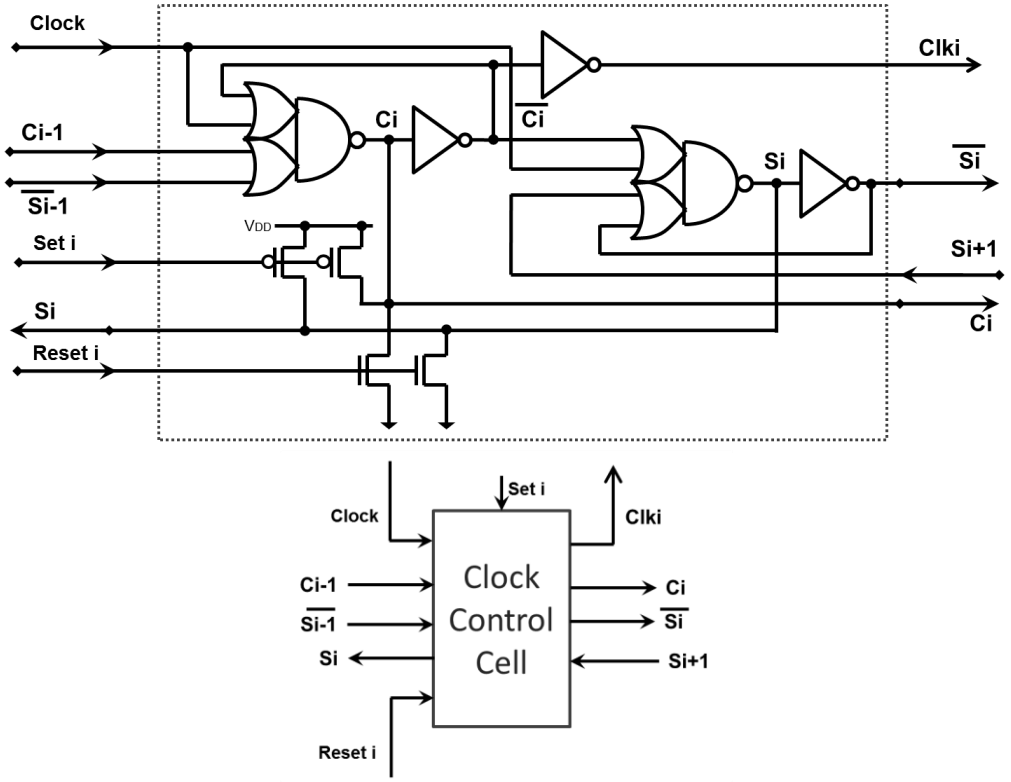


Figure 5.3 An asynchronous state cell to control a clocking signal

The clocking control circuit proposed in Figure 5.5 has operational limits to operate with the main system clock. Therefore, the maximum delay time between the rising edges of the main clock and clocking signals was measured to obtain the minimum pulse width of the main clock to operate the clocking circuit efficiently. Table 5.2 shows the minimum pulse width of the main clock, minimum

clock cycle and the maximum clock frequency to operate the clocking circuit at supply voltages from 0.4V to 1.1V and at typical and slow process corners. At V_{DD} of 1V, the CCC can operate with a maximum of 2GHz clock frequency at a typical process corner and 1.5GHz at a slow process corner. Whereas, at V_{DD} of 0.4V, the CCC can operate with a maximum of 178MHz main clock frequency at the typical corner and 46.7MHz at the slow corner.

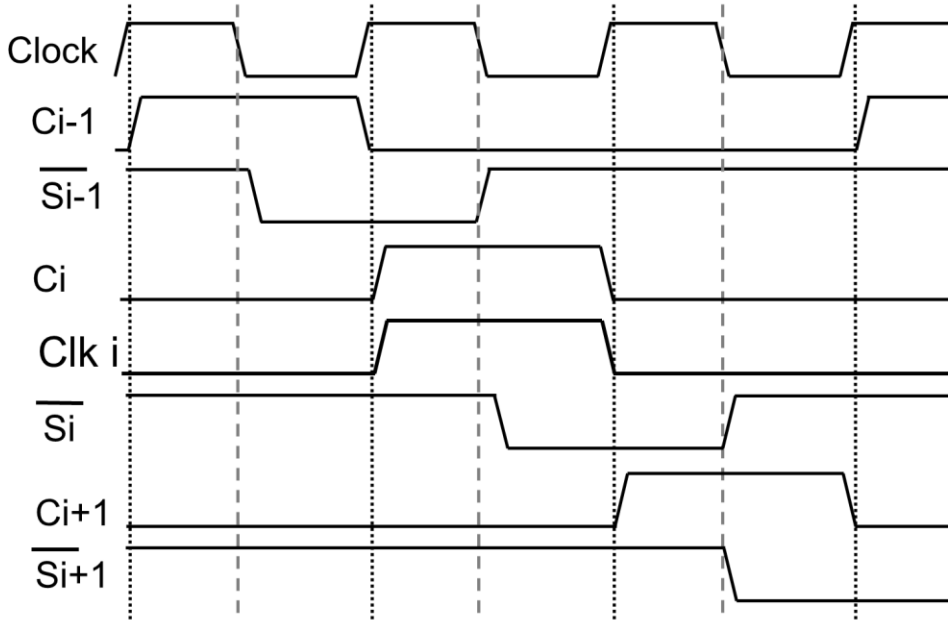
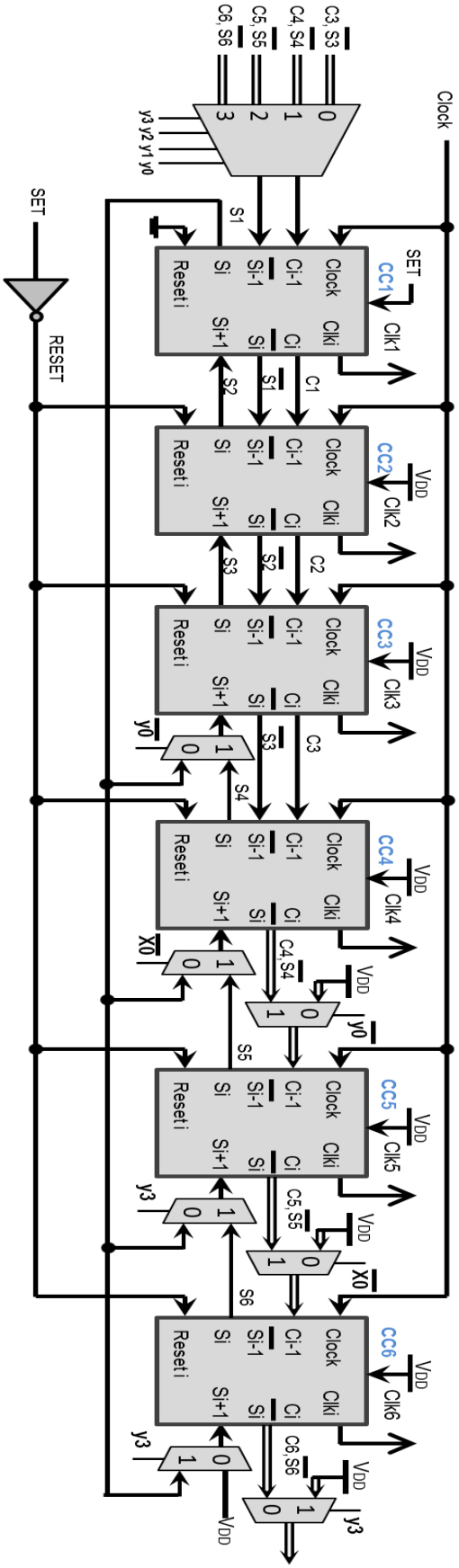


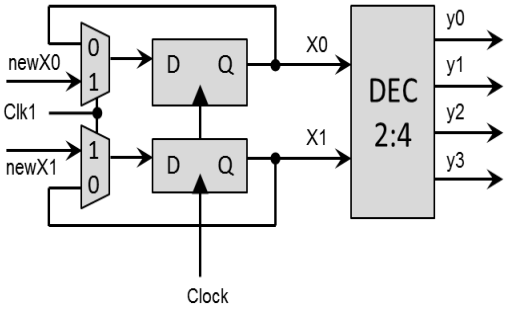
Figure 5.4 Timing diagram of signals in one CCC

Table 5.2 Maximum main clock frequency to operate the clocking circuit

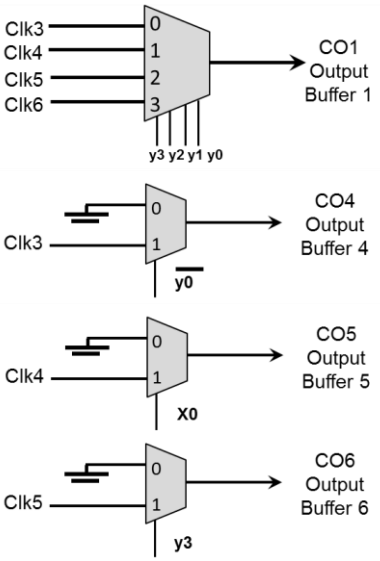
Process	**Typical Corner**			**Slow Corner**		
V_{DD} (V)	Clock pulse width (ns)	Min. T_{clk} (ns)	**Max f_{clk} (MHz)**	Clock pulse width (ns)	Min. T_{clk} (ns)	**Max f_{clk} (MHz)**
0.4	2.805	5.61	**178**	10.71	21.42	**46.7**
0.5	1.145	2.29	**437**	2.807	5.614	**178**
0.6	0.66	1.32	**758**	1.24	2.48	**403**
0.7	0.46	0.92	**1092**	0.74	1.48	**675**
0.8	0.355	0.71	**1417**	0.52	1.04	**958**
0.9	0.29	0.58	**1721**	0.4038	0.8075	**1238**
1	0.25	0.5	**2000**	0.3318	0.6635	**1507**
1.1	0.22	0.44	**2253**	0.2851	0.5702	**1753**



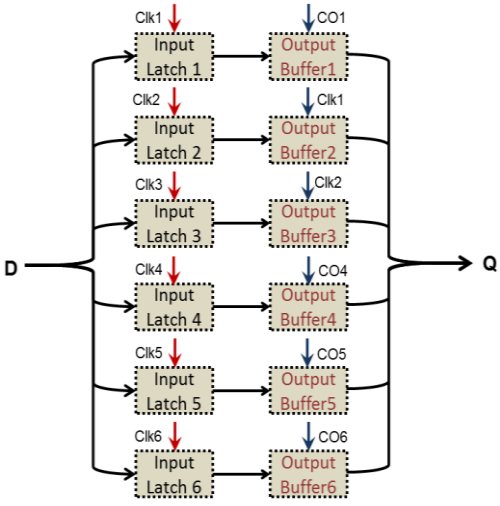
(a) Cells combined together



(b) Synchronized decoder

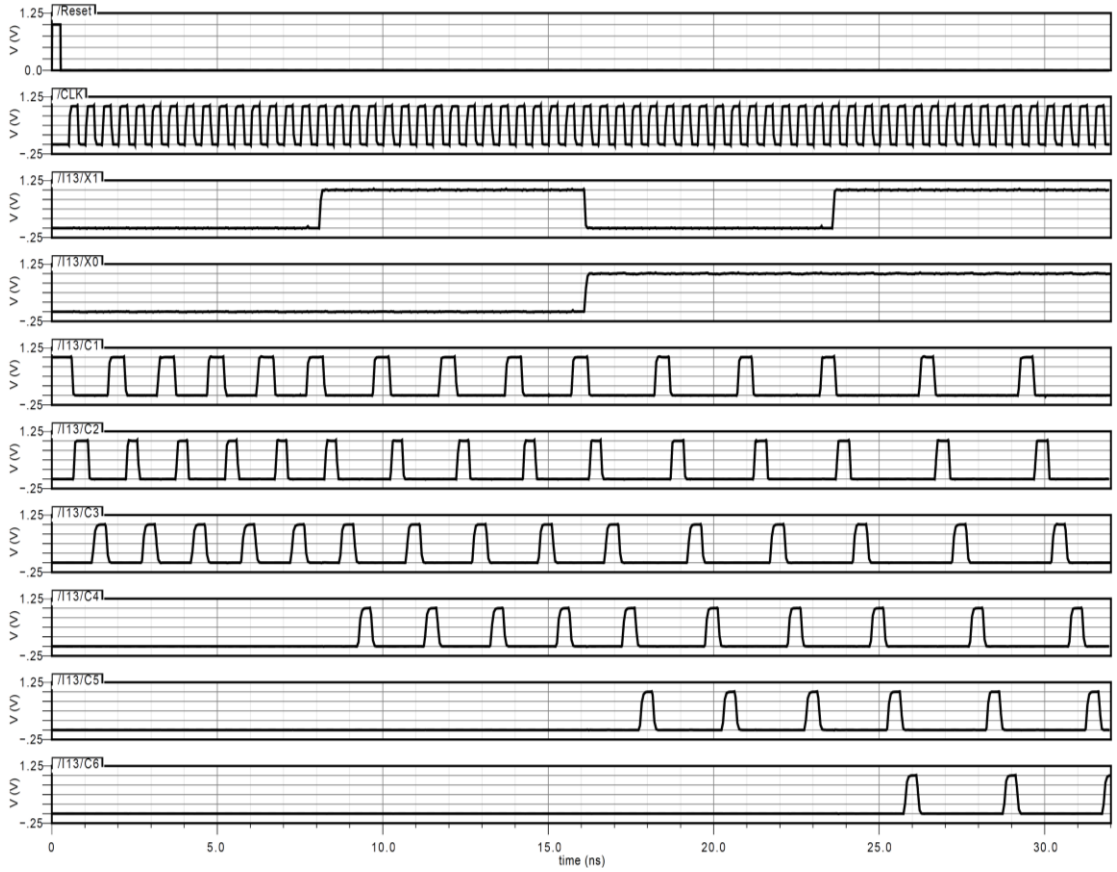


(c) Controlling output buffer clock signals in scheme (d) below

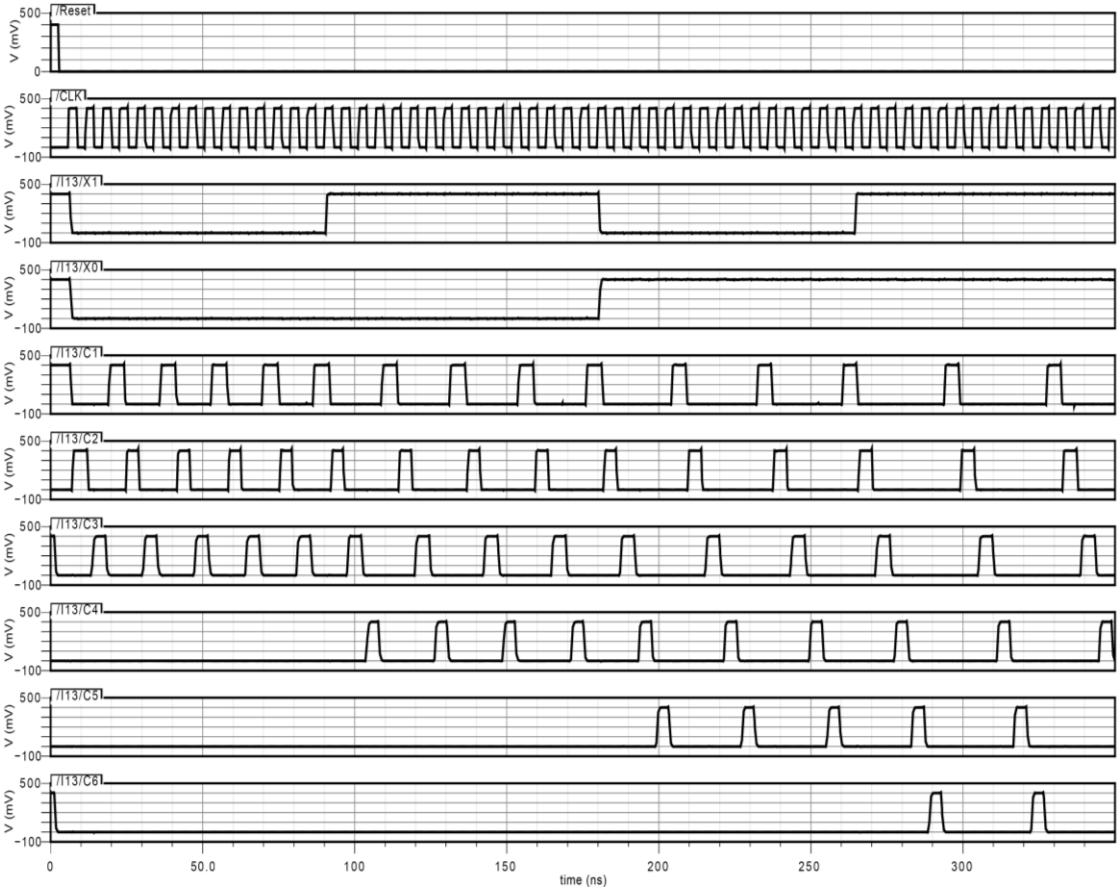


(d) Six way wagging structure

Figure 5.5 Reconfiguring CCC to produce clocking signals from three to six signals



(a) waveforms at 1V



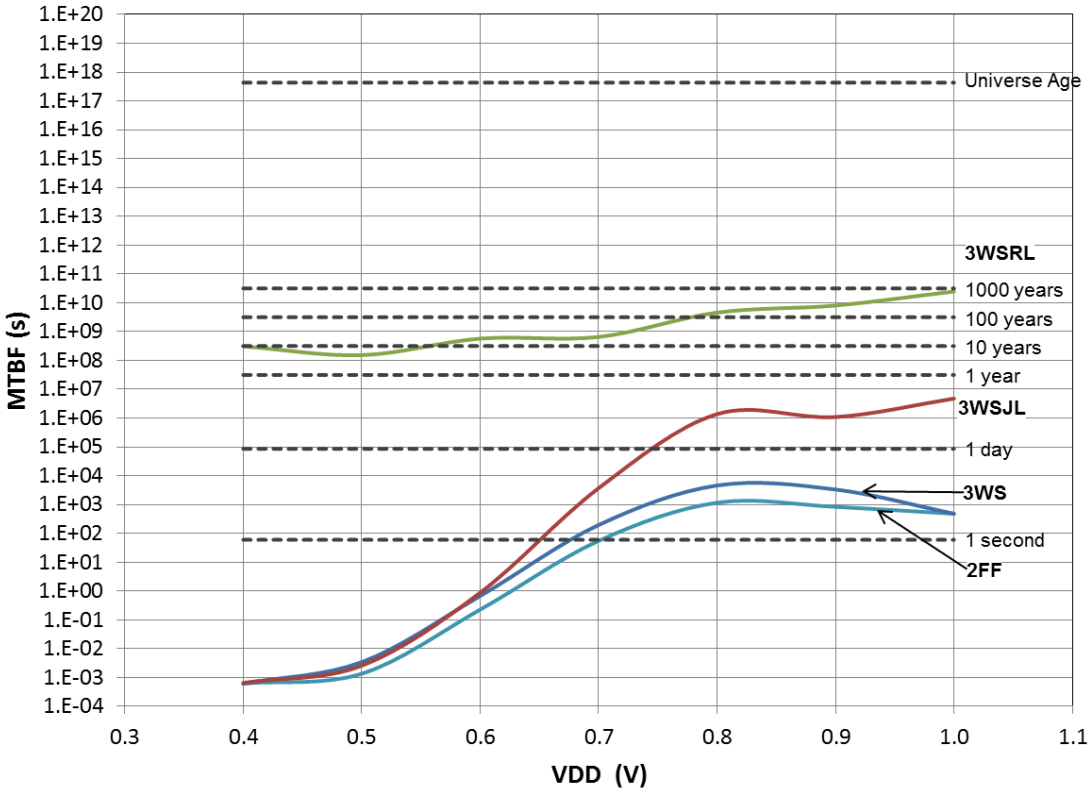
(b) waveforms at 0.4V

Figure 5.6 Reconfigurable CCC operates at 1.0V and 0.4V

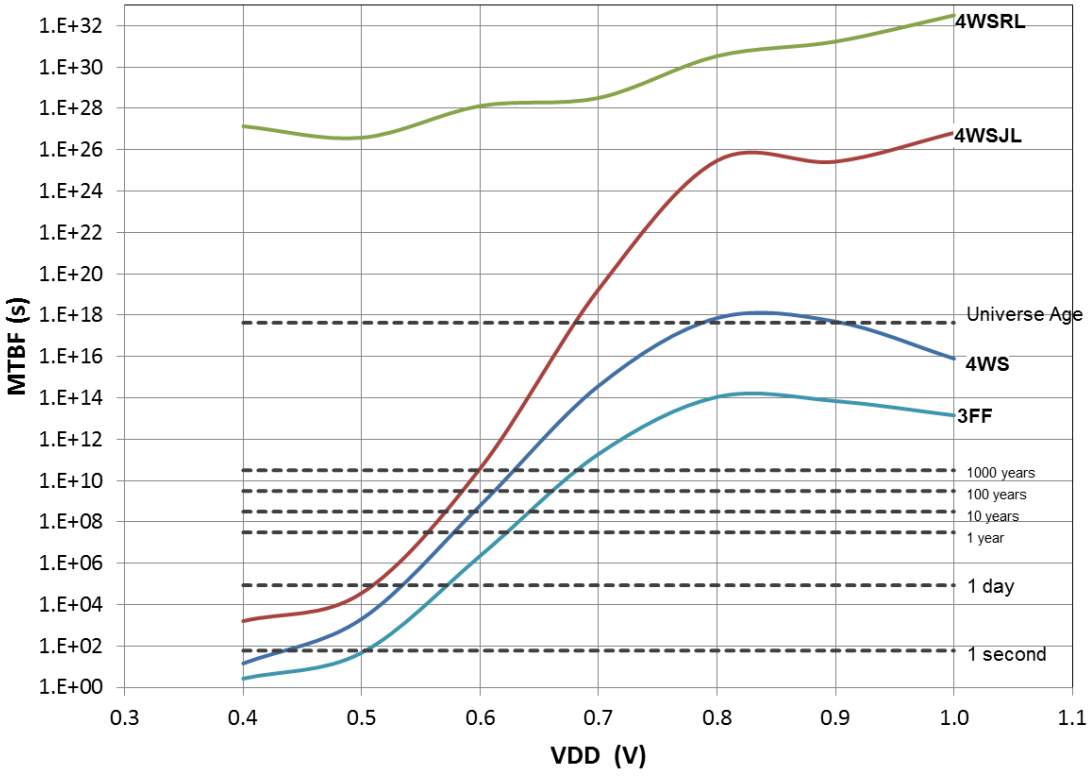
5.2.3 MTBF Synchronizers Results

In this section, one series-pipeline flip-flop synchronizer is compared to three improved parallel-pipeline wagging synchronizers. The former is based on the traditional C2MOSFF, and the latter is constructed of C2MOS latches, Jamb Latches and Robust Latches. The comparison is based on the failure rate at nominal voltage (1V) and low-voltage (0.4V) supply. The comparison results are illustrated in Figure 5.7 and Figure 5.8 below. It shows a set of four graphs, each one representing the MTBF for a number of synchronizing cycles (from two to five). Each graph shows the computed the MTBF, using Equation (2.14), against the supply voltage for the four synchronizers, at the maximum frequency obtained in Table 5.2. The available resolution time was computed using Equation (3.1) for the series-pipeline and Equation (3.2) for the parallel-pipeline. The values of T_w are approximated to the setup time plus hold time. Figure 5.7 and Figure 5.8 show the reliability of the wagging synchronizer is better than that of the flip-flop synchronizer. It also shows that using Jamb Latches in the wagging synchronizer increases the reliability further, whereas the Robust Latches boost the reliability particularly at low V_{DD} in contrast to other schemes. As expected, increasing the number of cycles increases the reliability further, but the wagging synchronizer obtains a greater reliability than the flip-flops with the number of cycles. For instance, an MTBF above 1000 years is possible in the wagging synchronizer operating four paths at supply voltages above 0.7V, five paths at 0.6V, and six paths at 0.5V. It can also operate reliably at 0.4V using six paths with estimated MTBF at about 100 years compared to 1 year using five flip-flops.

The failure rates show greater improvement for using the Jamb Latch and Robust Latch in the wagging structure. A four path wagging synchronizer with Jamb Latches provides an MTBF greater than 1000 years at 0.6V and above, and five paths at 0.5V and six paths at 0.4V. On the other hand, a four-path wagging synchronizer with Robust Latches is suitable for use with a V_{DD} value between 0.4V and 1V. This robust wagging synchronizer boosted the MTBF by at least 10000 times compared with Jamb Latches.

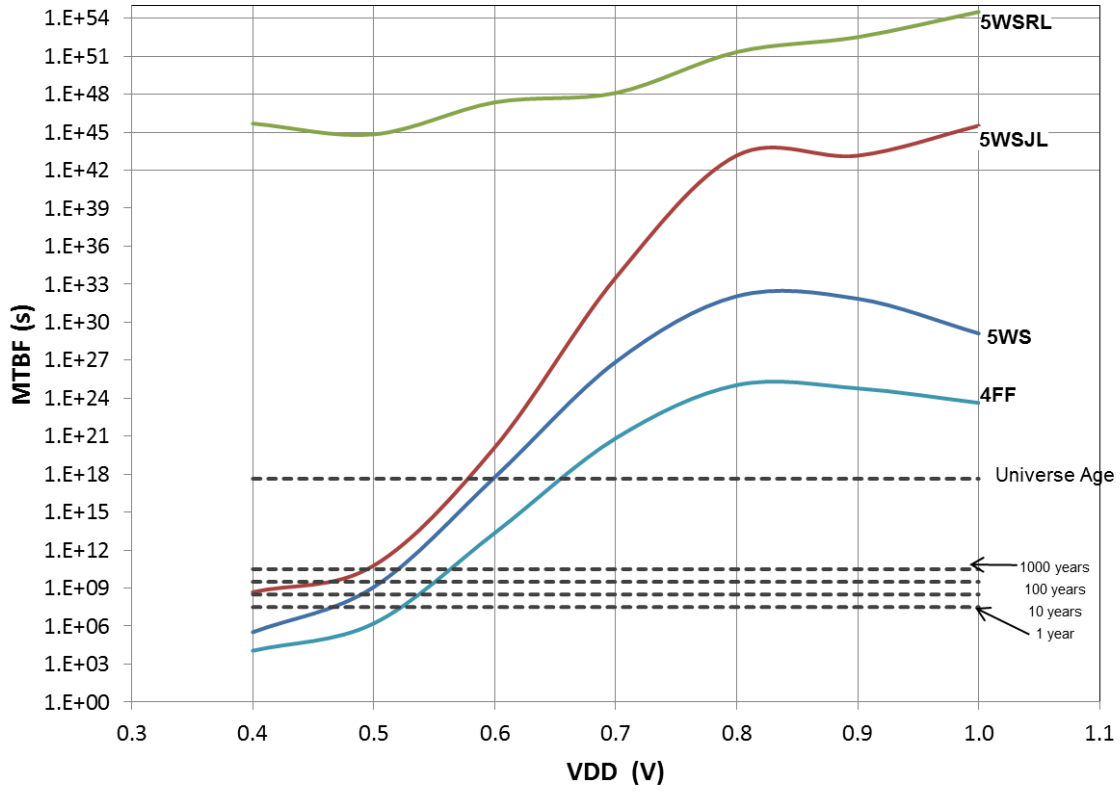


(a) Two cycles

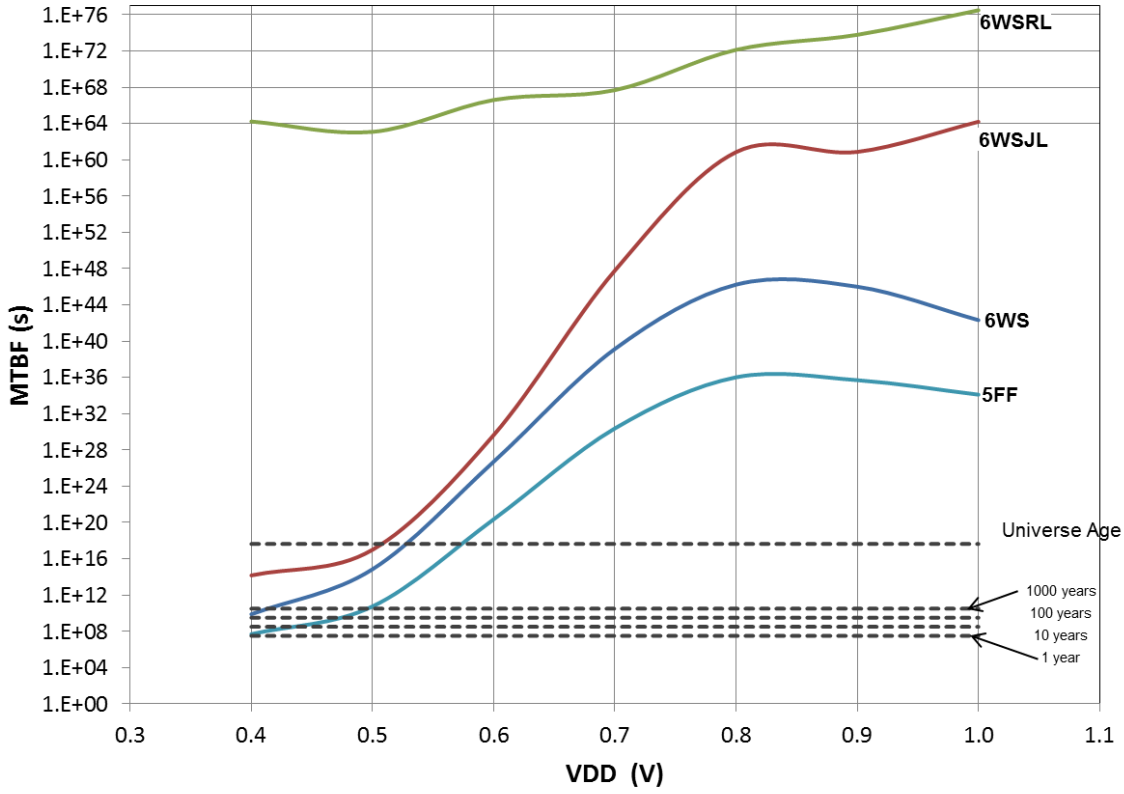


(b) Three cycles

Figure 5.7 MTBF of flip-flop and wagging synchronizers ($f_d = f_{clk}$ = max f_{clk})



(a) Four cycles



(b) Five cycles

Figure 5.8 MTBF of flip-flop and wagging synchronizers ($f_d = f_{clk}$ = max f_{clk})

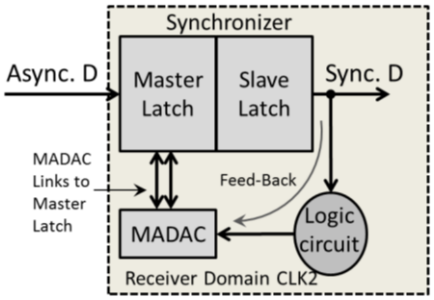
5.3 Flop Synchronizer with Metastability Auto-Detection And Correction

In a typical flip-flop, metastability behavior is dependent on circuit parameters, and it may resolve after some statistically un-deterministic period to either logic level randomly. From this point, a question is raised: is it possible to give the latch a push to one side to help it resolve faster by giving for example a '*hint*' to what value to resolve to? This suggests a method of detecting metastability events and correcting them during less than a clock cycle, this is called Metastability Auto-Detection And Correction (MADAC). This technique is similar to the metastability error detection/correction circuits in [90, 91], reviewed earlier in Chapter 2.

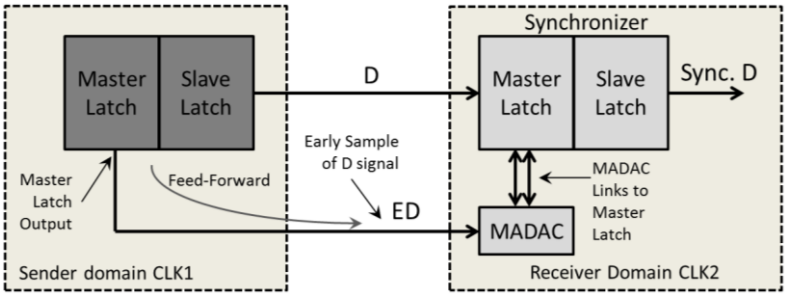
The MADAC technique is implemented on a MSFF with an additional circuit that is able to perform two sequential tasks: detect and then correct metastability. First, the detection circuit senses if the master latch is possibly metastable during and after the triggering edge of the clock. Then, the correction circuit manipulates the conductance of the master latch, based on a reference value, which can be either via a feedback or a feed-forward reference value. The MADAC with a feedback reference relies on a stable value that was stored in the previous clock cycle in one of the following stages, for example the output of the slave latch, as shown in Figure 5.9(a). The MADAC with a feed-forward reference relies on a possible stable new value taken from one of the previous stages, for example the master latch of the sending flip-flop, as shown in Figure 5.9(b). The latter method requires an earlier sample D signal to ensure it is stable and new value, because when metastability is detected D signal might still be in transition, which requires a specialized sending flip-flop with two outputs: one of the master latch, and the other of the slave latch. Based on these requirements, the MADAC method with feedback reference value is preferred, because the one with feed-forward reference value requires two specialized circuits.

As mentioned earlier in Chapter 2, the metastability detection and correction technique may fail, in case metastability naturally resolves in the master latch, while the correction signal is applied. To maintain the operation of the MADAC circuit, there are two main conditions that must be met. First, the length of time required to complete both tasks must be minimized to less than the clock-to-Q delay time so that no oscillation is created, and second, the reference value must be

maintained at a stable value before and during the time for detection and correction, to resolve the metastability condition faster and not worsen the situation.



(a) MADAC with feedback reference value



(b) MADAC with feed-forward reference value

Figure 5.9 MADAC technique

5.3.1 C-element MADAC

A possible implementation of the MADAC technique on MSFF is by using a static C-element followed by a clocked Transmission-Gate (TG) controlled by a slave clock, as shown in Figure 5.10. In this circuit, the master-latch node M0 and the feedback reference value QB are connected to the inputs of the C-element, while the output from the TG drives back to node M0. The location of node M0 in different flip-flops is shown in Figure 5.1. The C-element stores the value of QB, when QB and M0 have the same value, otherwise the C-element stored value is left unchanged. The state value in M0 and the stored value in the C-element should match during the transparent mode of the slave latch. This MADAC circuit operates as follows; at first, the C-element compares the value of the master latch node M0 with the feedback value of the output of the slave latch QB, and stores QB if M0 and QB are equivalent. The comparison is active all of the time, but the correction is only effective during or after the clock sampling transition, so that it will not disturb the master latch. At normal operation, the value at node M0 changes first, the C-elements waits until QB is changed after the sampling edge, then it changes its

stored value. However, if metastability occurs in the master-latch, then, the mid-level voltage (≈½V_{DD}) can be detected at node M0, then the MADAC forces M0 to match the reference value stored in the C-element. The main drawback of the MADAC circuit in Figure 5.10, the previous value stored in the slave latch may start to change shortly after the clock transition and before the correction is completed which may possibly end in a locked state at undefined and unknown level, possibly due to improper clocking. This circuit takes about two to three FO4 delays for the C-element detection time in addition to one FO4 delay for the clocked TG to pass the feedback reference value, resulting in a long response time to correct metastability that is susceptible to oscillation.

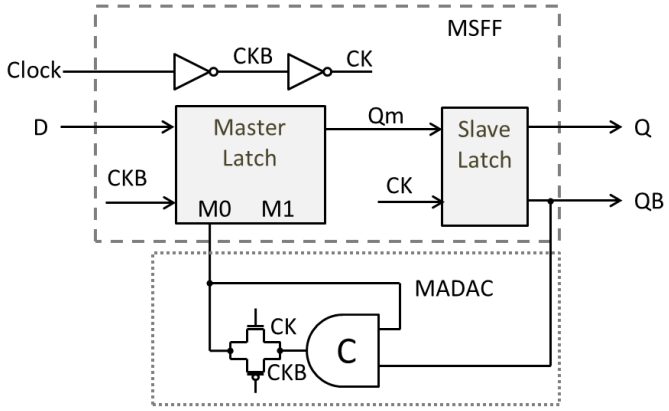


Figure 5.10 C-element plus TG MADAC with feedback

5.3.2 Compound TG MADAC

As an alternative implementation to the C-element, the MADAC concept can be achieved using a Compound TG (CTG) controlled by two signals from the master latch nodes M0 and M1, and linking the output signal QB to the master latch node M0 through the pass gates, as shown in Figure 5.11. In a similar manner, the state of M0 will follow that of QB if the slave latch is transparent. The location of nodes M0 and M1 in different flip-flops is shown in Figure 5.1. The CTG is normally switched off because M0 and M1 are in opposite states, which turns off one transistor in each path of the CTG. One the other hand, the CTG can be switched on if both node (M0 and M1) are at the same value, which can only happen at the metastable level (≈½V_{DD}) during the transition of the clock sampling edge. During normal operation, the CTG circuit is switched off disconnecting both nodes M0 and QB. In case metastability occurs in the master latch and persists for some time longer than the transition time of a transistor plus two TG delays, then the path

between M0 and QB will be opened and a current will be drawn between QB and M0. This current matches the value of M0 node to that of QB and forces metastability in the master latch to be resolved, after that the CTG is switched off.

To stabilize the feedback reference value, an additional slave latch may be inserted between QB and its MADAC input, which samples the previously stored value QB at the rising edge of the clock, as is shown in Figure 5.12. In this way, if metastability persists in the master latch and disturb the value of QB, it will certainly cannot disturbs the reference value QBs.

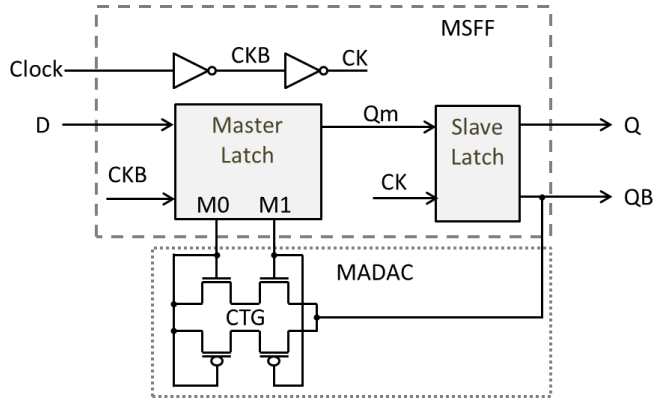


Figure 5.11 CTG MADAC with feedback

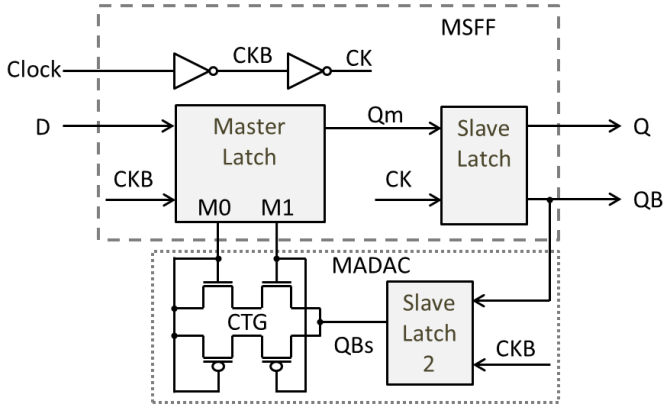


Figure 5.12 CTG MADAC with stable feedback

5.3.3 Robust MADAC Latch (RML)

In CTG MADAC circuits (Figure 5.11 and Figure 5.12), the use of low-V_{TH} transistors in the CTG expands the range of detectable metastable levels and increases the speed of detection and correction; however, at high temperatures, these transistors will have an even lower V_{TH}, which creates a short circuit current through the pass gates. Overall, the detection and correction time using CTG cuts down the excess time required to enhance the reliability of a typical synchronizer

especially at lower voltage levels. The conductance of the pass transistors in this technique suffers at lower supply voltages because their gate-source voltages are at metastable level near half-V_{DD}, that is, lower or near the threshold-voltage, which creates an additional obstacle rather than resolving the situation. To provide a tolerable performance at lower V_{DD} points, the MADAC technique is employed within the Robust synchronizer circuit to provide more current and set or reset one of the nodes (M0 and M1). This is implemented in the circuit shown in Figure 5.13 and called the Robust MADAC Latch (RML), which uses a stable feedback reference value to assist during metastable events. In comparison to the Robust Latch where g_m is boosted during metastability and the metastable point of both sides of the latch are moved together to another point with higher g_m, on the other hand, the RML once it detects metastability, conceals the metastable point and observes its level as logic '1' from one side. This is illustrated in Figure 5.14.

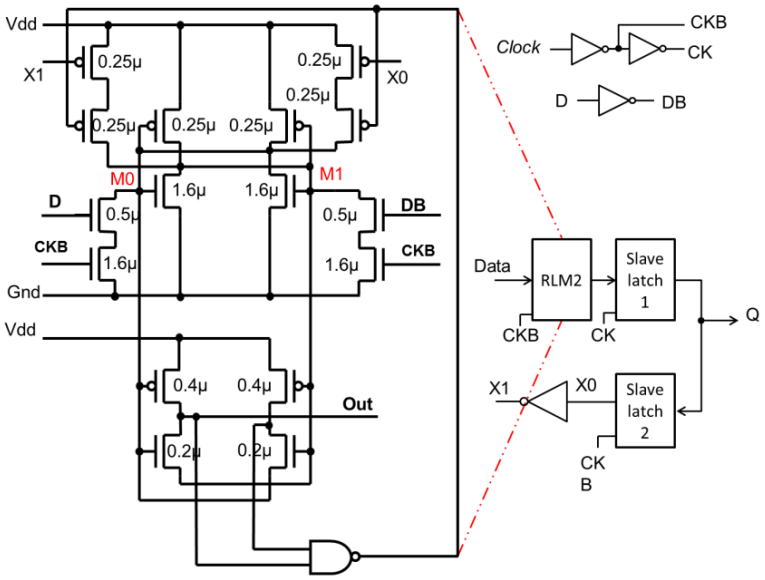
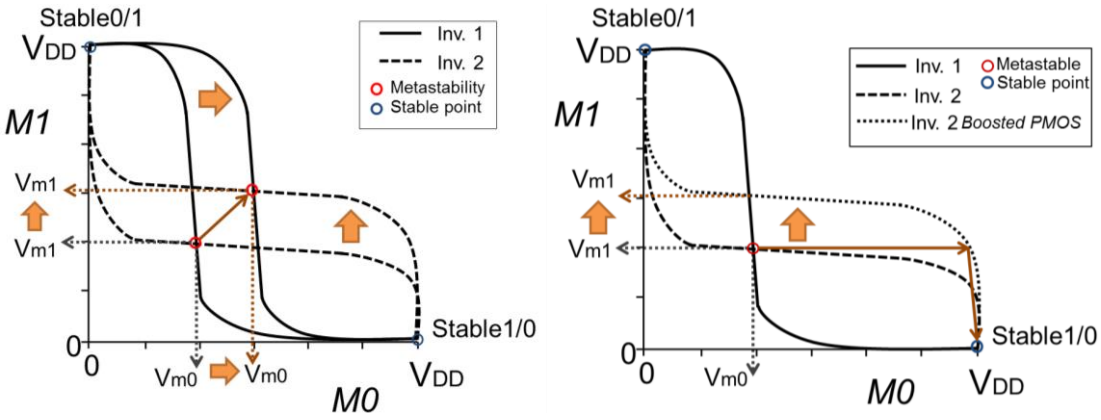


Figure 5.13 Robust MADAC Latch (RML) circuit with feedback reference



(a) in the Robust Latch (b) in the Robust MADAC Latch

Figure 5.14 Movement of the metastable point in RL and RML

5.4 Single-Flop Synchronizers Results

All circuits were modeled in the UMC 90nm process using a minimum channel length of 80nm. The circuit setup is shown in Figure 5.15. The flip-flop and latch circuits in Figure 5.1 were constructed as a master-slave edge-triggered flip-flop arrangement for FF1 and FF2 with and without the MADAC circuitry of Figure 5.12, in addition to the RML in Figure 5.13. In this way, nine flip-flops were under test: namely, TGFF, C2MOSFF (C2MOS), JLFF (JL), RLFF (RL), TGFF with MADAC (TGFF-M), C2MOS with MADAC (C2MOS-M), JL with MADAC (JL-M), RL with MADAC (RL-M), and RML.

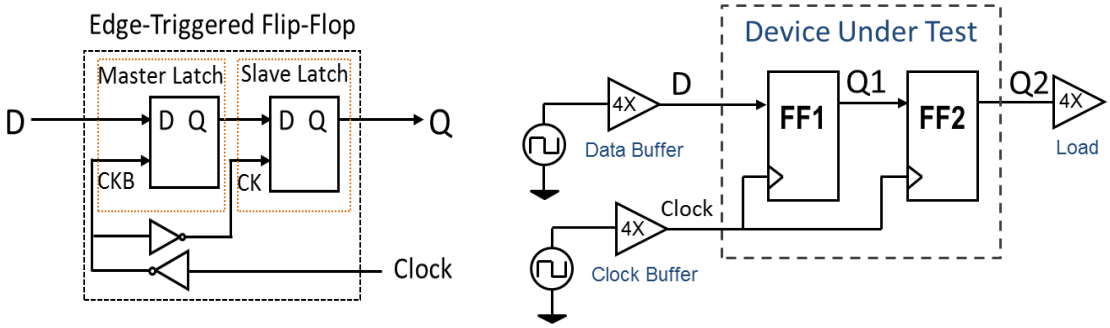


Figure 5.15 Device under test simulation setup

The inputs signals from pulse sources D and Clock are buffered through 4X buffers, and the output Q2 is driving 4X load buffers, as shown in Figure 5.15. By means of a series of *SPICE-level* simulations, the Clock to Q delay time, setup time and hold time were measured as well as the τ time constant of the master latch. The values of τ were simulated using the short circuit method [74]. Power and energy measurements are included for a switching activity of 25%. Voltage supply impact was simulated from 1.2V down to 0.4V with a 100mV step. Process variability simulations were carried out using Monte Carlo statistical analysis under process variations of ±3σ.

5.4.1 Impact of Supply Voltage Reduction

In this section, the impact of supply voltage reduction, on τ , delay time, setup plus hold time, and the power and energy results, was measured and observed for all nine flip-flops: C2MOS, TGFF, JL, RL, C2MOS-M, TGFF-M, JL-M, RL-M, and RML.

Figure 5.16 shows the metastability time constant τ against V_{DD} and the improvement of inserting the MADAC circuit. At a nominal 1V, flip-flops without the MADAC circuit have a value of τ of 15.9ps in the C2MOS, 10ps in the TGFF,

11.36ps in the JL and 9.6ps in the RL. While, the flip-flops with the MADAC circuit have a value of τ of 5.55ps in the C2MOS, 6.9ps in the TGFF, 4.9ps in the JL, 4.98ps in the RL and 3.3ps in the RML. In this way, at 1V, the MADAC circuit provided a lower τ by 30% in the TGFF, 48% in the RL, 58% in the JL and 65% in the C2MOS circuit; in addition, the RML has lower τ by 65% compared to the RL. The benefit of the MADAC technique is only limited to supply voltages at and above 0.7V. For instance, at 0.6V, the MADAC circuit reduced τ by 3% in the TGFF, 5% in the RL, 12% in the C2MOS and 23% in the JL circuit. However, at 0.5V and 0.4V, the MADAC circuit is inefficient to improve τ in the RL, the C2MOS, and the TGFF circuits. This is because the pass transistors, in the CTG circuit, are not switching 'ON' as a result of the metastable level is near or below their threshold voltage. On the other hand, the RML has lower value of τ compared to others, by less than 65% at nominal 1V, by 56% at 0.6V and by 50% at 0.4V compared to the RL.

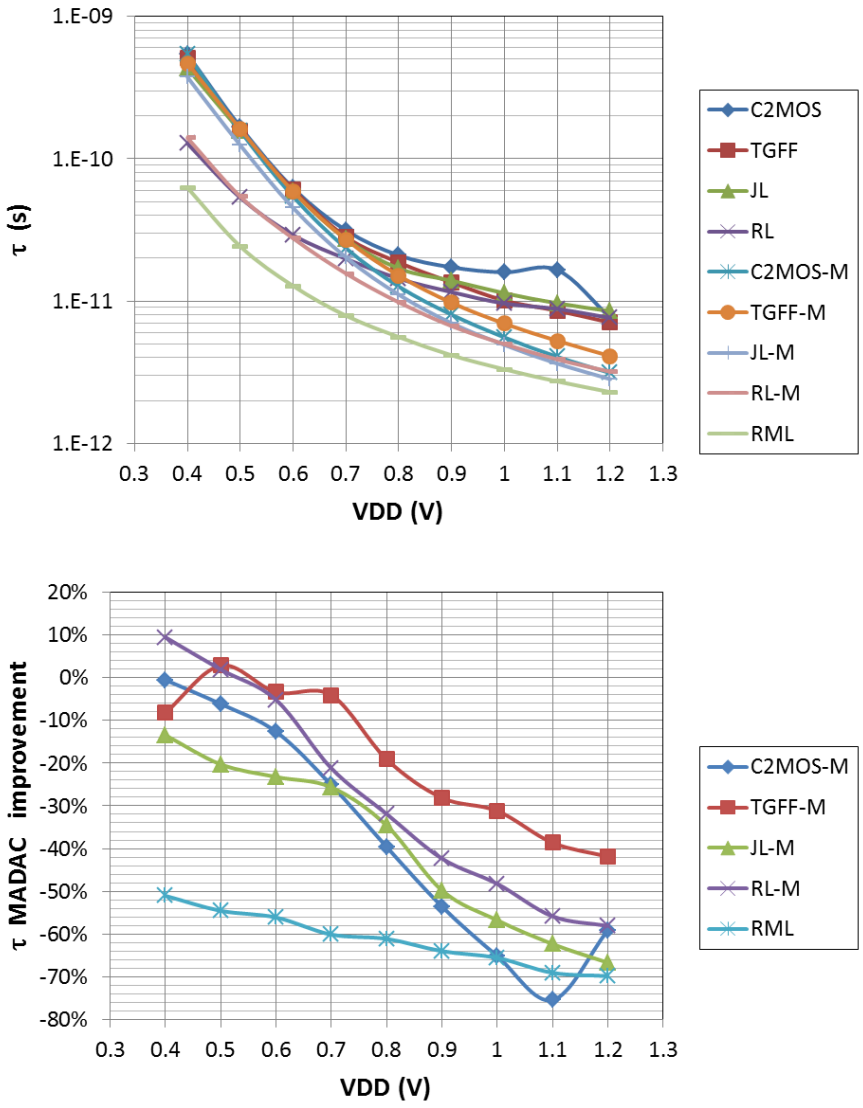


Figure 5.16 Metastability time constant of flip-flops without and with MADAC

Figure 5.17 and Figure 5.18 shows the impact of supply voltage reduction on the D to Q delay time and setup plus hold 'window' time for all nine flip-flops, in addition to the impact of MADAC insertion. At nominal 1V, the addition of the MADAC circuit increases the delay time, by 9% in the C2MOS, 10% in the JL, 21% in the TGFF, and 28% in the RL, and setup plus hold time, by 78% in the C2MOS, 31% in the JL and 43% in the RL. Although the MADAC circuit insertion increases the delay and the setup hold window, the reduction of τ dominates because of its exponential effect on the failure rate in a synchronizer circuit. In comparison to the RL, the RML has a minimum increase in delay and window time. The TGFF has the fastest D to Q time and smallest window time, because it uses transmission gates to open and close its latches.

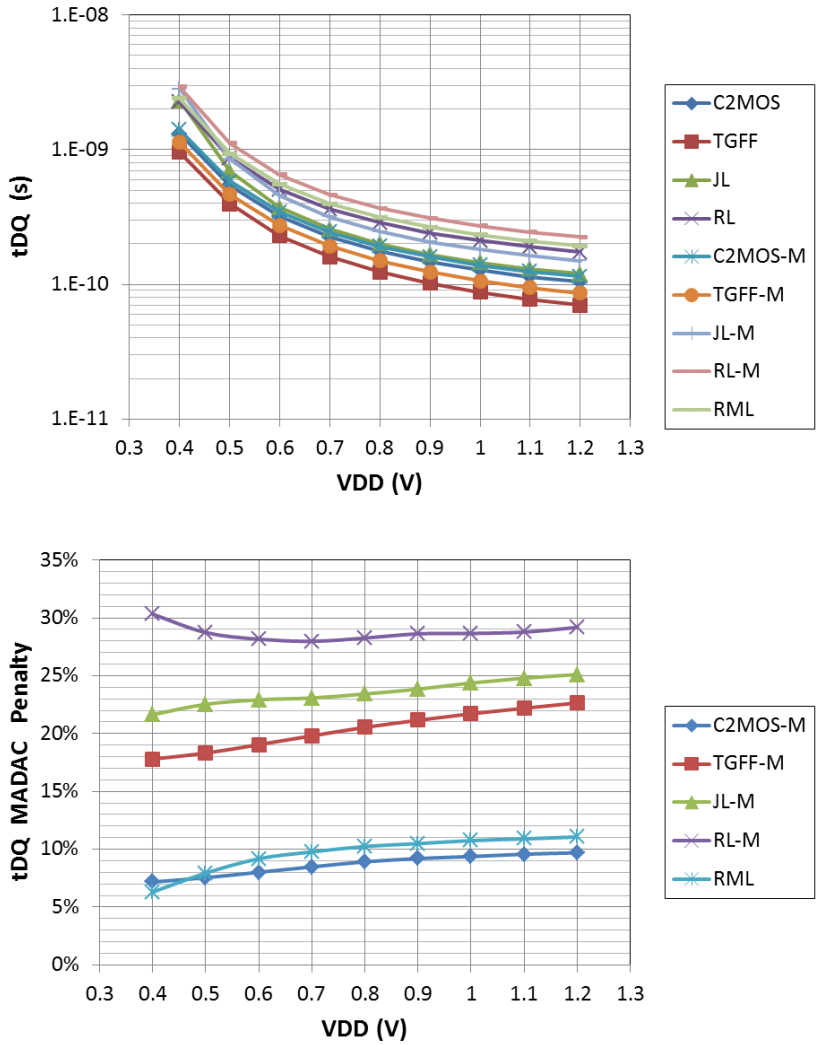


Figure 5.17 Data to Q delay time of flip-flops without and with MADAC

Figure 5.19 shows the impact of supply voltage reduction on power and energy measurements for all nine flip-flops. The typical TGFF consumes less energy

compared to others, whereas the Robust Latch with MADAC consumes more energy. The minimum energy point is located between 0.6V and 0.8V. Overall, flip-flops with MADAC consume more power than those without due to the additional circuitry including the CTG and the second slave latch. Since the required number of synchronizer circuits is much less than that of flip-flops and latches used in memory, the increase in power requirement is acceptable as long as the failure rate is maintained at a minimum over the range of operation.

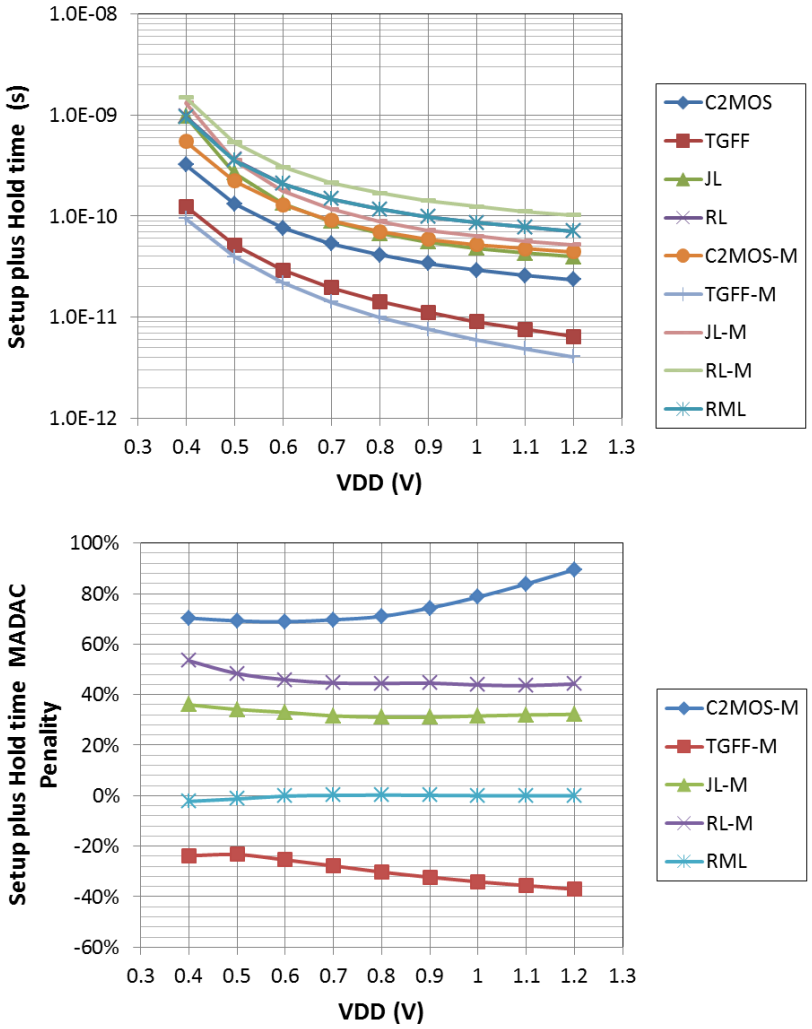


Figure 5.18 Setup and Hold 'window' time of flip-flops without and with MADAC

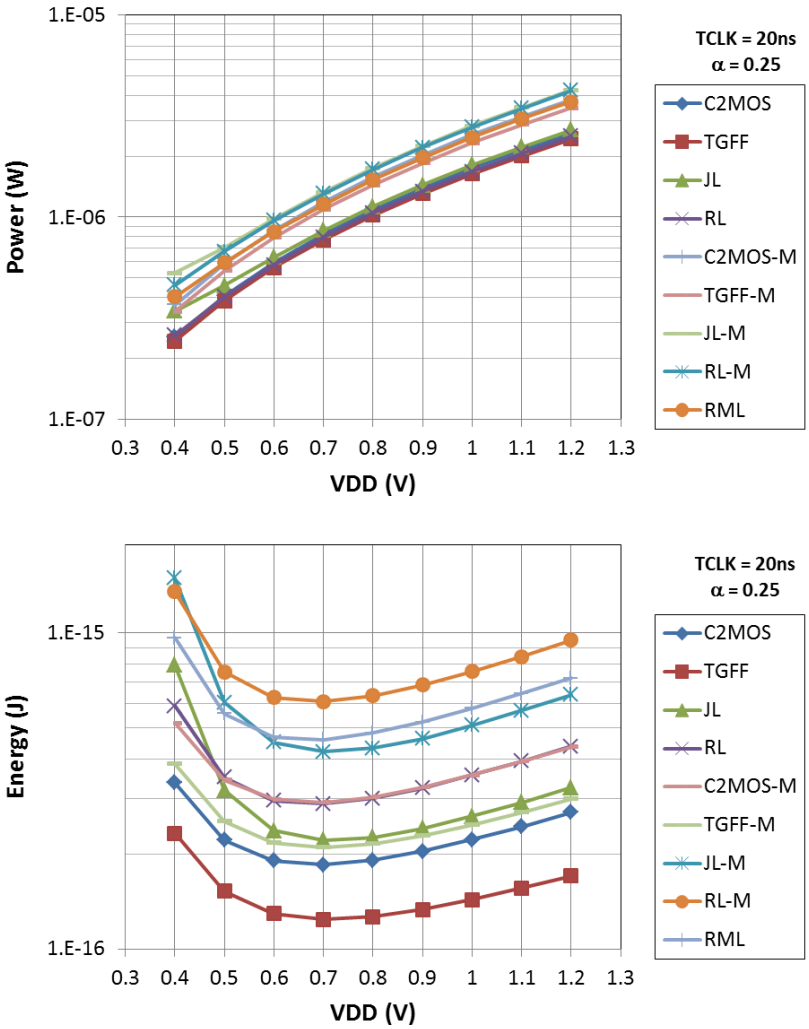
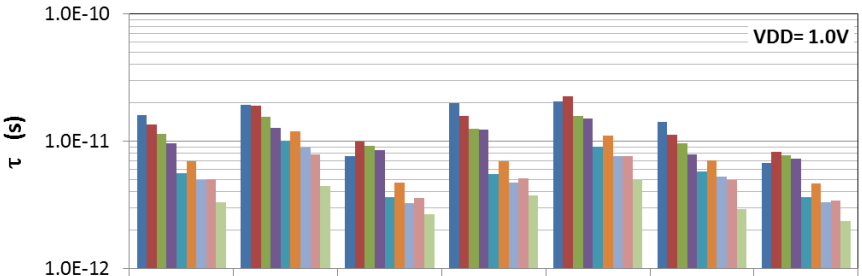


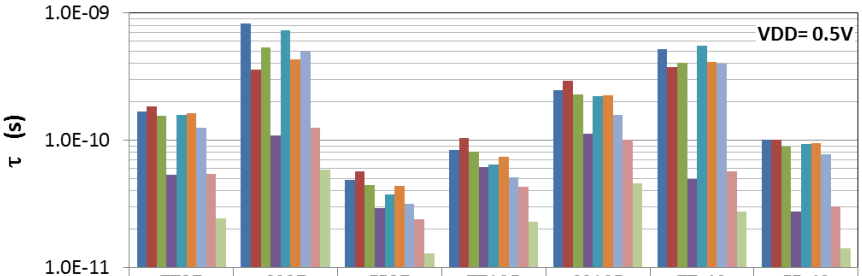
Figure 5.19 Power and energy measurements

5.4.2 PVT Corners

In a similar manner, the nine flip-flops were characterized for a number of process corners and temperatures. The impact on τ is shown in Figure 5.20. The worst corner at 1V is SS125 and at 0.5V is SS27. Flip-flops with MADAC have τ value that can tolerate all process and temperature corners at nominal voltage of 1V, but they are ineffective at 0.5V, because the CTG transistors are not switching 'ON' as the metastable level is near or below their threshold voltage. On the other hand, the RML shows significant improvement in the value of τ compared to the other circuits at 1V and 0.5V over all corners.



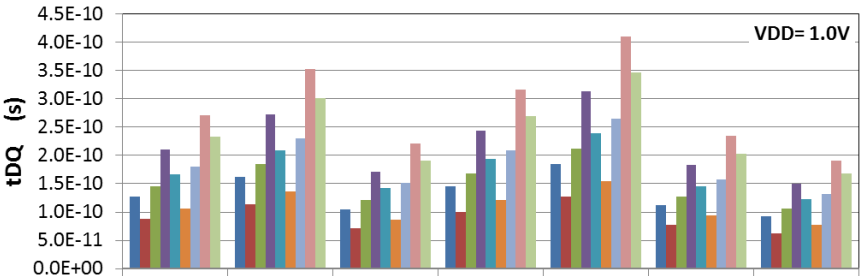
	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
C2MOS	1.59E-11	1.93E-11	7.67E-12	1.98E-11	2.05E-11	1.42E-11	6.73E-12
TGFF	1.35E-11	1.89E-11	9.87E-12	1.57E-11	2.26E-11	1.12E-11	8.28E-12
JL	1.14E-11	1.55E-11	9.15E-12	1.26E-11	1.57E-11	9.65E-12	7.69E-12
RL	9.62E-12	1.27E-11	8.44E-12	1.23E-11	1.51E-11	7.82E-12	7.22E-12
C2MOS-M	5.55E-12	1.00E-11	3.63E-12	5.47E-12	9.05E-12	5.75E-12	3.63E-12
TGFF-M	6.93E-12	1.19E-11	4.71E-12	6.98E-12	1.11E-11	7.03E-12	4.64E-12
JL-M	4.91E-12	8.92E-12	3.23E-12	4.71E-12	7.64E-12	5.22E-12	3.30E-12
RL-M	4.98E-12	7.90E-12	3.56E-12	5.08E-12	7.64E-12	4.96E-12	3.43E-12
RML	3.31E-12	4.45E-12	2.66E-12	3.75E-12	5.03E-12	2.90E-12	2.33E-12



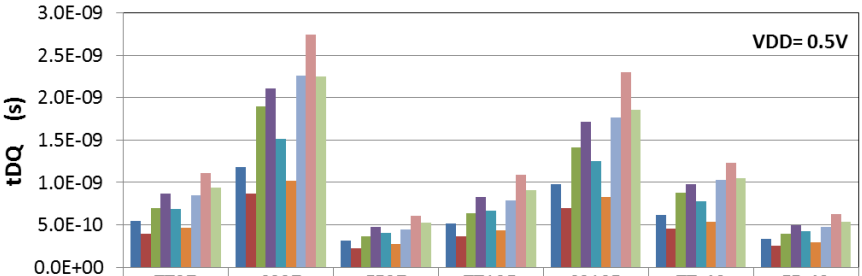
	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
C2MOS	1.67E-10	8.23E-10	4.86E-11	8.34E-11	2.48E-10	5.20E-10	1.01E-10
TGFF	1.85E-10	3.58E-10	5.64E-11	1.04E-10	2.94E-10	3.74E-10	1.01E-10
JL	1.56E-10	5.34E-10	4.40E-11	8.12E-11	2.29E-10	4.08E-10	8.87E-11
RL	5.32E-11	1.08E-10	2.91E-11	6.15E-11	1.11E-10	4.91E-11	2.74E-11
C2MOS-M	1.57E-10	7.34E-10	3.76E-11	6.44E-11	2.21E-10	5.54E-10	9.25E-11
TGFF-M	1.62E-10	4.29E-10	4.33E-11	7.37E-11	2.24E-10	4.13E-10	9.49E-11
JL-M	1.24E-10	5.05E-10	3.17E-11	5.12E-11	1.58E-10	4.00E-10	7.75E-11
RL-M	5.43E-11	1.26E-10	2.40E-11	4.29E-11	1.00E-10	5.66E-11	2.99E-11
RML	2.42E-11	5.84E-11	1.29E-11	2.28E-11	4.60E-11	2.74E-11	1.40E-11

Figure 5.20 τ response to PVT corners

The delay and window times, shown in Figure 5.21 and Figure 5.22, are worse with the MADAC circuit because the addition of the loading capacitances to the nodes, which corresponds to an increase in delay time and increase in sensitivity to process corners.

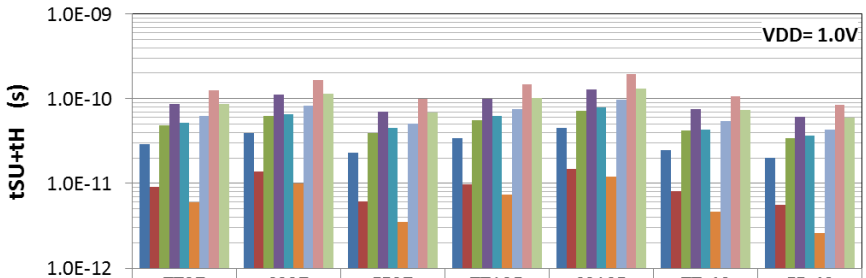


	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
C2MOS	1.27E-10	1.62E-10	1.05E-10	1.46E-10	1.84E-10	1.12E-10	9.21E-11
TGFF	8.74E-11	1.13E-10	7.09E-11	9.96E-11	1.28E-10	7.74E-11	6.28E-11
JL	1.45E-10	1.85E-10	1.21E-10	1.67E-10	2.12E-10	1.28E-10	1.06E-10
RL	2.11E-10	2.73E-10	1.72E-10	2.43E-10	3.14E-10	1.84E-10	1.51E-10
C2MOS-M	1.67E-10	2.09E-10	1.42E-10	1.94E-10	2.39E-10	1.45E-10	1.23E-10
TGFF-M	1.06E-10	1.37E-10	8.71E-11	1.22E-10	1.55E-10	9.39E-11	7.68E-11
JL-M	1.81E-10	2.30E-10	1.51E-10	2.10E-10	2.64E-10	1.58E-10	1.32E-10
RL-M	2.71E-10	3.53E-10	2.21E-10	3.16E-10	4.09E-10	2.35E-10	1.91E-10
RML	2.33E-10	3.01E-10	1.91E-10	2.70E-10	3.47E-10	2.04E-10	1.68E-10

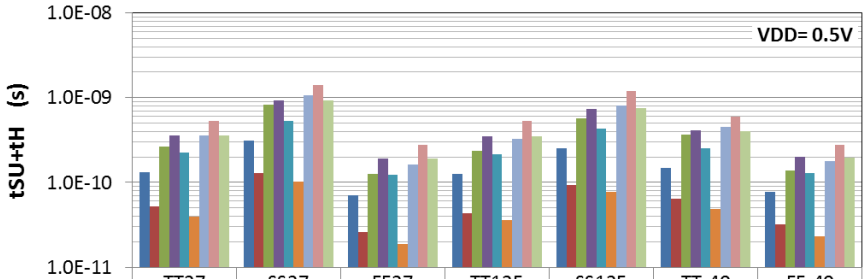


	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
C2MOS	5.42E-10	1.18E-09	3.13E-10	5.15E-10	9.76E-10	6.12E-10	3.37E-10
TGFF	3.95E-10	8.71E-10	2.26E-10	3.66E-10	6.97E-10	4.55E-10	2.49E-10
JL	6.94E-10	1.90E-09	3.61E-10	6.32E-10	1.41E-09	8.75E-10	3.93E-10
RL	8.66E-10	2.11E-09	4.77E-10	8.32E-10	1.72E-09	9.79E-10	4.98E-10
C2MOS-M	6.91E-10	1.51E-09	4.02E-10	6.63E-10	1.25E-09	7.77E-10	4.28E-10
TGFF-M	4.67E-10	1.02E-09	2.70E-10	4.38E-10	8.26E-10	5.35E-10	2.95E-10
JL-M	8.50E-10	2.26E-09	4.42E-10	7.88E-10	1.76E-09	1.03E-09	4.76E-10
RL-M	1.11E-09	2.74E-09	6.08E-10	1.09E-09	2.31E-09	1.23E-09	6.24E-10
RML	9.34E-10	2.25E-09	5.21E-10	9.05E-10	1.85E-09	1.05E-09	5.40E-10

Figure 5.21 D to Q delay time response to PVT corners



	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
C2MOS	2.92E-11	3.90E-11	2.31E-11	3.39E-11	4.50E-11	2.50E-11	1.99E-11
TGFF	9.02E-12	1.40E-11	6.11E-12	9.72E-12	1.50E-11	8.09E-12	5.56E-12
JL	4.80E-11	6.25E-11	3.94E-11	5.60E-11	7.23E-11	4.21E-11	3.43E-11
RL	8.64E-11	1.12E-10	6.98E-11	9.98E-11	1.30E-10	7.51E-11	6.11E-11
C2MOS-M	5.22E-11	6.62E-11	4.51E-11	6.27E-11	7.82E-11	4.33E-11	3.71E-11
TGFF-M	5.94E-12	1.00E-11	3.52E-12	7.31E-12	1.21E-11	4.62E-12	2.63E-12
JL-M	6.32E-11	8.31E-11	5.13E-11	7.57E-11	9.84E-11	5.41E-11	4.35E-11
RL-M	1.24E-10	1.65E-10	9.92E-11	1.48E-10	1.95E-10	1.06E-10	8.40E-11
RML	8.64E-11	1.14E-10	6.95E-11	1.01E-10	1.33E-10	7.44E-11	6.01E-11



	TT27	SS27	FF27	TT125	SS125	TT-40	FF-40
C2MOS	1.32E-10	3.14E-10	7.07E-11	1.25E-10	2.53E-10	1.50E-10	7.68E-11
TGFF	5.19E-11	1.29E-10	2.59E-11	4.37E-11	9.31E-11	6.42E-11	3.23E-11
JL	2.67E-10	8.36E-10	1.26E-10	2.36E-10	5.72E-10	3.68E-10	1.39E-10
RL	3.63E-10	9.29E-10	1.93E-10	3.46E-10	7.41E-10	4.16E-10	2.01E-10
C2MOS-M	2.23E-10	5.33E-10	1.22E-10	2.16E-10	4.31E-10	2.53E-10	1.30E-10
TGFF-M	3.98E-11	1.03E-10	1.89E-11	3.57E-11	7.69E-11	4.85E-11	2.29E-11
JL-M	3.58E-10	1.07E-09	1.64E-10	3.29E-10	8.11E-10	4.56E-10	1.78E-10
RL-M	5.38E-10	1.42E-09	2.76E-10	5.38E-10	1.20E-09	5.92E-10	2.78E-10
RML	3.58E-10	9.21E-10	1.90E-10	3.49E-10	7.49E-10	4.03E-10	1.94E-10

Figure 5.22 Setup plus hold 'window' time response to PVT corners

5.4.3 Process Variations Response

Based on Monte Carlo simulation statistical results for all nine flip-flops are shown in Appendix E, the values of the mean and standard deviation for the values of τ are presented below in Figure 5.23. The mean value is similar to that obtain before in Figure 5.16. The addition the MADAC circuit in flip-flops slightly reduces the standard deviation of τ. On the other hand, Figure 5.24 shows significant reduction of mean and standard deviation of the RML circuit in comparison to the other eight circuits. It also shows that the standard deviation of τ in the RML circuit is significantly lower than that of the other circuits, by 20% and up to 50% compared to the RL-M and by 70% and up to 90% compared to the other circuits.

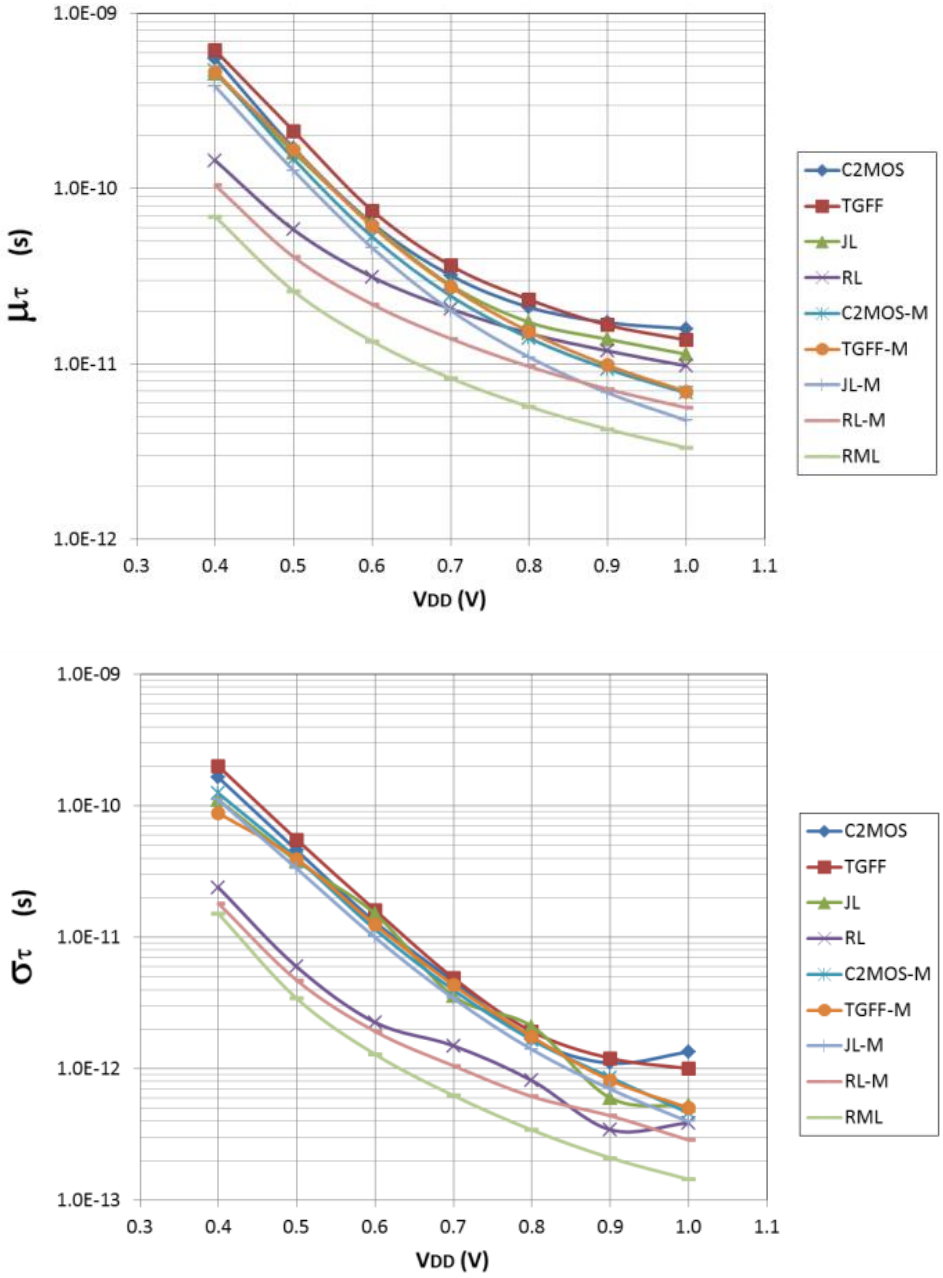


Figure 5.23 Mean and standard-deviation of τ

The standard-deviation of τ for RL is changing against reduced supply voltage differently from that of the other circuits without MADAC, as shown in Figure 5.23. Referring to histograms of τ for RL presented in Figure E.4, at 1V τ varies at σ of ±0.388ps around the mean of 9.75ps, whereas at 0.9V it varies at ±0.343ps around 11.86ps, this because RL uses small transistors (see Figure 5.1(d)) with ±3σ process variation simulation tend to cause more impact. The reduction of standard-deviation of τ between 1V and 0.9V is considered very small, less than 0.05ps. At voltages below 0.9V, the g_m booster is more effective at reducing the

spread of variation around the mean. On the other hand, in Figure 5.24, the σ of τ for the RML improvement over that for the RL shows that the percentage values are not changing linearly which may assume to be an anomaly, however, this is because the spread of τ variations around its mean for the RML is much narrower compared to the other circuits, as shown in Figure 5.23 and Figure E.5, therefore a small change of the σ_τ for RL between 1V and 0.9V appears very large difference when compared to that for RML.

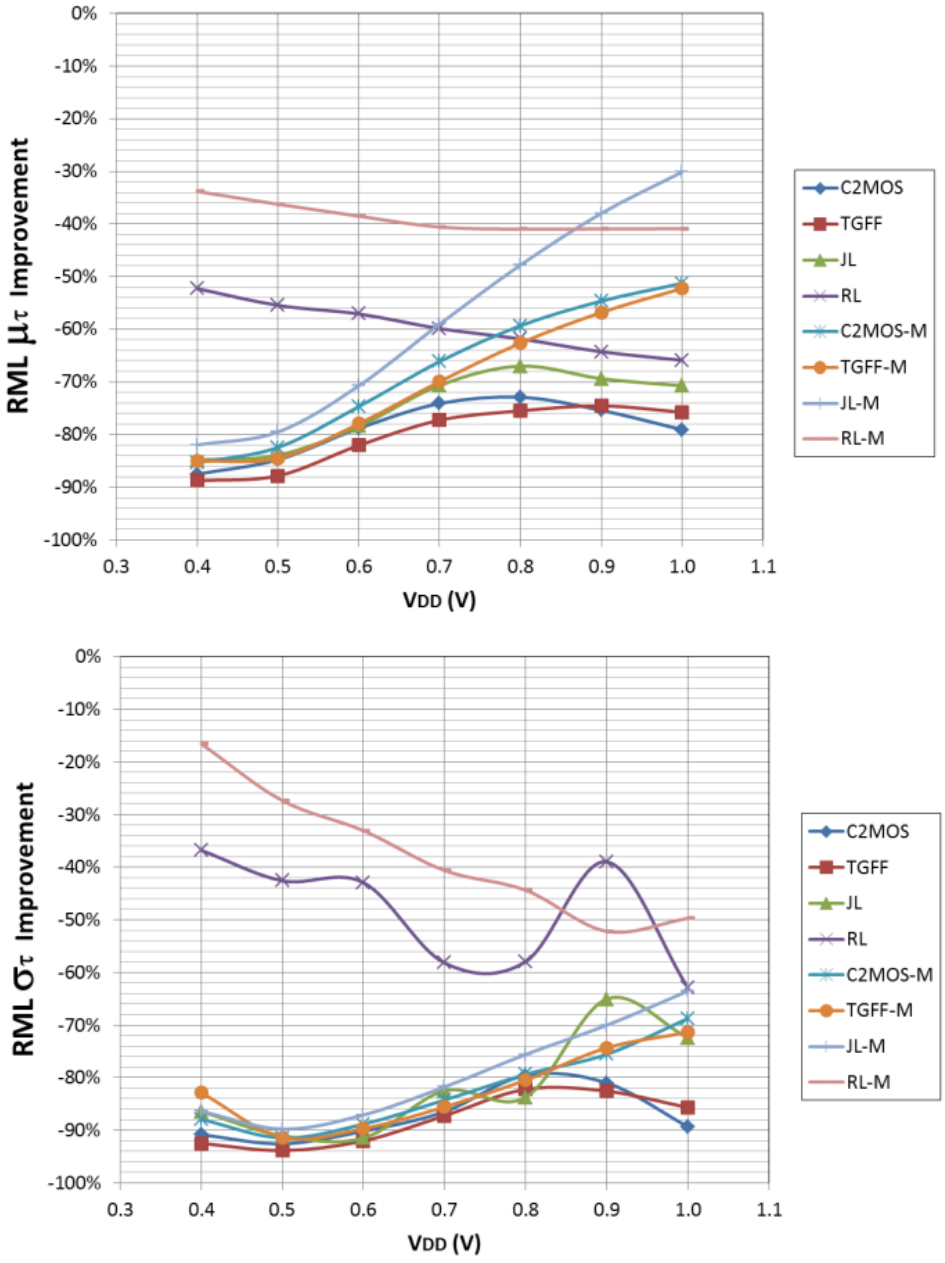


Figure 5.24 Robust MADAC Latch Improvement of τ variability compared to others

Figure 5.25 and Figure 5.26 shows the mean and standard-deviation of D to Q time and window time for all nine flip-flops. Their mean values are similar to their nominal values observed earlier in Figure 5.17 and Figure 5.18.

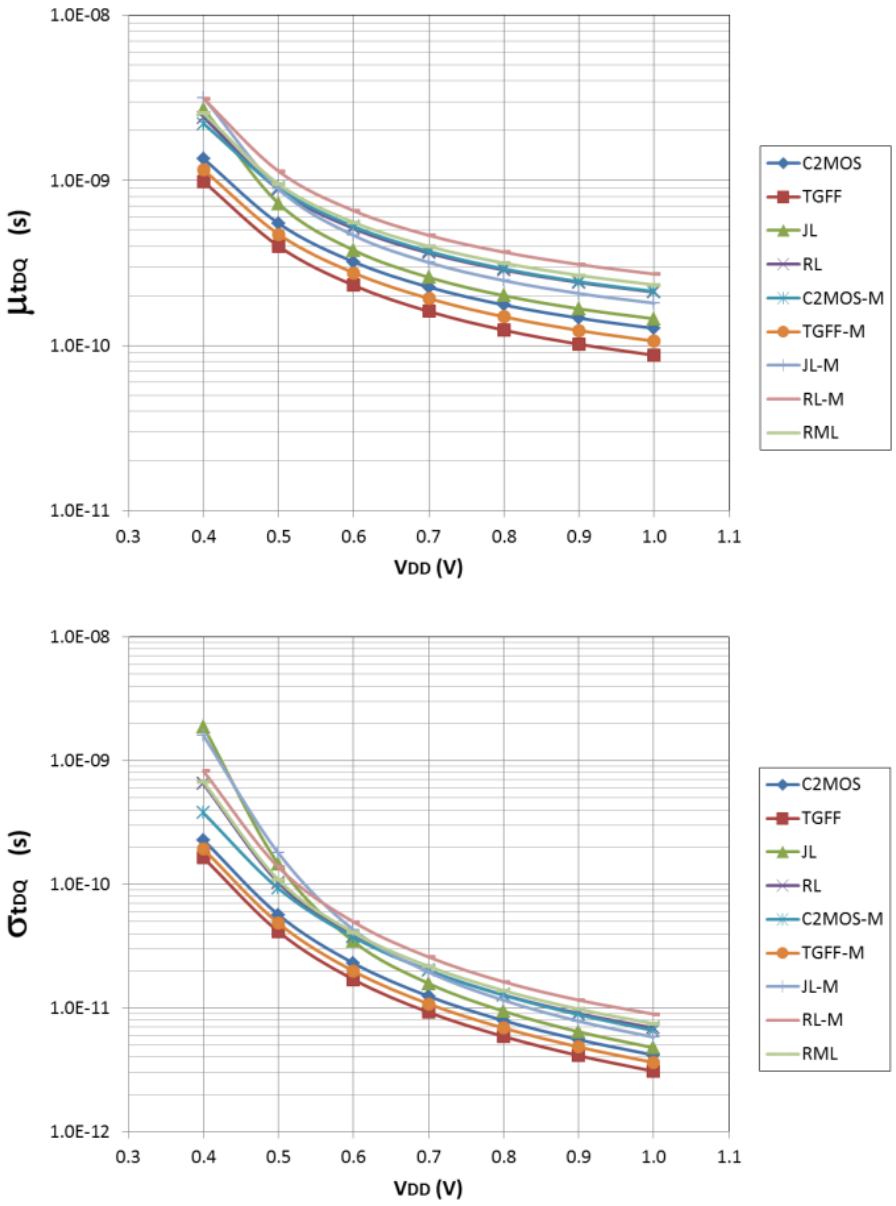


Figure 5.25 Mean and standard-deviation of D to Q time

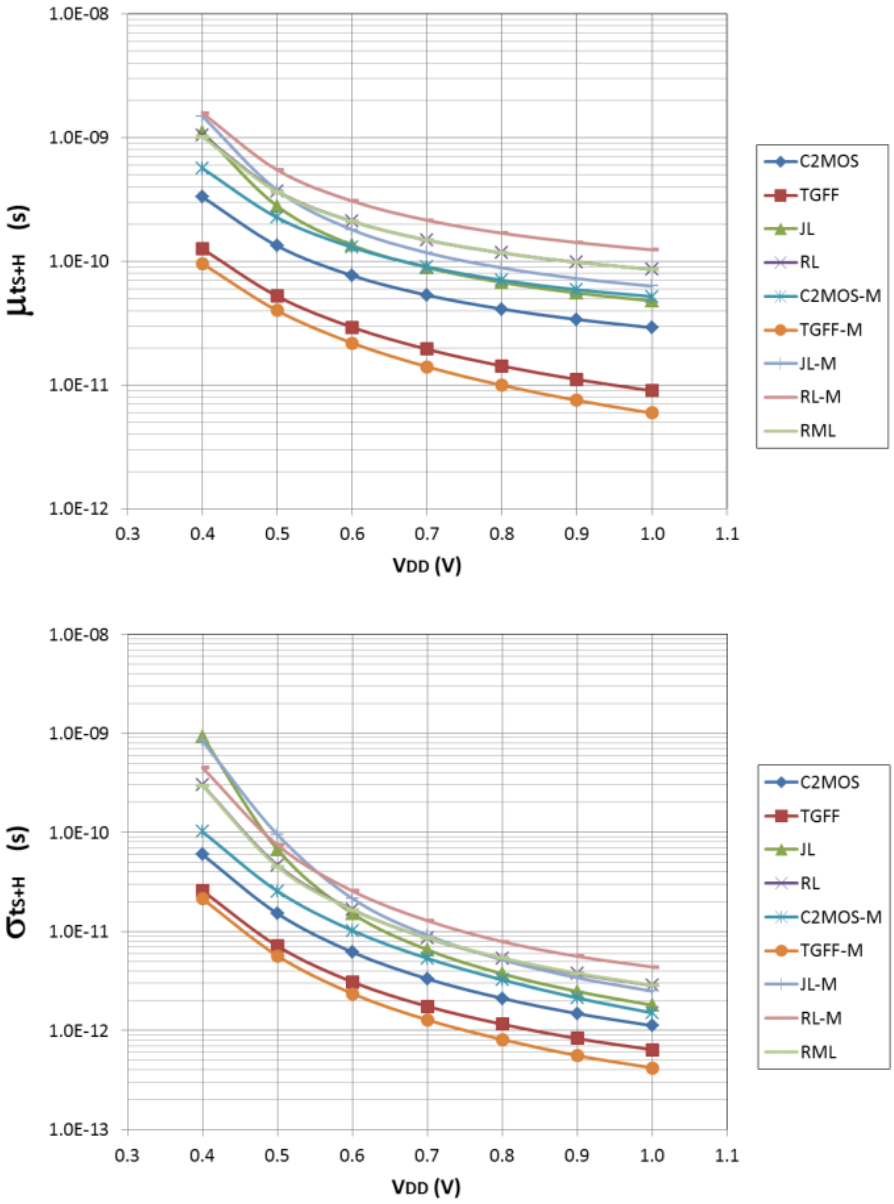


Figure 5.26 Mean and standard-deviation of setup plus hold times

5.5 Summary

In this chapter, two techniques were presented to lower failures due to metastability in synchronizer circuits. The first method is based on the wagging structure with improved τ latches and reconfigurable clock cycles. The results showed the advantages of the wagging structure against the conventional flip-flop synchronizer in terms of higher reliability for the same number of synchronizing clock cycles. The robustness of the wagging technique can be enhanced by using Robust Latches or adding one more cycle of synchronization. The use of Robust Latches improves the reliability considerably, particularly at low supply voltages.

The main limitation of the wagging synchronizer is that it requires a clocking control circuit that may reduce the maximum frequency of the system master clock.

The second method introduced the MADAC approach to detect and correct metastability. The insertion of the CTG MADAC circuit the flip-flop circuits in Figure 5.1 has lowered τ by at least 30% at nominal voltage supply, but only a small improvement at low V_{DD}. This was improved by the Robust MADAC Latch, which reduced τ between 52% and 70% compared to that of the Robust Latch for supply voltages ranging between 0.4V and 1.2V. However, the delay and window times increased by the addition of the MADAC circuit, because of the increased capacitive loading on both the master and the slave latches. They also require more energy than standard circuits. In general, synchronizer circuits are dominated by the value of τ, unlike memory cells and registers that are governed by the delay time; furthermore the number of synchronizers used in a system is much smaller than the number of registers in a system or the number of cells in an embedded memory block. This implies to that the above penalties in timing and energy can be disregarded provided the failure rate is minimized.

Chapter 6 Multiple Voltage Domain Synchronizers

The main objective of multiple voltage design is to reduce the overall power consumption by providing different voltage domains that are either constant or variable [55]. Sometimes, the reduction of voltage supply requires a reduced clock frequency as in the case of Dynamic Voltage and Frequency Scaling (DVFS), which creates a Multiple-Clock Domain (MCD) challenge. Hence, there is a need to study the design of synchronizers placed between multiple voltage domains.

The typical case for level-shifting is to place a dual-supply level-shifter before the input of each synchronizer. This method requires two power lines: one from the sending domain and the other from the receiving end [4, 55, 106]. This requirement becomes a critical problem if the receiving block takes signals from multiple senders with different voltage supplies, which results in power supply connection congestion. Instead, single-supply level-shifters, such as [109, 110], may be used before each synchronizer, which reduces the power routing. However, their main drawback is the excess leakage current through the PMOS path during upshifting an input value of a logic '1' because the input driven PMOS transistor is not fully turned off. One level-shifter per line is required between two domains where at least one voltage supply is fixed; otherwise, two level-shifters are necessary to avoid problems concerning the conversion correctness and accuracy of the signals [111].

In order for signals crossing domain boundaries to be recognized correctly at the destination, voltage re-leveling is required in an MVD and retiming is required in a MCD. In an MVD that scales frequency with voltage, either at predictable points as in a DVFS or unpredictably as in an adaptive-voltage scaling design [55], retiming is as important as level-shifting because signal timing will certainly vary with voltage. Voltage scaling goes together with clock frequency scaling and the boundaries of a clock domain would be the boundaries for voltage scaling.

A synchronous interface between a scaled-voltage-frequency domain and the rest of the system is incapable of operating efficiently as the voltage and frequency are varied, because the clock tree delays and skew will vary too. On the other hand, asynchronous interfaces with synchronizers will resolve the wide variation in frequency and voltage. Figure 6.1 shows an example of a handshake synchronization and level-shifter interface between two variable voltage/clock domains separated by an intermediate voltage-domain as recommended by [111].

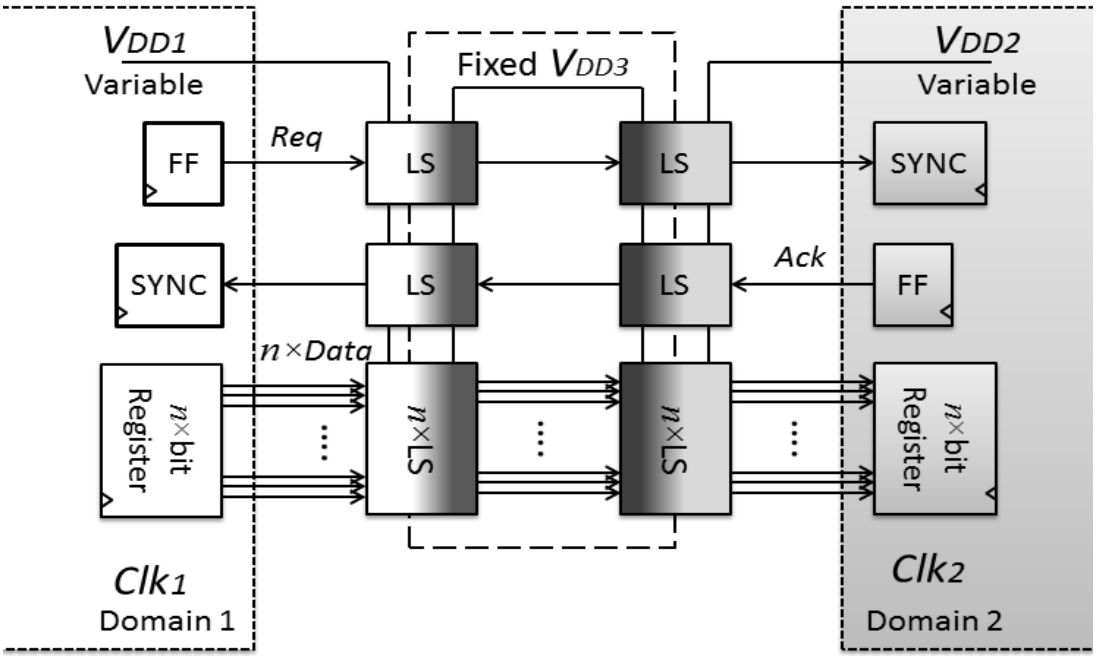


Figure 6.1 Handshake between variable MVD/MCD domains

An alternative approach uses Level-Shifting Flip-Flops (LSFF) as a synchronizer. Numerous dual-supply LSFF designs appear in literature, such as [4, 101, 106, 122, 123], but there is no previous work on single-supply LSFFs. This is because the design of a single-supply LSFF synchronizer may not be feasible within a single cell. For example, a single-supply level-shifter described in [109] may be transformed into a static latch, however because level-shifting in this circuit is based on weak feedback transistors, it will naturally exhibit poor metastability behavior [50].

This chapter proposes new interface circuit techniques to transfer signals between multiple voltages multiple clock domains that do not require the addition of conventional level-shifters or dual-supply connections. The proposed circuits provide level-shifting and synchronization between signals over a wide range of voltage-supplies and clock frequencies.

6.1 Latch Level-Shifter

6.1.1 Dual-Supply Level-Shifting Latch

Typical level-shifters are based on differential inputs with a bistable feedback latch to pull one of its nodes down to zero. The Jamb Latch circuit with differential inputs, shown in Figure 6.2, is a level-shifting circuit, and if the latch τ is good enough, it is possible to use it as a level-shifting synchronizer latch as the master-latch in a master-slave flip-flop configuration.

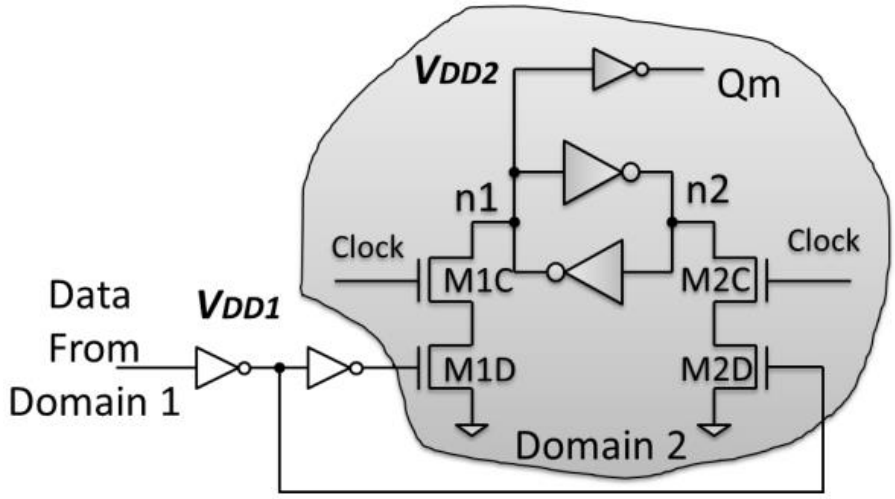


Figure 6.2 Jamb Latch with differential input as a dual-supply level-shifter

The main difference between the typical Jamb Latch and the one proposed as a level-shifter is that the data and clock inputs are swapped. In comparison to the typical Jamb Latch, the data input drives the gate-terminal of the NMOS transistor whose source-terminal is connected to the ground node, because it will have more gain when upshifting from low voltage domains due to the cascoded NMOS transistor driven by clock. The result of that swap, shown in Figure 6.3, is that the level-shifting Jamb Latch can upshift from 0.7V to 1.1V, whereas the typical one is incapable at doing that. In this way, the level-shifters becomes redundant in synchronizer circuits because their function is undertaken by the master-level of the synchronizer circuit, and as a result power consumption and area overhead, as well as the propagation delay time between the two-domains, are reduced.

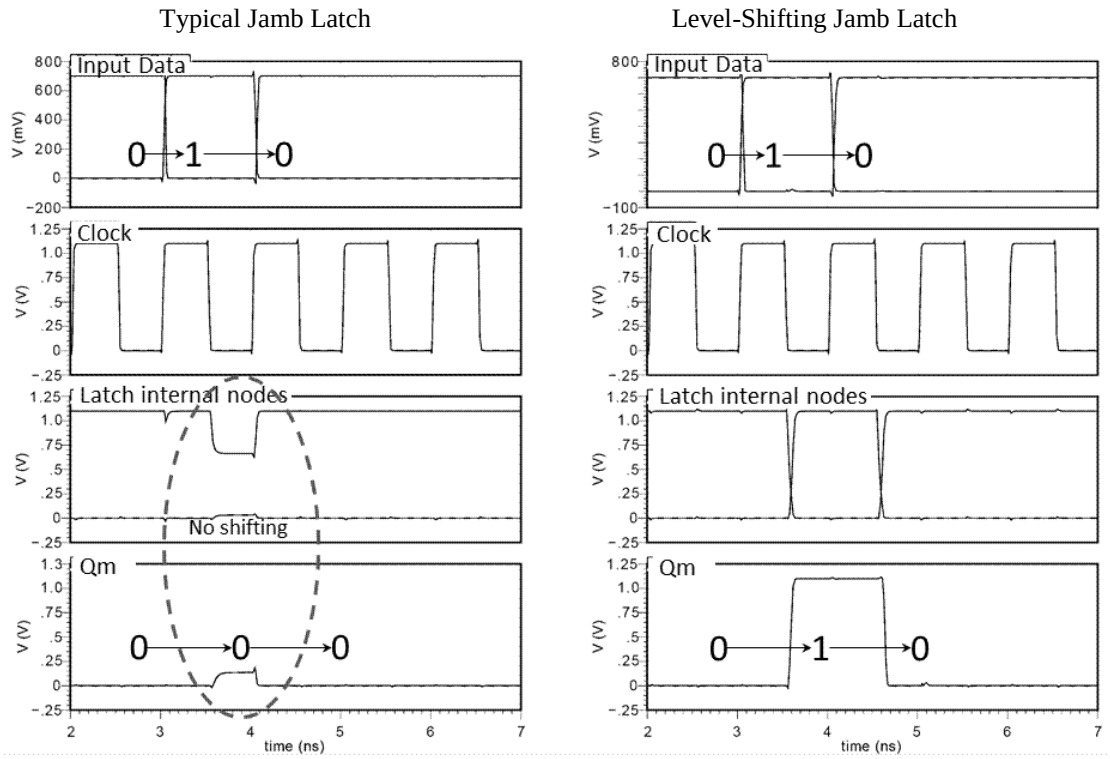


Figure 6.3 Upshifting 0.7V to 1.1V in Jamb Latches

6.1.2 Single-Supply Level-Shifting Latch

An alternative to the differential input latch is the single input latch with reset, shown in Figure 6.4, which can be used instead in order to remove the sender power rail. This technique relies on converting only the low-to-high transitions and afterwards resetting the latch value to low based on a signal protocol via feedback from one of the following clocked stages. The sending domain might require an additional signal to reset the input signal, which is possible under specific conditions.

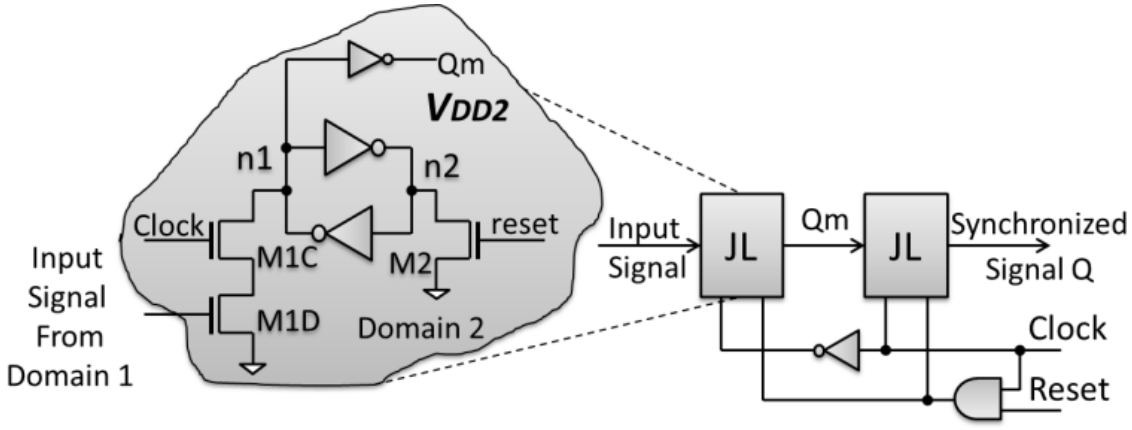


Figure 6.4 Jamb Latch flip-flop with reset as a single-supply level-shifter

6.1.3 Simulation Results

Table 6.1 shows simulation results for the Jamb Latch with a differential input and a reset signal. The timing parameters were measured only for setting the master-latch value to a logic '1' for downshifting and upshifting. The results show small differences between the differential Jamb Latch and the one with the reset. The hold-time during downshifting is negative and upshifting is positive, which is similar to one of the flip-flops measured in [123]. Typically, this is because of the race between the data signal to discharge the bistable node inside the latch and the clock edge [4]. In level-shifting flip-flops the relationship between supply voltages of domain 1 and 2 is also an important factor, because it affects the transition time of the input data, clock and output data. Naturally, within the cross-coupled inverters in Figure 6.2, the internal node n1 is discharged and pulled down to ground by the turned on NMOS transistor M1D driven by the data input signal when the clock is active. The charge stored in node n1 (q_{n1}) is typically equivalent to the product of the potential voltage between that node and ground and the equivalent capacitor at n1, that is, $q_{\mathrm{n1}} = V_{DD2} \times C_{\mathrm{n1}}$. The discharge time of q_{n1} is limited by the size of the discharging current path through NMOS transistors M1C and M1D, which is inversely proportional to their on-resistances. For instance, in the case of downshifting, node n1 will discharge faster because both transistors M1C and M1D are fully opened and q_{n1} is considered small. Having a negative hold time is typical in this case, because the cause and effect of changing input data is sensed quickly through the internal nodes n1 and n2, and to avoid disturbing the output data, input data must not change before the clock. On the other hand, during upshifting, transistor M1C is fully open but transistor M1D is not fully open, which leads to an increase in the on-resistance of M1D which increases the discharge time. Considering a large q_{n1}, the internal node n1 will take an even longer time to discharge. Since the response time to a change in the input data to the internal nodes is large, therefore data may change before the clock edge having a positive hold time.

The left plot of Figure 6.5 shows the possible ranges for the Level-Shifting Jamb Latch (LSJL) to convert from VDD1 to VDD2. In order to increase the shifting range a Robust Latch similar to [82] can be used with the same modification in Figure 6.2. The Level-Shifting Robust Latch (LSRL) can upshift lower voltages than the

Jamb Latch, as shown in the right plot of Figure 6.5. For example, it can upshift from 0.4V to 0.7V, while the Jamb Latch can only do it up to 0.5V. The range is slightly reduced at a slow process technology and temperature of 120°C, as shown in Figure 6.6. For instance, the LSRL can upshift from 0.5V to 1.1V, compared to 1.3V at the nominal corner, and the LSJL can upshift from 0.5V to 0.8V, compared to 1.0V at the nominal corner. Overall, both circuits provide enough range to upshift voltages above 0.6V.

Table 6.2 shows simulation results for two flip-flops comprising of Robust Latches: one with a differential input and the other with a reset signal. In a similar manner to results in Table 6.1, the results in Table 6.2 show small differences between the differential Robust Latch and the one with the reset.

Table 6.1 Level-Shifting Jamb Latch (LSJL) Flip-Flop

Down Shifting	1.2V to 0.8V			
	tCQ(ps)	tSU(ps)	tH(ps)	Power(μW)
Differential	84.1	48.4	-16.5	1.040
With reset	84.9	45.9	-15.4	1.100
Up Shifting	0.8V to 1.2V			
	tCQ(ps)	tSU(ps)	tH(ps)	Power(μW)
Differential	51.8	42.9	14.7	2.460
With reset	51.9	41.0	11.2	2.620

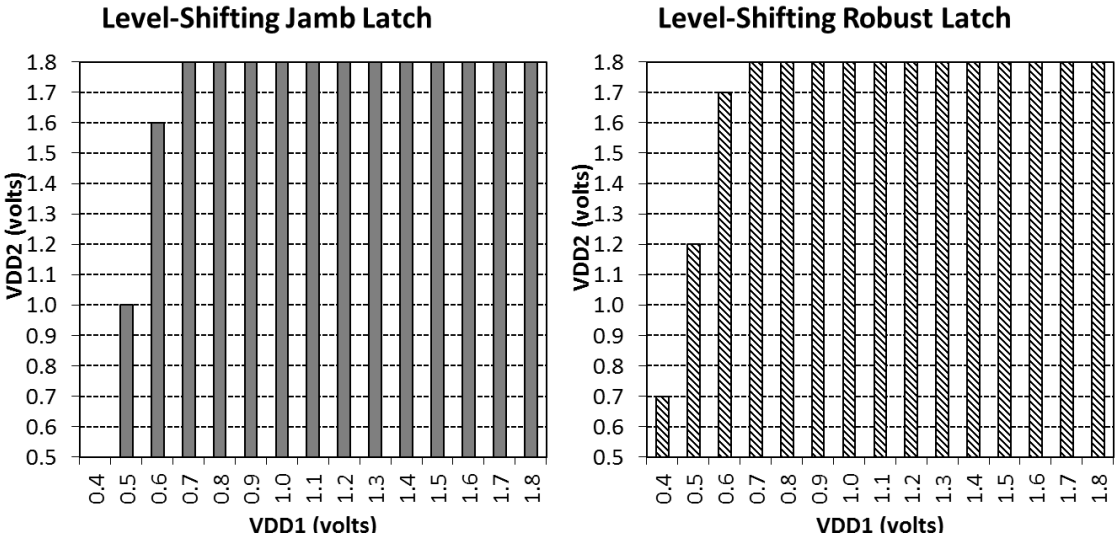


Figure 6.5 Range of level-shifting at nominal corner

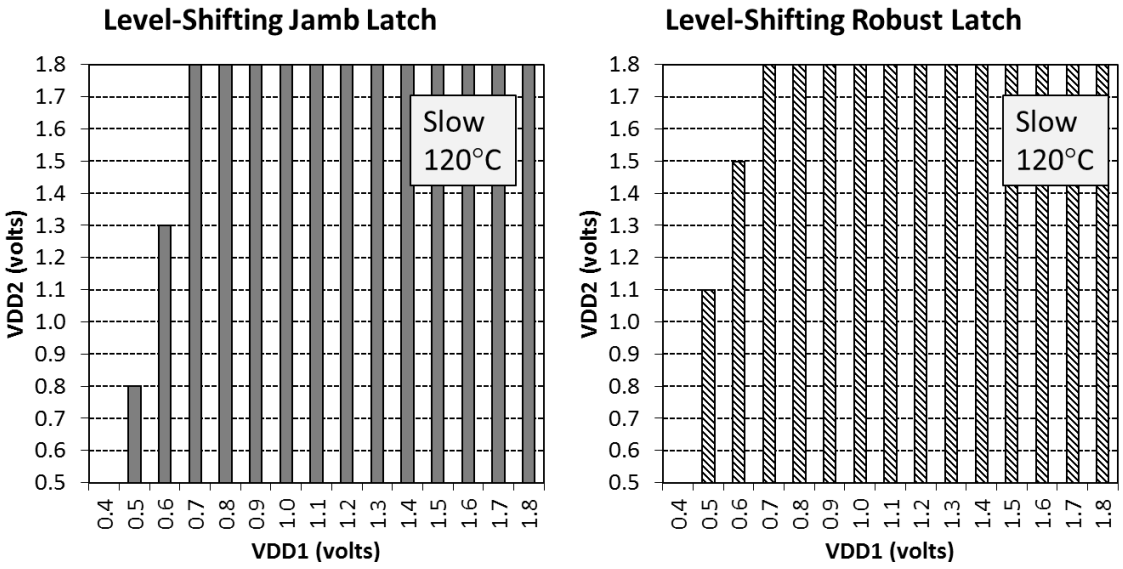


Figure 6.6 Range of level-shifting at slow process corner and 120°C temperature

Table 6.2 Level-Shifting Robust Latch (LSRL) Flip-Flop

Down Shifting	1.2V to 0.8V			
	tCQ(ps)	tSU(ps)	tH(ps)	Power(μW)
Differential	331.0	297.7	-16.3	0.856
With reset	293.0	255.4	-15.6	0.990
Up Shifting	0.8V to 1.2V			
	tCQ(ps)	tSU(ps)	tH(ps)	Power(μW)
Differential	204.0	198.7	10.6	2.040
With reset	180.0	171.0	10.4	2.490

6.2 Handshake Synchronization protocol for level-shifting

In multiple-clock and multiple-voltage domains the communication between the two domains becomes challenging, because of the necessity to convert the voltage and synchronize the signals between the domains. It is worth mentioning that the use of an asynchronous handshake signaling protocol provides a reliable link, but to convert the voltage one level-shifting circuit is required per communicating signal, if both or one of the voltage-domains is fixed. In case both domains have variable voltage supplies, the number of level-shifting circuits is doubled, in addition to a requirement for a third supply voltage that is fixed to facilitate the transfer between the two domains. This scenario is further complicated by using dual-supply level-shifters.

6.2.1 Pseudo Single-Supply Level-Shifting Handshake Synchronization

This section proposes a design technique for single-supply level-shifting and synchronization via a synchronous set and an asynchronous reset. This technique is called Level-Shifting Handshake Synchronization (LSHS) and it utilizes the four-phase handshake protocol with the proposed Jamb Latch with reset in Figure 6.4 and additional control circuits distributed between the sender and receiver.

At the receiving end, a two-flop synchronizer, composed of a Jamb flip-flop with reset and a differential Jamb flip-flop shown in Figure 6.7, is used to synchronize the low-to-high transition request (REQ) signals to set the master Jamb Latch which acts as a level-shifter. The output of the second flip-flop S2 is the synchronized request signal, and it is used to reset the first Jamb flip-flop, assert the acknowledgment (ACK) signal, and enable the register to accept input data.

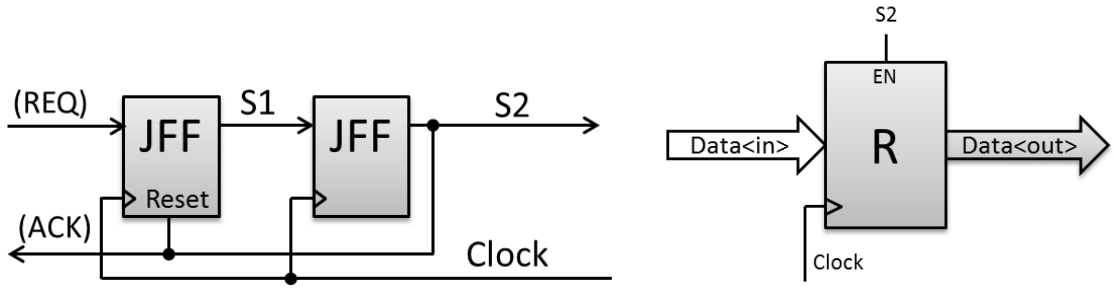


Figure 6.7 Receiving LS synchronizer

At the sending end, a control circuit is placed in between the REQ and ACK signals and the internal signals from the sending domain. The circuit is shown in Figure 6.8 and it is composed of a 2 by 1 multiplexor and a custom C-element. The multiplexor passes the internal request signal R1 or zero to the REQ signal based on the C-element output A1. The C-element is controlled by two signals; the internal request signal R1 and the ACK signal, and acts as a level-shifter. Its output rises to logic '1' if R1 and ACK are true, and falls to logic '0' if only R1 is false. The output of the C-element drives a 2FF synchronizer and the multiplexer, which passes out R1 if A1 is low and a logic '0' if A1 is high to REQ. In this way REQ is deasserted once the ACK signal is sensed by the C-element, which does not cause any conflict with resetting the Jamb FF at the receiver end.

The operation of the control circuit is described by initially setting R1 to logic'1' while A1 is zero to pass it to the REQ signal. When ACK is transitioned from low-to-high, A1 transitions to high and switches the REQ signal to zero. After the A1

signal is synchronized, the signal R1 is reset to logic '0', then A1 is reset to zero and R1 passed again to REQ.

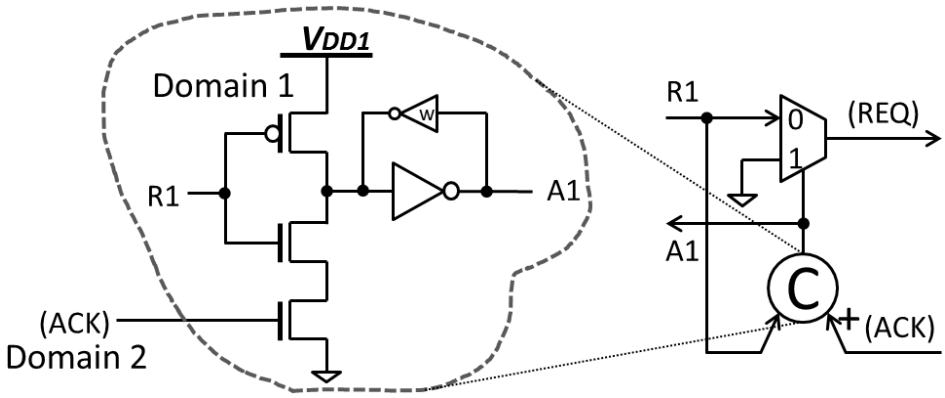


Figure 6.8 Sender handshake circuit to send request and receive acknowledgment

The receiving register circuit, in Figure 6.7, is important because it will need at least *n* level-shifters for an *n*-bit register, as shown in Figure 6.1. A level-shifter with a reset signal is used before the storage element, as shown in Figure 6.9. The level-shifter is based on a C-element circuit, which sets if input data and the enable EN signal are ones, and resets if EN and Reset signals are zeroes. The enable EN signal is a buffered signal of the 'acknowledgment' S2 signal from circuit in Figure 6.7. The output QLS of the level-shifter needs to stay zero in order to sense if data inputs are logic ones or not. The operations in the data register starts when the EN signal is asserted by the S2 signal; then the level-shifter shifts the logic '1' inputs and passes its output QLS to the input QE of a D flip-flop, where the shifted data is stored at the rising edge of the receiving domain clock. Then, if the stored data value is logic '1', the reset signal turns on the PMOS transistor in the level-shifter and waits until S2 is deasserts the enable signal, and finally the value of QLS is reset back to zero.

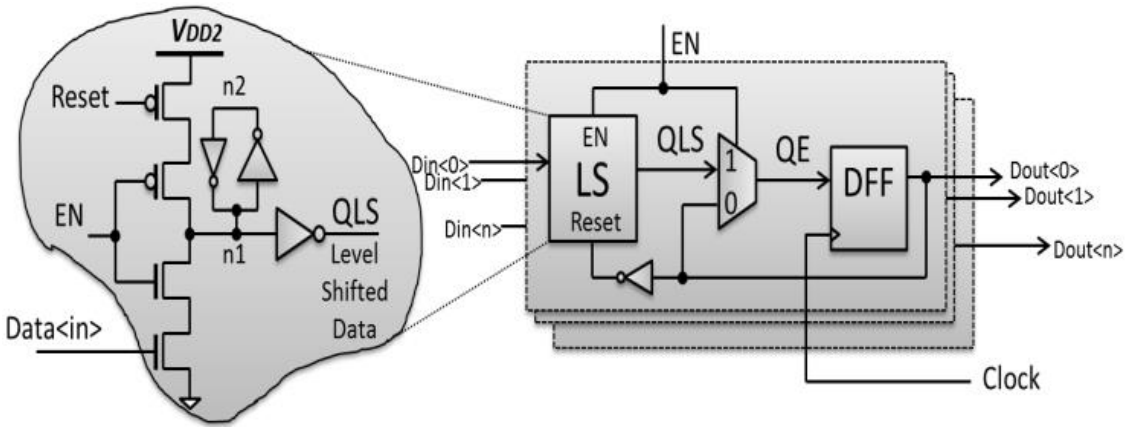


Figure 6.9 Data-register with resettable input latch acting as a level-shifter

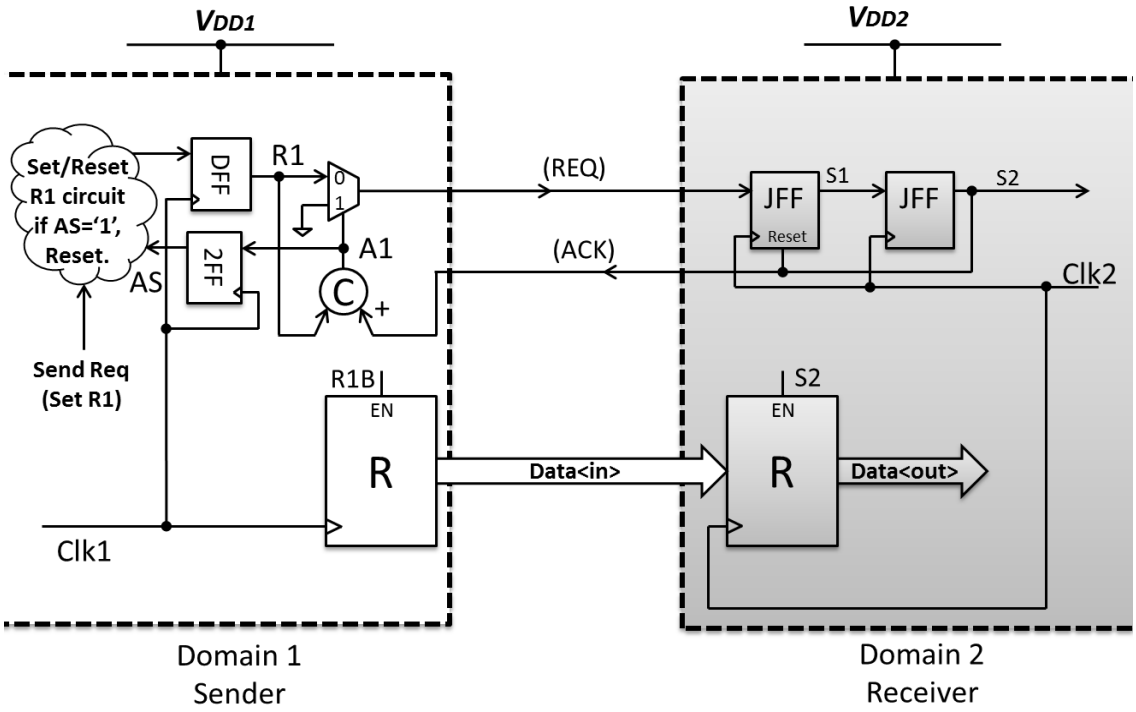


Figure 6.10 LSHS scheme

Figure 6.10 shows the schematic of the whole interface to convert the voltage and synchronize the signals between two domains. It shows handshake process to send data from domain 1 to domain 2. This design is valid under the timing condition that the ACK signal deasserts at most by the time R1 deasserts. In other words, the ACK pulse-width (t_{ACKpw}), which equals to one cycle of receiver clock (T_{CLK2}), must be narrower than the time taken from the ACK rising edge to the AS rising-edge ($t_{ACK+\rightarrow AS+}$), which equals to two cycles of the sender clock (T_{CLK1}) to synchronize A1 signal, assuming the time between AS+ and R1– is negligible. This condition can be written as in Equation (6.1) and Equation (6.2).

$$t_{ACK+\rightarrow AS+} \geq t_{ACKpw} \quad (6.1),$$

$$\therefore 2 \times T_{CLK1} \geq T_{CLK2} \quad (6.2).$$

This assumption states the condition that the sending clock must not be more than two-times faster than the receiving clock. If this condition is disregarded, it may lead to one or two critical outcomes. One concern comes straight after R1 is reset, which releases the hold on data in the sending register, where it may change abruptly to new value. This change in data value will affect the data in the receiving register by either changing data from 0 to 1 or violating the input timing and cause metastability in its D flip-flops. The other concern is that the sender could assert

another request with new data while the ACK signal is still at logic‘1’, which is seen by the sender as a new acknowledgment, and deasserts the new request. This scenario will continue until the ACK signal deasserts.

Figure 6.11 shows a signal-transition-graph of the operation of LSHS, which is initiated by the sender command to send a request to assert R1 and REQ. At the edge of receiver clock Clk2, S1 transits to logic ‘1’. Then at the second Clk2 edge, ACK is asserted, by which the receiving register is enabled to and A1 transits to logic ‘1’ which deasserts REQ. The next steps may occur concurrently or one ahead of the other. The upper path in Figure 6.11, at the third Clk2 edge, S1 is reset by ACK, finally ACK is deasserted at the fourth Clk2 edge. The lower path in Figure 6.11, AS signal transits to logic ‘1’ after two-cycles of the sender clock Clk1 from the time A1 has become one. This signal resets R1 at the following Clk1 edge, by which A1 is reset, and finally AS is reset after two more cycles of Clk1. The dotted line linking arrows leading to ACK– and R1– is the timing condition stated earlier in Equation (6.1) and Equation (6.2) between these two transitions.

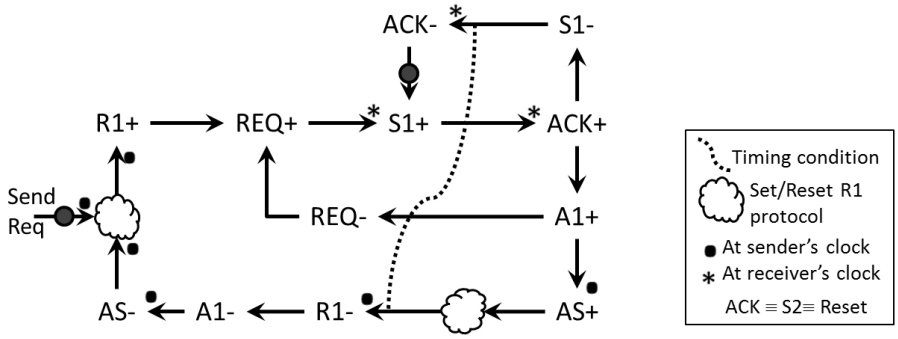


Figure 6.11 Operation of LSHS STG diagram

This design was modeled in UMC90nm CMOS process technology and tested for two cases upshifting and downshifting between two domains; one with V_{DD} of 1.2V and clock period of 1ns, and the other with V_{DD} of 0.8V and clock period of 5ns. Figure 6.12 shows the waveforms for the current operation at typical conditions. These waveforms show the process flow of signals inside and around the LSHS interface demonstrating its operation as shifting and synchronization, assuming no back to back requests, that is, the sender does not assert a new request straight after the de-assertion of the acknowledged request signal. Considering the downshifting waveforms on the right of Figure 6.12, when signal R1 deasserts, domain 1 has the opportunity to send another request, as condition in Equation (6.2) is not fulfilled.

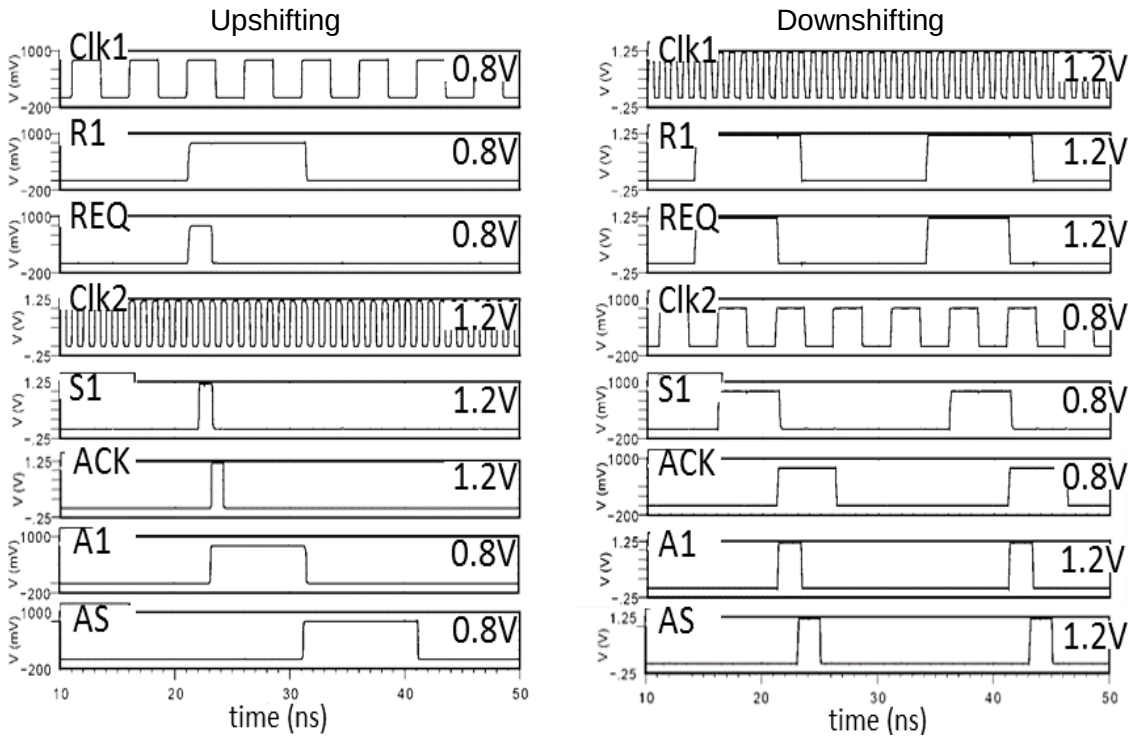


Figure 6.12 LSHS interface waveforms

Figure 6.13 shows the smooth transfer of data from domain 1 to 2 for the same two cases described in the previous paragraph. It shows the shifting of input data of logic '1' is shown and the resetting of the level-shifter without affecting the captured data in the D flip-flop.

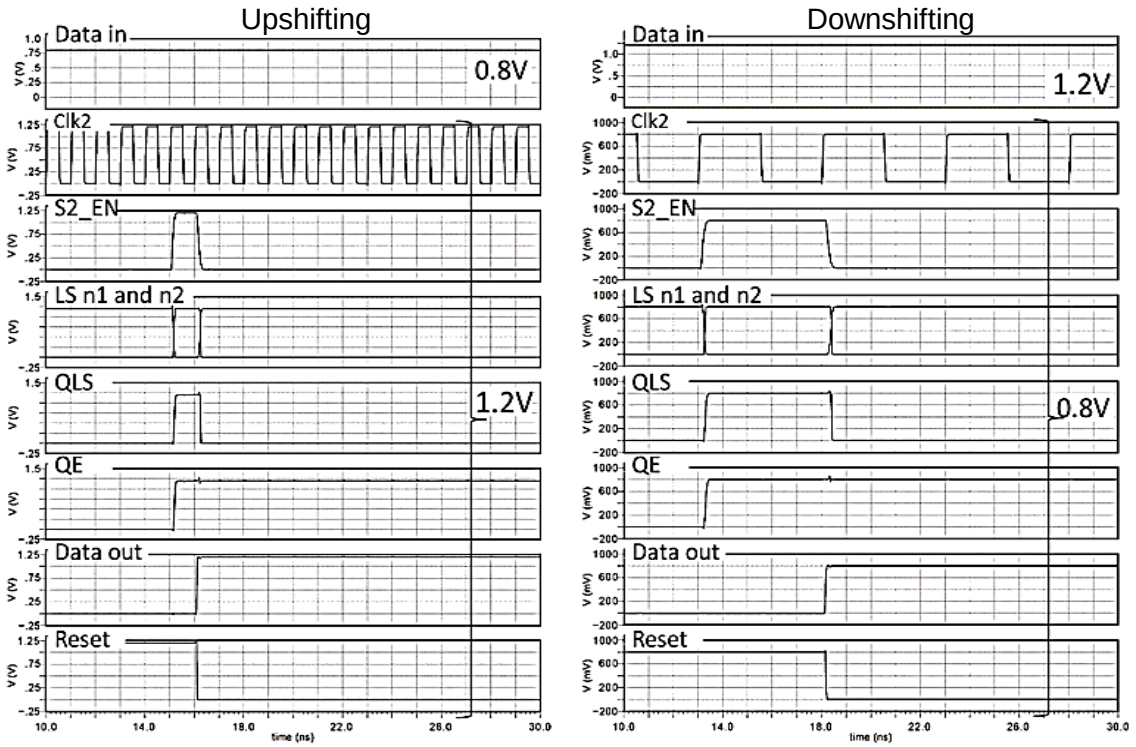


Figure 6.13 Data passing through the register at receiving end of the LSHS

6.2.2 Pseudo LSHS2 for Wider Range of Clock Frequencies

To provide a wider range of clock frequencies between domain 1 and domain 2, the receiving circuits need to be slightly modified. Instead of producing the ACK signal using the enable signal, it is produced by adding a third flip-flop followed by a positive-edge detector to assert a short pulse as the acknowledgment to domain 1, and using the output of the additional flip-flop to reset the first two flip-flops, as shown in Figure 6.14.

In this way, the sending domain clock can be set at a much higher frequency than the receiving domain without any opportunity to create further problem, such as the ones described earlier that could occur in the LSHS design. The condition stated in Equation (6.1) is valid across wider range of sender and receiver clock-cycles. This is because the ACK pulse-width is equivalent to a 3 inverter time-delay in the positive-edge-detecting circuit which is independent of the receiver clock-cycle. The condition in Equation (6.1) can be rewritten for LSHS2 case as in Equation (6.3).

$$2 \times T_{CLK1} \geq 3 \times t_{inv} \tag{6.3}.$$

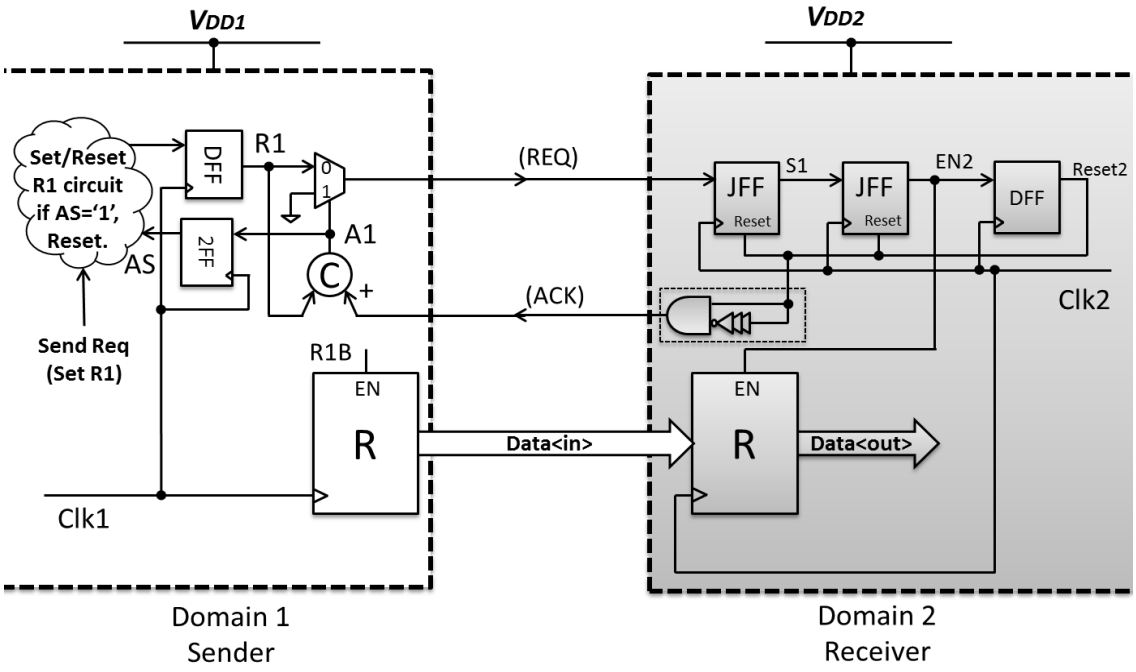


Figure 6.14 Pseudo LSHS2 scheme

The ranges of clock cycles relationship for the LSHS and LSHS2 schemes based on the conditions in Equation (6.2) and Equation (6.3) are shown in Figure 6.15. The range of clocks for the LSHS scheme is half that for the LSHS2 scheme. The

minimum clock cycle is governed by the minimum path delay time between clocked elements [4] plus the time allocated for metastability to resolve [50].

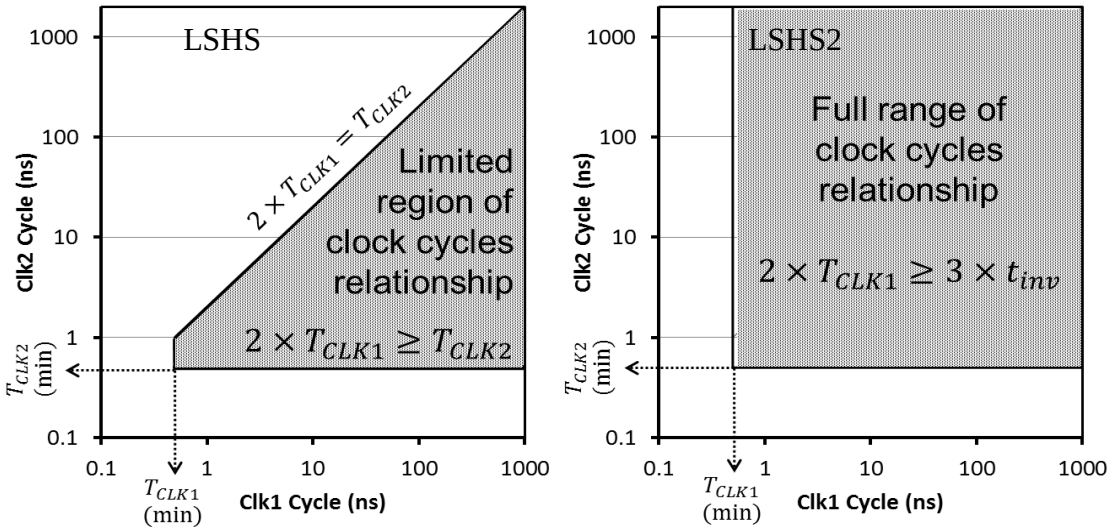


Figure 6.15 Clocks cycles defined ranges for LSHS and LSHS2

Figure 6.16 shows a signal-transition-graph of the operation of LSHS2. In a similar manner to LSHS, the operation starts with the sender asserting R1 and REQ. Then, S1 transits to 1 at Clk2 edge, followed by EN2 at the next Clk2 edge, which enables the receiving register. At the third Clk2 edge, the Reset signal transit to 1. This signal resets S1 and EN2, and asserts ACK signal. The asserted ACK sets A1 to one followed by the de-assertion of REQ, which occurs around the time ACK deasserts. The next steps may occur concurrently or one ahead of the other. The upper path in Figure 6.16, the signal Reset transits to logic-zero at the fourth Clk2 edge. The lower path in Figure 6.16 acts similarly to the lower path in Figure 6.11 described earlier.

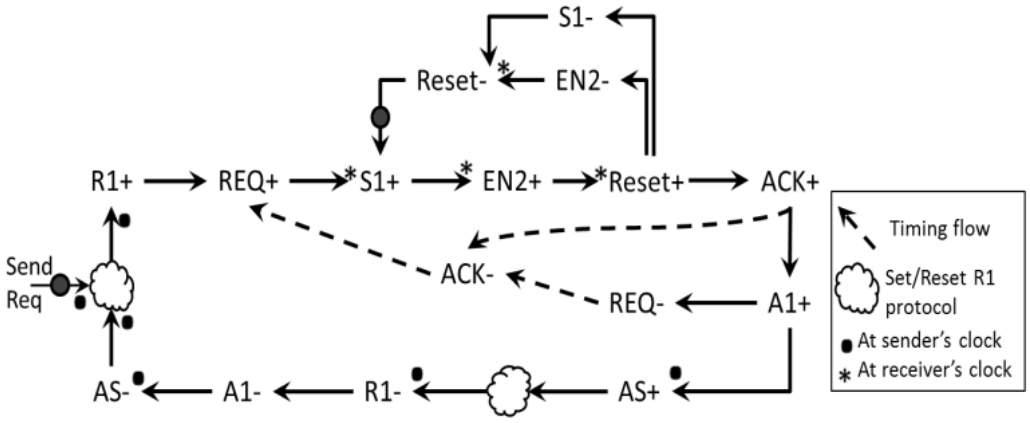


Figure 6.16 Operation of LSHS2 STG diagram

This design was also modeled in UMC90nm CMOS process technology. Figure 6.17 shows two sets of waveform results: one for upshifting and downshifting from

domain 1 to domain 2. One of the domains is with V_{DD} of 1.2V and clock period of 500ps, while the other is with V_{DD} of 0.8V and clock period of 1ns. Both sets of waveforms follow the operation flow presented in Figure 6.16.

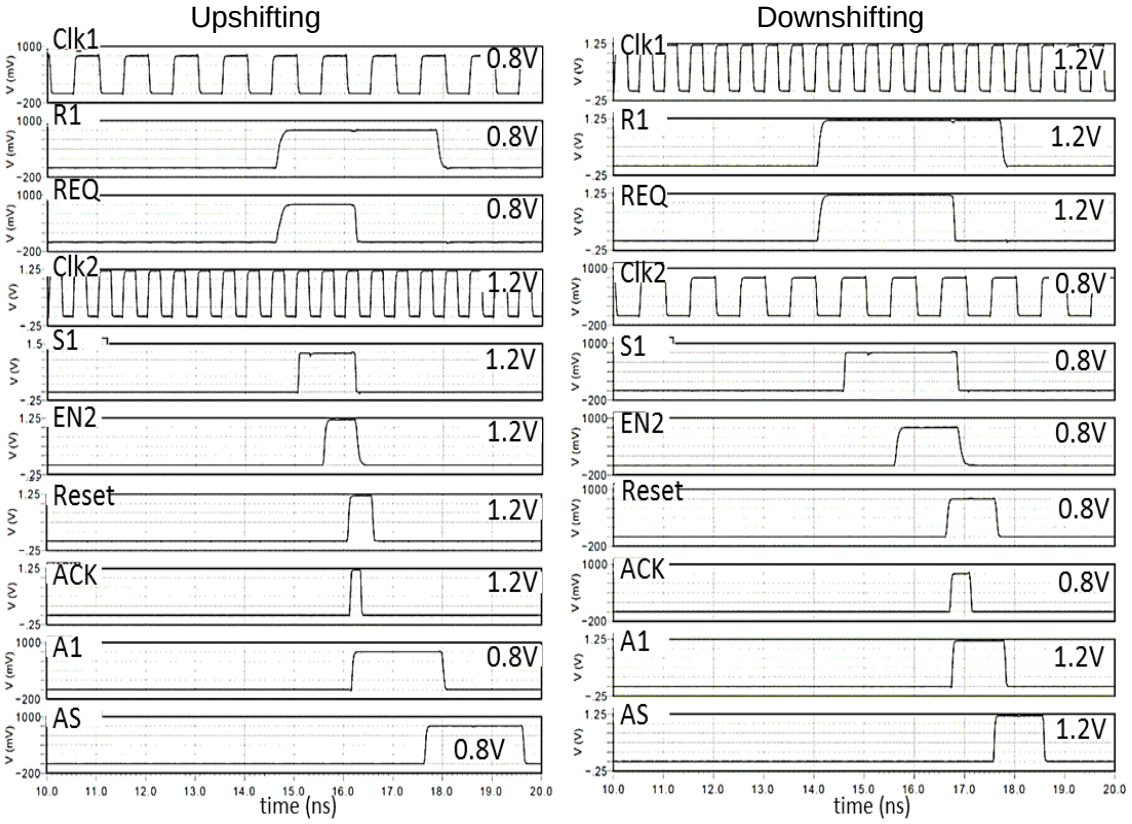


Figure 6.17 LSHS2 interface waveforms

Figure 6.18 shows another two sets of waveforms between the domains at V_{DD} of 1V with 20 times difference between clock period of domain 1 and domain 2. One set shows slow-to-fast transfer with T_{CLK1} = 10ns and T_{CLK2} = 500ps, and the other set shows fast-to-slow transfer with T_{CLK1} = 500ps and T_{CLK2} = 10ns.

A different testing circuit was used to emulate the DVFS mechanism to analyze the LSHS2 interface. Clock signals are generated using a ring of inverters to provide a realistic effect of varying the supply voltage on the clock frequency. At first, LSHS2 was tested for similar DC levels varying a sine-wave amplitude at 40% of the DC voltage and at a frequency of 10MHz for domain 1 and 20MHz for 2, the waveforms are shown in Figure 6.19. From the results, it can be seen that the LSHS2 circuit can adapt to a wide range of voltage-supplies and clock frequency points. This approach eliminates the need for dual-supply connections and additional intermediate voltage-domains.

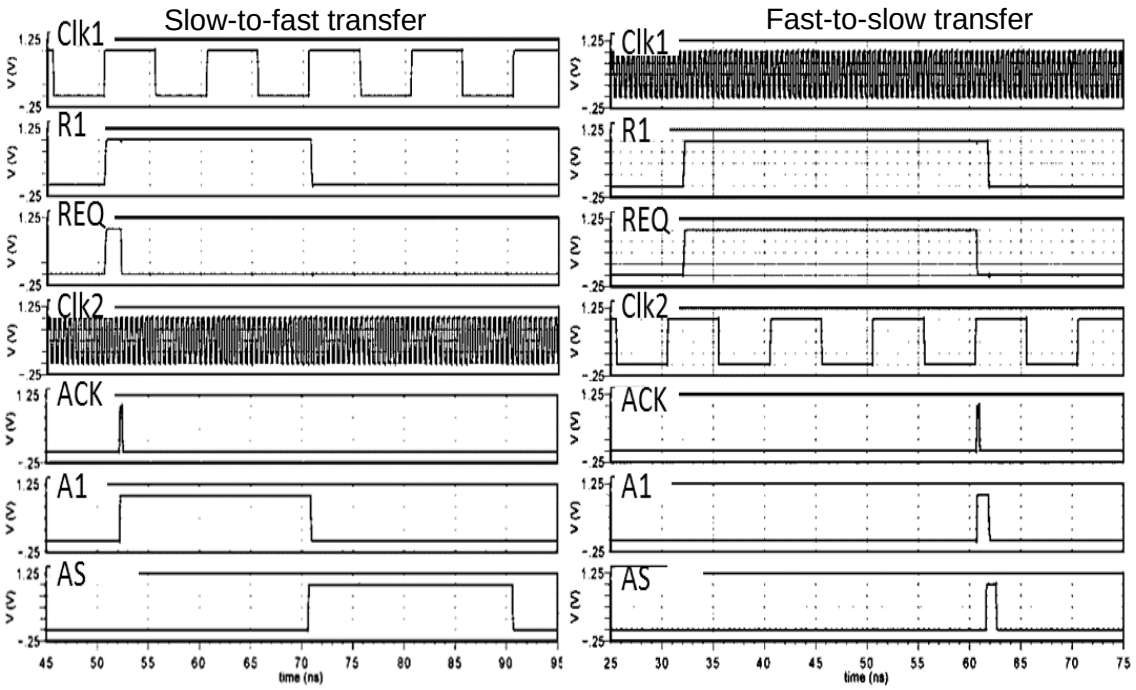


Figure 6.18 LSHS2 slow-to-fast and fast-to-slow transfers

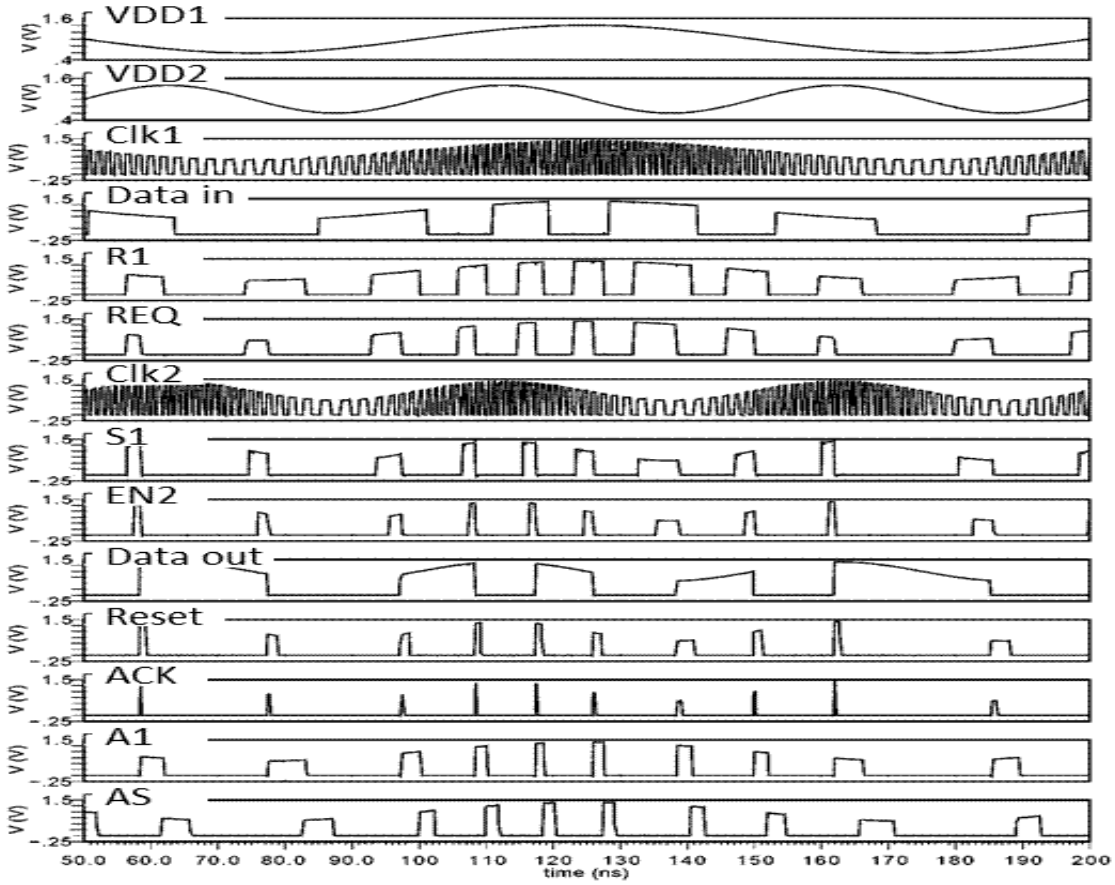


Figure 6.19 LSHS2 emulation under DVFS effect: DC level at 1V

A similar test was repeated but at different DC levels, where domain 1 is set at 0.8V and domain 2 is set at 1.2V. The result waveforms are shown in Figure 6.20. These waveforms show one incident of voltage shifting limits, as explained earlier

section 6.1.3 and presented in Figure 6.5. When REQ is asserted at 70ns, the output of the level-shifting synchronizer S1 did not sense REQ until about 80ns. This is because around 70ns VDD1 was between 0.5V and 0.4V while VDD2 was between 1.4V and 1.2V; their relationship is outside the specified ranges. A similar incident occurred when REQ is asserted at 160ns and S1 is synchronized slightly before 180ns.

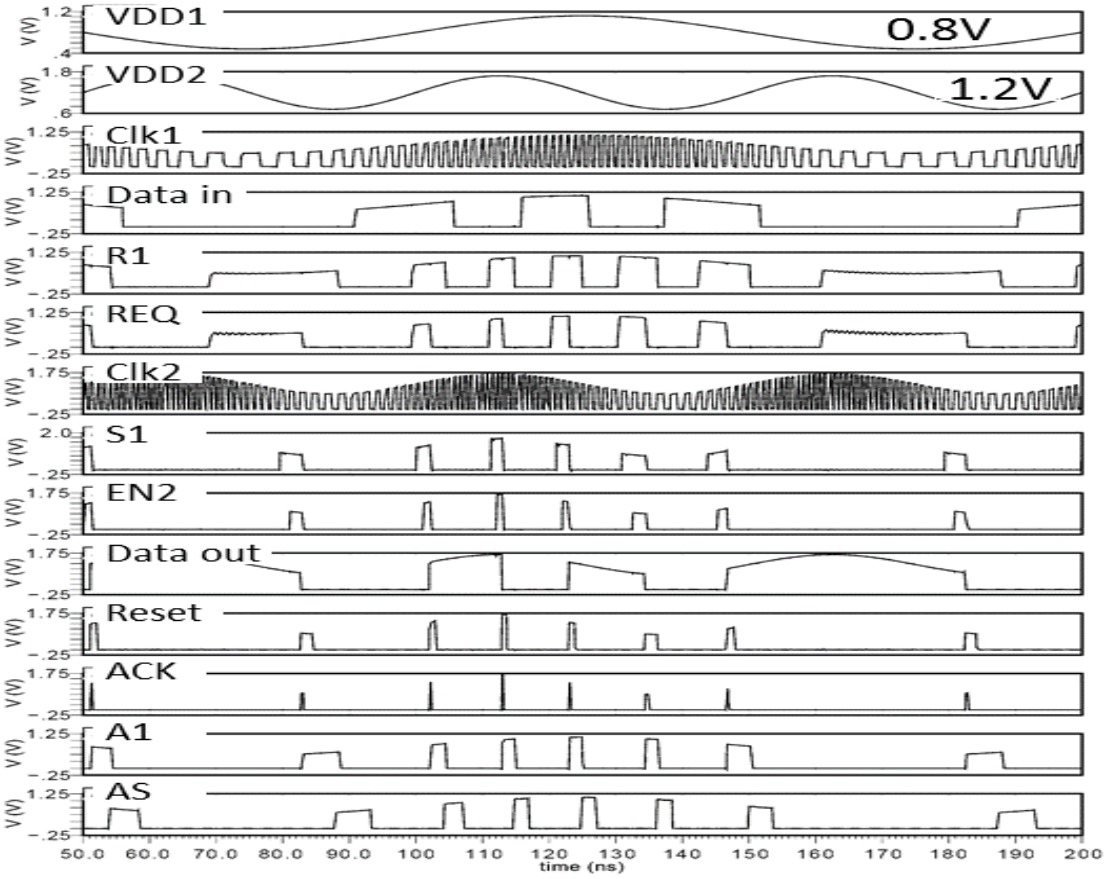


Figure 6.20 LSHS2 emulation under DVFS effect with different DC levels

6.3 Bidirectional Level-Shifting Handshake Synchronization

To add a bi-directional handshake between domains a simple four-phase arbiter can be used with additional circuitry to mimic the reset protocol. Implementing the technique based on 4-phase RGD (Request-Grant-Done) protocol [50], once the arbiter receives a request REQ1 from domain 1 while the channel is free it asserts the G1 signal to domain 2, once domain 2 is done with the task, it asserts ACK1 to the arbiter and domain 1 which releases the channel for the next request. Figure 6.21 shows a complete schematic of a bidirectional LSHS to demonstrate the operation of the level-shifting arbiter.

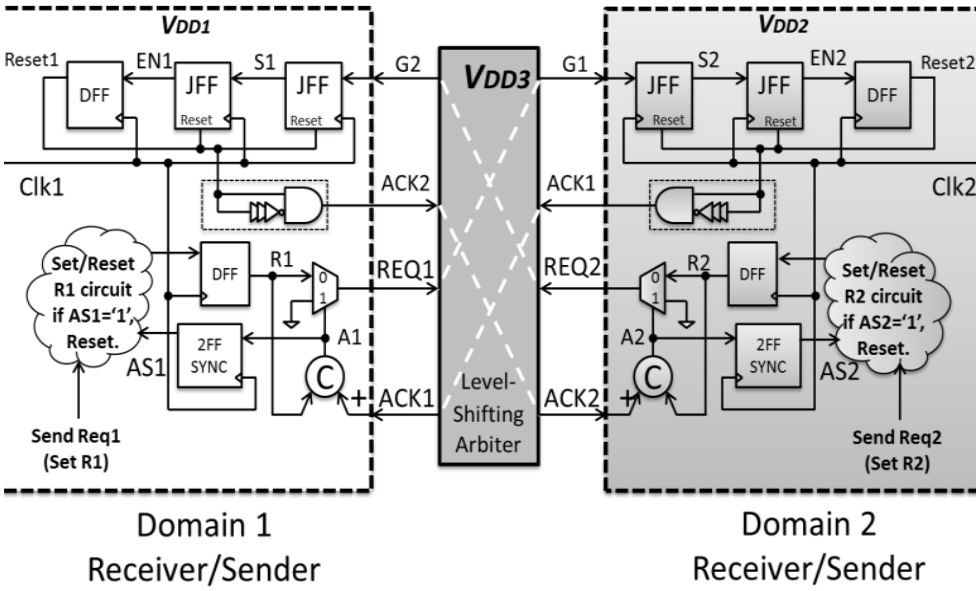


Figure 6.21 Pseudo bidirectional LSHS scheme

The arbiter circuit is shown in Figure 6.22 is based on a MUTEX with two level-shifting latches inserted at its inputs in order to receive the request signals (REQ1 and REQ2) and the done signals (ACK1 and ACK2) from both domains. The feedback from the MUTEX output is inverted and delayed in order to provide additional time for the granted domain to deassert its request signal, and not cause any faulty transitions at one of the MUTEX inputs.

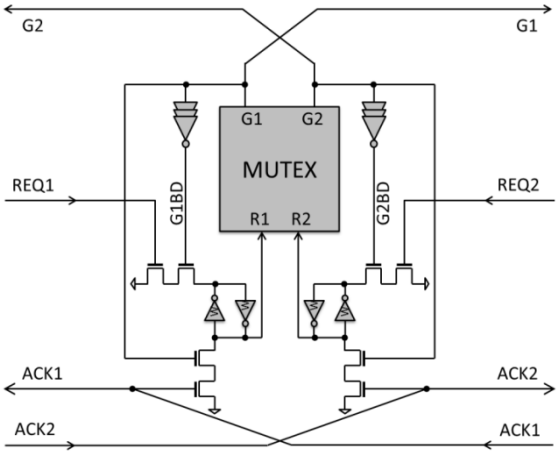


Figure 6.22 Level-shifting arbiter

The timing condition in this circuit is that the resetting of the request signal must occur before or by the time the resetting of the corresponding grant signal is completed. This condition ensures that there are no glitches to be taken by the corresponding level-shifting latch resulting in an incorrect request signal at the corresponding input of the MUTEX. This method also removes the need to route two or more power rails to each domain.

The signal transition graph of the bidirectional LSHS is shown in Figure 6.23, which is a duplication of the one discussed earlier in section 6.2.2 and presented in Figure 6.16. The main difference is the addition of MUTEX function and its timing condition. The MUTEX can only assert one grant signal at a time and once the grant is deasserted then it can produce the other one. The dotted line indicates the timing condition between the REQ1– and G1–, as well as REQ2– and G2–.

The bidirectional LSHS design was modeled in UMC90nm CMOS process technology and tested for three V_{DD} and clock period domains: domain 1 at 0.8V and 5ns clock period, domain 2 at 1.2V and 500ps, and the arbiter at 1.2V. Figure 6.24 shows waveforms for the flow of signals inside and around the bidirectional LSHS interface.

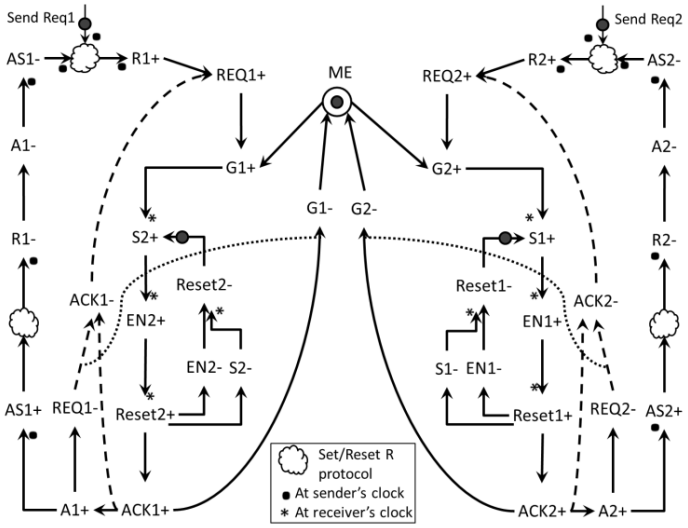


Figure 6.23 STG diagram of the bidirectional LSHS

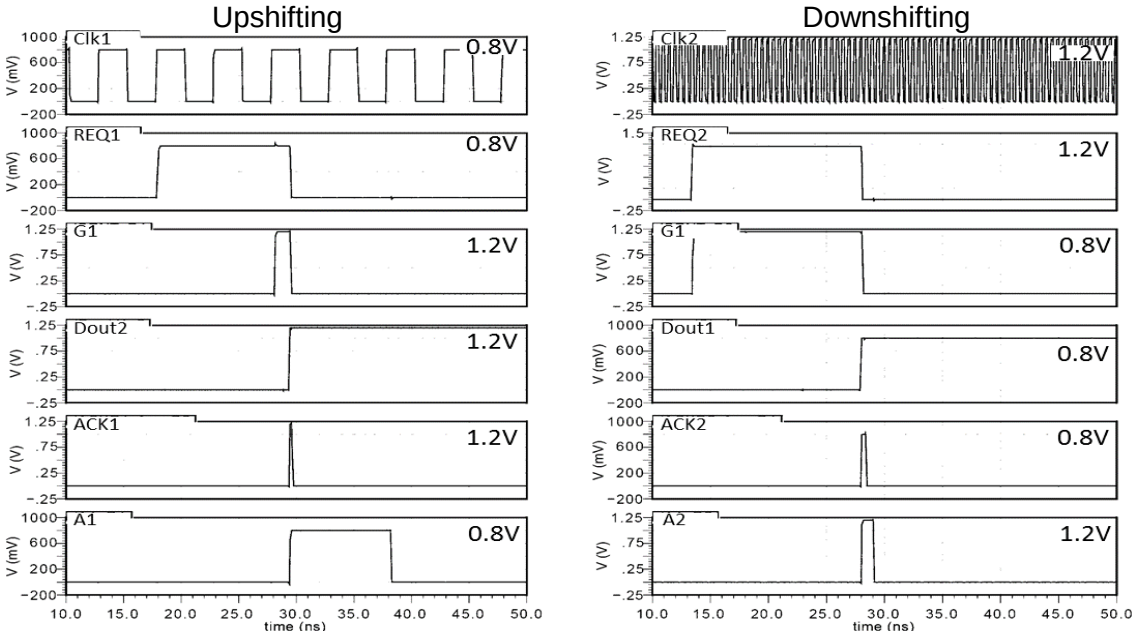


Figure 6.24 Bidirectional LSHS interface waveform

6.4 Summary

This chapter proposed a new technique for transferring signals between multiple-voltage and multiple-clock domains. The level-shifting function can be included in the synchronizer circuit by increasing the latch size to improve the metastability response and swapping the clock and data inputs to provide enough gain to upshift the input signals. This circuit can be used with a single-supply scheme by incorporating a single input for upshifting ones and another for resetting the latch by a feedback signal from one of the following stages based on a specified protocol. It is important that the sender and receiver understand the protocol; it is either the sender keeping the input signal pulse for a short time or the receiver acknowledging the sender in order to withdraw its signal.

The original LSHS design provides a suitable means of data transfer between two domains with different clock and supply voltages via handshake synchronization, but it is limited to different clock frequencies, that is, the sending clock frequency has to be equal or less than the double of the receiving clock frequency, in order for the timing assumption to be valid and not cause any failures. LSHS2 scheme was proposed as an improvement to the LSHS scheme, to accept wider different clock frequencies in both domains by reducing the pulse width of the acknowledgment signal. The LSHS2 structure can handle double the range of clock relationships of that of LSHS. Generally, the transfer between very low V_{DD} and higher ones is limited by technology, as the relationship between voltage-supplies needs to fall within an acceptable range.

The LSHS2 design can be used in bidirectional signaling applications by adding a level-shifting handshake arbiter that is able to sense the request signals and reset them once an acknowledgment is declared. The arbiter cell may fail if its timing condition is unsatisfied if the request signal de-asserts later than the de-assertion of the corresponding grant signal. One way to eliminate this condition is to hold the delayed-inverted-grant signal (G1BD and G2BD) until its request is deasserted using another shifting circuit. In other words, from the assertion of the acknowledgment signal, the request signal must reset faster than the reset of the grant signal. This assumption is valid because the delay time in a multiplexor is much less than the delay time in the level-shifter plus MUTEX plus 3 inverters, unless the sender is operating below the voltage range, such as sub-threshold.

Chapter 7 Discussion, Conclusions and Future Work

7.1 Discussion

The continued scaling of semiconductor technology creates the potential of a system integrated on a single chip. Chip scaling provides a higher transistor density, lower area, cost and power consumption, and faster switching. However, process variations, due to manufacturing limitations, can cause deterioration in the chip's performance and functionality, and consequently, in the production yield. These variations are expected to increase with every new technology node and significantly increase the effects on circuit performance, as variations in timing and power consumption [1, 14, 45].

Multiple voltage domain techniques are currently one of the popular approaches to reduce the power overhead on a chip [4, 55] and reduce the effects of process variations [52-54]. Sometimes, the reduction of voltage supply follows a reduced clock frequency and vice versa, as in the case of DVFS [105]. If multiple processing elements on a chip had different voltage and clock domains, such as [124], they will definitely need to level-shift the voltages and re-time the signals between the multiple domains, which is an MVD/MCD challenge.

In comparison to synchronous techniques, asynchronous techniques have shown to be more tolerant to timing variations caused by process variations or voltage level shifting. Some of these asynchronous techniques use arbiters and synchronizer as the interface block between modules. Nowadays, they have been utilized in different SoCs [58-64]; and ITRS reports predict they will be exploited even more within a single SoC in the future [1]. This will lead to an increase in the number of arbiters and synchronizers needed on a chip, which are required to be more tolerant to variations in process as well as voltage supply and temperature.

This thesis investigated two main components of synchronizers and arbiters: flip-flop and mutual-exclusion element (MUTEX), and how they should be designed to

tolerate PVT variations. Both components can violate the input timing conditions, setup and hold window times, which could cause metastability inside their bistable elements and possibly end in failures. The mean time between failures is an important reliability feature of any synchronizer and depends exponentially on the metastability resolution time constant τ, the available resolution time and the delay through the synchronizer.

Initially, the study in Chapter 3 focused on the design requirements for flip-flops when used in an edge-triggered register and when used in a synchronizer application. In a register, the minimum clock period is determined by the setup time and the delay time from the clock sampling edge to the output Q transition edge. In a synchronizer, the minimum clock period is determined by the required reliability and the structure of the synchronizer. In comparison to ETDFF, the DETFF can perform as well as a register and as a synchronizer; because it has a significantly shorter path between D and Q, which increases the available metastability resolution time and the total latency is reduced by 20% compared with a two flip-flop synchronizer and 30% for a three flip-flop synchronizer. This leads to the design concept of reliable wagging synchronizers, which can be built with significantly lower latency than more conventional designs.

The second study in Chapter 4 focused on the typical MUTEX, in addition to nine proposed M-MUTEXes to minimize the value of τ and enhance its tolerance to PVT variations. The nine approaches are based on either adding current sources, reducing capacitive load, using an error checking circuitry, compensating for the Miller capacitance, unbalancing the latch, or a combination of either two. In comparison to ME0, five modified circuits showed significant improvements on reducing and maintaining τ with high tolerance to PVT variations and changes in latch and load size. As supply voltages are reduced and under process variations, circuits ME0, ME2B, ME3A and ME4A will become less effective at resolving metastability, whereas, ME1A, ME1B, ME2A, ME3B, ME5A and ME5B circuits may resolve metastability faster.

Three design approaches are proposed to provide variation-tolerant synchronizers. First, the wagging synchronizer is modified to significantly increase reliability and robustness in comparison to that of the conventional flip-flop synchronizer, by utilizing Robust Latches and reconfigurable clocking circuits. The

results in Chapter 5 of the wagging synchronizers showed to have higher MTBF for the same number of synchronizing clock cycles.

Second, the MADAC technique, which can be used in a single flop synchronizer, relies on swiftly detecting a metastable event and correcting it by enforcing the previously stored logic value. Based on results in Chapter 5, the initial proposed MADAC approach with CTG circuit lowers τ by at least 30% at 1V, but only a small improvement at lower V_{DD}. On the other hand, the proposed RML lowers τ significantly even at low supply voltages.

Finally, the novel level-shifting handshake synchronization technique proposed in Chapter 6 showed that it can transfer signals between MVD and MCD without the need of conventional level-shifters between the domains or multiple power supplies within each domain. This employment of level-shifting and synchronization is normal as the boundaries of clock domain are normally chosen to be the borderline between voltage domains because of the natural relationship between the clock switching frequency and the supply voltage. The LSHS and LSHS2 interfacing schemes demonstrated the concept and feasibility of the proposed technique; yet, it is not limited to only these schemes. Still, the chosen synchronizer circuits need to be carefully designed to keep an acceptable MTBF.

7.2 Conclusions

This thesis presented a number of contributions in the field of synchronizers for multiple clock and voltage domains. As the scaling of integrated circuit continues, there is a high demand to utilize multiple clock techniques on system on a chip, to eliminate the global clock problems, mainly in terms of timing variations, along with the concept of multiple voltage domains, to reduce the power overhead. This demand raises the need to investigate the possible new design techniques for synchronizers and arbiters to enhance their performance and to be tolerant to PVT variations. Synchronizer and arbiter circuits are prone to metastable failures, which are related to the metastability resolution time constant τ that has an exponential impact on their performance and reliability.

Flip-flop and MUTEX circuits were primarily investigated is this thesis. First, the trade-off in flip-flops when used as registers or used in a synchronizer application

was analyzed, and led to the concept of the 'wagging' synchronizer. Then, the MUTEX design was investigated and modified at the circuit and transistor levels, to adjust the value τ and its tolerance. The results showed five modified circuits has significantly improved their τ and maintained a high tolerance toward variations in PVT and size of the latch and load. Then, the 'wagging' synchronizer was proposed to significantly increase reliability over that of the conventional two flip-flop synchronizer. The robustness of the wagging synchronizer can be enhanced in multiple voltage environments by using robust τ latches or adding one more cycle of synchronization. Then, the MADAC technique was also proposed to detect a metastable event and correct it by enforcing the previously stored logic value in a flop synchronizer. This technique significantly reduces the resolution time uncertainty, even under process variations and low-voltages. Finally, a pseudo LSHS technique was proposed to transfer signals between MVD/MCD that do not require conventional level-shifters between the domains or multiple power supplies within each domain. This interface circuit used a synchronous set and feedback reset protocol which provides level-shifting and synchronization of all signals between the domains, from a wide range of voltage-supplies and clock frequencies.

Digital designers considering a new synchronizer design with an adaptable number of cycles may consider the reconfigurable wagging synchronizer. On the other hand, if they are targeting a similar synchronizer circuit plus operating at low supply voltages, they should consider the robust wagging synchronizer. However, if their design specifications require a single robust flop synchronizer that tolerates lower supply voltages and process variations, then the RML synchronizer is recommended, because of its significant capability to tolerate variations of τ caused by process variability and low V_{DD}. Lastly, as future designs employ MCD/MVD on chip, the LSHS2 concept may be utilized, but not necessarily using the same circuits.

In conclusion, synchronizer circuits can be designed to tolerate PVT variations to a greater extent by employing the wagging technique or using a MADAC latch, while MUTEX tolerance can suffice with small circuit modifications. Communication between MVD/MCD can be achieved by an asynchronous handshake with internal resetting protocols without a need for adding level-shifters. Overall, this thesis proposed a number of contributions to build reliable and resilient circuits

intended for synchronizers and arbiters that reliably operate under high PVT variability conditions.

7.3 Future Work

This thesis has shown how synchronizers and arbiters should be designed to tolerate PVT variations. Still, there are some issues have not been considered in this study as well as possible solutions, which can be investigated as part of future research work.

- Over all this work, the design of synchronizers and arbiters was considered primarily at the circuit level. However, there are a number of synchronization design issues need to be investigated at the perspective of system and architectural levels. For example, a synchronizer-based GALS interface or FIFO can be optimized to obtain lower latencies and higher throughput considering the presence of PVT variations as well as multiple clock and voltage domains.
- Soft-errors, caused by alpha-particles striking the substrate of the transistors, need to be investigated in the design of synchronizers and arbiters to be soft-error tolerant, although these circuits are usually comprised of large transistors. In a flip-flop with MADAC technique, current spikes in the master latch can be resolved quickly, unless the charge is large enough to flip the latch and the charge build-up speed is faster than the detection and correction time of the MADAC circuit.
- The reconfigurable wagging synchronizer, in Chapter 5, requires monitoring by the processing element responsible for scaling the voltage supply and the clock frequency, which may prevent the exploitation of this circuit due to design difficulties. Alternatively, this wagging synchronizer design can be enhanced if it has additional intelligent sensing circuitry intended for measuring the value of τ and estimating the MTBF and the required resolution time, then deciding the number of path-ways to be activated in this wagging synchronizer, to become adaptive and MTBF aware synchronizer.
- In the design of modified MUTEX circuits, a number of designs had

improved τ significantly. However, the delay has considerably increased in all M-MUTEXes compared to the ME0 circuit. There is a need to balance the modifications to trade-off between the MUTEX τ and delay. Additionally, the design of multi-way exclusion elements needs to be investigated, and particularly how to eliminate the oscillatory behavior in the tri-flop, and how to provide a robust metastability and oscillation filter.

- In the multiple voltage design, single-supply level-shifting latch or MUTEX circuit may be comprised using the analogy to the analog amplifier, for example, the common-source NMOS amplifier with an active PMOS load. Then, the metastability behavior in these types of circuits needs to be investigated considering the different possible voltage domains, while maintaining lower overhead power consumption compared to the conventional dual-supply level-shifting circuits.

Appendix A Lexicon Abbreviations

Abbreviation	Description
2FF	Two Flip-Flops
3FF	Three Flip-Flops
3WS	Three-way Wagging Synchronizer
4FF	Four Flip-Flops
4WS	Four-way Wagging Synchronizer
5FF	Five Flip-Flops
5WS	Five-way Wagging Synchronizer
6WS	Six-way Wagging Synchronizer
AVS	Adaptive Voltage Scaling
C2MOS	Clocked CMOS logic gate
CCC	Clock Control Circuit/Cell
CMOS	Complementary MOS
CTG	Compound TG
DCVS	Differential Cascode Voltage Switch/Shifter
DETFF	Dual-edge Triggered D Flip-Flop
DSLS	Dual-Supply Level-Shifter
DVFS	Dynamic Voltage Frequency Scaling
ETDFF	Edge Triggered D Flip-Flop
FF	Flip-Flop
FO4	Fan-Out-of-4, delay of inverter driving 4 similar inverters
FPGA	Field Programmable Gate Array
GALS	Globally Asynchronous and Locally Synchronous
GBP	Gain Bandwidth Product
JL	Jamb Latch
LS	Level-Shifter
LSFF	Level-Shifting Flip-Flop
LSHS	Level-Shifting Handshake Synchronization
LSJL	Level-Shifting Jamb Latch
LSRL	Level-Shifting Robust Latch
MADAC	Metastability Auto-Detection And Correction
MCD	Multiple Clock Domain
MCV	Multiple Clock and Voltage
ME	Mutual Exclusion element
ME0	Typical MUTEX
ME1A	M-MUTEX with current source biased by output feedback
ME1B	M-MUTEX with current source and reduced load
ME2A	M-MUTEX with g_m boosting during metastability

Abbreviation	Description
ME2B	M-MUTEX with g_m boosting during metastability, current-source and reduced-load
ME3A	M-MUTEX with Cascode current source feedback
ME3B	M-MUTEX with Cascode current source feedback and g_m boosting during metastability
ME4A	OAI based MUTEX utilizing Cascode current source feedback
ME5A	M-MUTEX with biased choice
ME5B	M-MUTEX with biased choice during metastability
M-MUTEX	Modified MUTEX
MOS	Metal Oxide Semiconductor
MSFF	Master-Slave Flip-Flop
MTBF	Mean Time Between Failures
MUTEX	Mutual Exclusion element
MVD	Multiple Voltage Domain
*N*FF	*N* (number of FF) Flip-Flops
NMOS	N-type MOSFET
NoC	Network-on-Chip
*N*WS	*N* (number of ways) Wagging Synchronizer
OAI	Or-And-Invert logic gate
PG	Pass-Gate
PGLS	Pass-Gate Level-Shifter
PLL	Phase Locked Loop
PMOS	P-type MOSFET
PVT	Process, Voltage and Temperature
RGD	Request-Grant-Done
RML	Robust MADAC Latch
RL	Robust Latch
SETFF	Single Edge Triggered D Flip-Flop
SoC	System-on-Chip
SSLS	Single-Supply Level-Shifter
STG	Signal Transition Graph
TG	Transmission Gate
TGFF	Transmission Gate D Flip-Flop
WS	Wagging Synchronizer

Appendix B Symbols Annotation

Symbol	Description
α	Switching activity
α_{LL}	Latch to Load size ratio
μ , μ_{eff}	Carrier mobility, effective carrier mobility
τ	Metastability resolution time constant
v_{sat}	Velocity saturation of carriers
ω_{-3dB}	Frequency bandwidth
A	Voltage amplifier gain
C_{DB}	Drain to Body junction capacitance
C_{GB}	Gate to Body parasitic capacitance
C_{GD}	Gate to Drain parasitic capacitance
C_{GS}	Gate to Source parasitic capacitance
C_M	Miller effect capacitance
C_{out}	Output capacitance
C_{ox}	Oxide capacitance
C_{SB}	Source to Body junction capacitance
f_c , f_{CLK}	Clock frequency
f_d , f_{Data}	Data frequency
f_{R1} , f_{R2}	Average frequency of request 1,2
g_{out}, g_o	Output conductance $= 1/R_{out}$
g_m	Transconductance
I_D	Drain current
I_{Dsat}	Saturation drain current
L	Channel length of MOS transistor
P	Probability
R_{out}	Output resistance
T_C , T_{CLK}	Clock period
t_{CQ}	Clock to output propagation delay time
t_{DC}	Data to Clock separation time
t_{DQ}	Data to output propagation delay time
t_H	Hold time
t_{inv}	Inverter delay time
t_{ox}	Oxide thickness
t_R	Available resolution time for metastability
t_{SU}	Setup time
T_w	Metastability window
V_{CM}	Common mode voltage
V_{dd} , V_{DD}	Voltage supply in CMOS technology

Symbol	Description
V_{DDH}	Voltage supply of a domain with higher V_{DD}
V_{DDL}	Voltage supply of a domain with lower V_{DD}
V_{DM}	Differential mode voltage
V_{DS}	Drain to Source voltage
V_{GS}	Gate to Source voltage
V_{Dsat}	Saturation drain voltage
V_M	Middle\switching \inverting voltage point
V_{SS} , gnd	Ground potential of 0 volts
V_T, V_{TH}	Threshold voltage of MOS transistor
W	Channel width of MOS transistor

Appendix C UMC 90nm Model

This appendix lists of the Spectre Model parameters for the UMC 90nm Standard Performance CMOS transistors (SP_RVT1.0V/1.2V).

C.1 NMOS Models

(i) Nominal Model

```
// DEVICE 1
 model n_10_sp bsim4 type=n
+ version=4.3000e+00       binunit=1.0000e+00       paramchk=1.0000e+00
+ mobmod=0.0000e+00        capmod=2.0000e+00        igcmod=1.0000e+00
+ igbmod=1.0000e+00        geomod=0.0000e+00        diomod=2.0000e+00
+ rdsmod=0.0000e+00        rbodymod=0.0000e+00
rgatemod=0.0000e+00
+ permod=1.0000e+00        acnqsmod=0.0000e+00
trnqsmod=0.0000e+00
+ rgeomod=1.0000e+00       fnoimod=1.0000e+00       tnoimod=0.0000e+00
+ toxe=2.2500e-09 + dtoxe_n_10_sp
+ toxp=1.8220e-09 + dtoxp_n_10_sp                  toxm=2.2500e-09
+ epsrox=3.9000e+00        xj=1.2000e-07            ngate=1.3000e+20
+ ndep=1.0000e+17          nsd=1.0000e+20           rsh=8.0000e+00
+ wint=2.0210e-08 + dwint_n_10_sp                  lint=-4.0910e-09
+ vth0= - 1.0000e-03 + dvth0_n_10_sp               k1=1.5690e-01
+ k2=4.0000e-03           k3= - 1.2880e+00 + dk3_n_10_sp
+ k3b=2.9280e+00          w0=9.0000e-08            dvt0=3.9630e+00
+ dvt1=5.6320e-01         dvt2=-3.3200e-02         dvt0w=5.2510e-01
+ dvt1w=1.1170e+07        dvt2w=-7.7000e-01        dsub=3.9000e-02
+ minv=7.7040e-01         voffl= - 4.9270e-09 + dvoffl_n_10_sp
+ dvtp0=8.9100e-06        dvtp1=-8.0630e-01
+ lpe0=1.0000e-10 + dlpe0_n_10_sp                  lpeb=-1.6990e-07
+ phin=8.7670e-02         cdsc=4.6490e-04          cdscb=1.5000e-04
+ cdscd=0.0000e+00        cit=1.5520e-03           voff=-6.3870e-02
+ nfactor=1.0000e-01      eta0=5.0000e-05          etab=-1.8500e-04
+ vfb=-1.0000e+00         u0=2.3200e-02            ua=-1.5500e-09
+ ub=3.4800e-18           uc=1.7330e-10            vsat=1.6250e+05
+ a0=8.8340e+00           ags=1.0020e+00           a1=0.0000e+00
+ a2=1.0000e+00           b0=0.0000e+00            b1=1.0000e-08
+ keta=-4.4080e-02        dwg=-5.4000e-09          dwb=4.8000e-09
+ pclm=1.0000e-01         pdiblc1=1.0000e-07       pdiblc2=3.9540e-02
+ pdiblcb=1.0000e-01      drout=5.5990e-01         pvag=8.6180e+00
+ delta=1.0000e-02        pscbe1=6.5350e+09        pscbe2=3.3000e-01
+ fprout=1.0000e-02       pdits=6.1100e-01         pditsd=8.8000e-01
+ pditsl=1.0000e+05       rdsw=5.0000e+01 + drdsw_n_10_sp
+ rdswmin=5.0000e+01      prwg=2.8000e-01          prwb=4.4700e-01
+ wr=1.0000e+00           alpha0=2.0000e-07        alpha1=4.0000e+00
+ beta0=1.5000e+01        agidl=1.1080e-08         bgidl=1.3900e+09
+ cgidl=2.9630e-01        egidl=9.4400e-01         toxref=2.2500e-09
+ dlcig=1.8000e-08        aigbacc=1.1980e-02       bigbacc=8.0130e-03
+ cigbacc=6.2560e-01      nigbacc=4.3970e+00       aigbinv=1.5300e-02
+ bigbinv=4.8520e-03      cigbinv=1.0000e-03       eigbinv=1.1000e+00
+ nigbinv=1.6000e+00      aigc=1.1380e-02          bigc=1.8790e-03
+ cigc=1.0000e-04         aigsd=9.8830e-03         bigsd=1.2690e-03
```

```
+ cigsd=1.5540e-01          nigc=1.0000e+00          poxedge=1.0000e+00
+ pigcd=2.5000e+00          ntox=1.0000e+00          dlc=1.6400e-08
+ dwc=-3.0000e-08           xpart=1.0000e+00
+ cgso=5.0000e-11 + dcgso_n_10_sp
+ cgdo=5.0000e-11 + dcgdo_n_10_sp                    cgbo=0.0000e+00
+ cgdl=2.2000e-10 + dcgdl_n_10_sp
+ cgsl=2.2000e-10 + dcgsl_n_10_sp                    clc=1.0000e-07
+ cle=6.0000e-01            cf=9.2600e-11 + dcf_n_10_sp
+ ckappas=3.0000e+00        vfbcv=-1.0000e+00        acde=2.8080e-01
+ moin=1.1830e+01           noff=2.4860e+00          voffcv=-1.3720e-02
+ ef=0.9448
+ noia=3.8700000e+41        noib=1.8600000e+25       noic=6.7000000e+08
+ em=6.3600000e+06          ntnoi=1.0
+ xl= - 1.0000e-08 + dxl_n_10_sp
+ xw=0.0000e+00 + dxw_n_10_sp                         dmcg=1.6000e-07
+ dmci=1.0000e-07           dwj=0.0000e+00           jss=2.3350e-07
+ jsws=7.0330e-14           jswgs=3.2986e-14         ijthsfwd=3.4450e-03
+ ijthsrev=1.6910e-03       bvs=1.1470e+01           xjbvs=1.0000e+00
+ pbs=6.1000e-01            cjs=1.0700e-03 + dcjs_n_10_sp
+ mjs=2.9000e-01            pbsws=9.9000e-01
+ cjsws=1.2600e-10 + dcjsws_n_10_sp                   mjsws=1.0000e-01
+ pbswgs=6.0000e-01         cjswgs=2.3100e-10 + dcjswgs_n_10_sp
+ mjswgs=9.8900e-01         tnom=2.5000e+01          kt1=-3.8000e-01
+ kt1l=1.0000e-09           kt2=-4.0740e-02          ute=-1.0220e+00
+ ua1=4.3500e-09            ub1=-4.1040e-18          uc1=2.6360e-10
+ prt=0.0000e+00            at=1.0000e+05            njs=1.0560e+00
+ tpb=1.3000e-03            tcj=9.0000e-04           tpbsw=3.5150e-03
+ tcjsw=4.0000e-04          tpbswg=1.2470e-03        tcjswg=8.2290e-03
+ xtis=3.0000e+00           ll=4.3480e-16            wl=-4.0050e-15
+ lln=9.0000e-01            wln=1.0000e+00           lw=3.2080e-15
+ ww=-1.5010e-15            lwn=1.0000e+00           wwn=1.0000e+00
+ lwl=-1.6220e-21           wwl=1.7820e-22 + dwwl_n_10_sp
+ llc=-9.0100e-16           wlc=0.0000e+00           lwc=0.0000e+00
+ wwc=1.0000e-15            lwlc=0.0000e+00          wwlc=0.0000e+00
+ lmin=8.0000e-08           lmax=5.0000e-05          wmin=1.2000e-07
+ wmax=1.0000e-04           pvth0= - 1.2500e-03 + dpvth0_n_10_sp
+ lk3=7.2000e-01            wk3=-1.3000e-01          lk3b=-2.0000e-01
+ pk3b=2.0000e-02           ldsub=-1.2720e-03        wdsub=5.0000e-04
+ llpe0=3.8910e-08 + dllpe0_n_10_sp                   lcit=7.0000e-05
+ wvoff=-1.3400e-03         leta0=1.3000e-05         weta0=3.7760e-05
+ letab=8.2510e-06          wu0=2.4000e-04           pu0=-6.5000e-05
+ lub=-2.5220e-20           wub=-3.0000e-20          pub=-6.5270e-21
+ wuc=-5.5000e-12           pvsat=-7.3390e+02        lags=8.0000e-01
+ lketa=4.3920e-03          pketa=-5.0000e-04        ldelta=5.5800e-04
+ lvoffcv=-5.3220e-03       pkt1=1.0000e-03          lute=7.5240e-02
+ wute=2.5000e-02           pute=7.4000e-03          lub1=6.5000e-20
+ wuc1=-7.2000e-12          saref=1.7600e-06         sbref=1.7600e-06
+ wlod=0.0000e+00           kvth0=5.0000e-08         lkvth0=3.9000e-06
+ wkvth0=9.0000e-08         pkvth0=0.0000e+00        llodvth=1.0000e+00
+ wlodvth=1.0000e+00        stk2=0.0000e+00          lodk2=1.0000e+00
+ lodeta0=1.0000e+00        ku0=-1.5200e-08          lku0=-6.2900e-09
+ wku0=-1.0000e-08          pku0=1.2800e-15          llodku0=1.0500e+00
+ wlodku0=1.0000e+00        kvsat=9.9000e-01         steta0=-2.8000e-11
+ tku0=0.0000e+00
```

(ii) Typical Process Corner

```
// DEVICE 1  nmos

parameters dtoxp_n_10_sp=0.000000e+000
parameters dwint_n_10_sp=0.000000e+000
parameters dllpe0_n_10_sp=0.000000e+000
parameters dxl_n_10_sp=0.000000e+000
parameters drdsw_n_10_sp=0.000000e+000
parameters dcgdo_n_10_sp=0.000000e+000
parameters dcgsl_n_10_sp=0.000000e+000
parameters dcf_n_10_sp=0.000000e+000
parameters dcjs_n_10_sp=0.000000e+000
parameters dcjsws_n_10_sp=0.000000e+000
parameters dcjswgs_n_10_sp=0.000000e+000
parameters dtoxe_n_10_sp=0.000000e+000
parameters dvth0_n_10_sp=0.000000e+000
parameters dcgso_n_10_sp=0.000000e+000
parameters dvoffl_n_10_sp=0.000000e+000
parameters dcgdl_n_10_sp=0.000000e+000
parameters dpvth0_n_10_sp=0.000000e+000
parameters dlpe0_n_10_sp=0.000000e+000
parameters dxw_n_10_sp=0.000000e+000
parameters dk3_n_10_sp=0.000000e+000
parameters dwwl_n_10_sp=0.000000e+000
```

(iii) Slow Process Corner

```
// DEVICE 1  nmos

parameters dtoxp_n_10_sp=1.000000e-010
parameters dwint_n_10_sp=2.000000e-009
parameters dllpe0_n_10_sp=1.400000e-009
parameters dxl_n_10_sp=1.302000e-009
parameters drdsw_n_10_sp=1.700000e+001
parameters dcgdo_n_10_sp=-5.000000e-012
parameters dcgsl_n_10_sp=-2.200000e-011
parameters dcf_n_10_sp=-9.260000e-012
parameters dcjs_n_10_sp=1.070000e-004
parameters dcjsws_n_10_sp=1.260000e-011
parameters dcjswgs_n_10_sp=2.310000e-011
parameters dtoxe_n_10_sp=1.000000e-010
parameters dvth0_n_10_sp=3.300000e-002
parameters dcgso_n_10_sp=-5.000000e-012
parameters dvoffl_n_10_sp=4.500000e-010
parameters dcgdl_n_10_sp=-2.200000e-011
parameters dpvth0_n_10_sp=1.700000e-004
parameters dlpe0_n_10_sp=0.000000e+000
parameters dxw_n_10_sp=-5.000000e-009
parameters dk3_n_10_sp=2.200000e+000
parameters dwwl_n_10_sp=4.000000e-023
```

(iv) Fast Process Corner

```
// DEVICE 1 nmos

parameters dtoxp_n_10_sp=-1.000000e-010
```

```
parameters dwint_n_10_sp=0.000000e+000
parameters dllpe0_n_10_sp=-1.550000e-009
parameters dxl_n_10_sp=-1.000000e-010
parameters drdsw_n_10_sp=-1.000000e+001
parameters dcgdo_n_10_sp=5.000000e-012
parameters dcgsl_n_10_sp=2.200000e-011
parameters dcf_n_10_sp=9.260000e-012
parameters dcjs_n_10_sp=-1.070000e-004
parameters dcjsws_n_10_sp=-1.260000e-011
parameters dcjswgs_n_10_sp=-2.310000e-011
parameters dtoxe_n_10_sp=-1.000000e-010
parameters dvth0_n_10_sp=-3.300000e-002
parameters dcgso_n_10_sp=5.000000e-012
parameters dvoffl_n_10_sp=-1.053000e-009
parameters dcgdl_n_10_sp=2.200000e-011
parameters dpvth0_n_10_sp=-1.400000e-004
parameters dlpe0_n_10_sp=0.000000e+000
parameters dxw_n_10_sp=1.800000e-009
parameters dk3_n_10_sp=-2.510000e+000
parameters dwwl_n_10_sp=-1.200000e-023
```

(v) Monte Carlo Model

```
// DEVICE 1    nmos

model n bsim4 type=n
+ version=4.3000e+00       binunit=1.0000e+00       paramchk=1.0000e+00
+ mobmod=0.0000e+00        capmod=2.0000e+00        igcmod=1.0000e+00
+ igbmod=1.0000e+00        geomod=0.0000e+00        diomod=2.0000e+00
+ rdsmod=0.0000e+00        rbodymod=0.0000e+00
rgatemod=0.0000e+00
+ permod=1.0000e+00        acnqsmod=0.0000e+00
trnqsmod=0.0000e+00
+ rgeomod=1.0000e+00       fnoimod=1.0000e+00       tnoimod=0.0000e+00
+ toxe=2.2500e-09 + dtoxe_n_10_sp
+ toxp=1.8220e-09 + dtoxp_n_10_sp                   toxm=2.2500e-09
+ epsrox=3.9000e+00        xj=1.2000e-07            ngate=1.3000e+20
+ ndep=1.0000e+17          nsd=1.0000e+20           rsh=8.0000e+00
+ wint=2.0210e-08 + dwint_n_10_sp                   lint=-4.0910e-09
+ vth0= - 1.0000e-03 + dvth0_n_10_sp + p_vth0_ma_n/sqrt(mf)
k1=1.5690e-01
+ k2=4.0000e-03            k3= - 1.2880e+00 + dk3_n_10_sp
+ k3b=2.9280e+00           w0=9.0000e-08            dvt0=3.9630e+00
+ dvt1=5.6320e-01          dvt2=-3.3200e-02         dvt0w=5.2510e-01
+ dvt1w=1.1170e+07         dvt2w=-7.7000e-01        dsub=3.9000e-02
+ minv=7.7040e-01          voffl= - 4.9270e-09 + dvoffl_n_10_sp
+ dvtp0=8.9100e-06         dvtp1=-8.0630e-01
+ lpe0=1.0000e-10 + dlpe0_n_10_sp                   lpeb=-1.6990e-07
+ phin=8.7670e-02          cdsc=4.6490e-04          cdscb=1.5000e-04
+ cdscd=0.0000e+00         cit=1.5520e-03           voff=-6.3870e-02
+ nfactor=1.0000e-01       eta0=5.0000e-05          etab=-1.8500e-04
+ vfb=-1.0000e+00          u0=2.3200e-02 * (1 - p_u0_ma_n/sqrt(mf))
+ ua=-1.5500e-09           ub=3.4800e-18            uc=1.7330e-10
+ vsat=1.6250e+05          a0=8.8340e+00            ags=1.0020e+00
+ a1=0.0000e+00            a2=1.0000e+00            b0=0.0000e+00
+ b1=1.0000e-08            keta=-4.4080e-02         dwg=-5.4000e-09
+ dwb=4.8000e-09           pclm=1.0000e-01          pdiblc1=1.0000e-07
+ pdiblc2=3.9540e-02       pdiblcb=1.0000e-01       drout=5.5990e-01
+ pvag=8.6180e+00          delta=1.0000e-02         pscbe1=6.5350e+09
+ pscbe2=3.3000e-01        fprout=1.0000e-02        pdits=6.1100e-01
```

```
+ pditsd=8.8000e-01        pditsl=1.0000e+05
+ rdsw=5.0000e+01 + drdsw_n_10_sp                  rdswmin=5.0000e+01
+ prwg=2.8000e-01          prwb=4.4700e-01          wr=1.0000e+00
+ alpha0=2.0000e-07        alpha1=4.0000e+00        beta0=1.5000e+01
+ agidl=1.1080e-08         bgidl=1.3900e+09         cgidl=2.9630e-01
+ egidl=9.4400e-01         toxref=2.2500e-09        dlcig=1.8000e-08
+ aigbacc=1.1980e-02       bigbacc=8.0130e-03       cigbacc=6.2560e-01
+ nigbacc=4.3970e+00       aigbinv=1.5300e-02       bigbinv=4.8520e-03
+ cigbinv=1.0000e-03       eigbinv=1.1000e+00       nigbinv=1.6000e+00
+ aigc=1.1380e-02          bigc=1.8790e-03          cigc=1.0000e-04
+ aigsd=9.8830e-03         bigsd=1.2690e-03         cigsd=1.5540e-01
+ nigc=1.0000e+00          poxedge=1.0000e+00       pigcd=2.5000e+00
+ ntox=1.0000e+00          dlc=1.6400e-08           dwc=-3.0000e-08
+ xpart=1.0000e+00         cgso=5.0000e-11 + dcgso_n_10_sp
+ cgdo=5.0000e-11 + dcgdo_n_10_sp                  cgbo=0.0000e+00
+ cgdl=2.2000e-10 + dcgdl_n_10_sp
+ cgsl=2.2000e-10 + dcgsl_n_10_sp                  clc=1.0000e-07
+ cle=6.0000e-01           cf=9.2600e-11 + dcf_n_10_sp
+ ckappas=3.0000e+00       vfbcv=-1.0000e+00        acde=2.8080e-01
+ moin=1.1830e+01          noff=2.4860e+00          voffcv=-1.3720e-02
+ ef=0.9448                noia=3.8700000e+41       noib=1.8600000e+25
+ noic=6.7000000e+08       em=6.3600000e+06         ntnoi=1.0
+ xl= - 1.0000e-08 + dxl_n_10_sp
+ xw=0.0000e+00 + dxw_n_10_sp                      dmcg=1.6000e-07
+ dmci=1.0000e-07          dwj=0.0000e+00           jss=2.3350e-07
+ jsws=7.0330e-14          jswgs=3.2986e-14         ijthsfwd=3.4450e-03
+ ijthsrev=1.6910e-03      bvs=1.1470e+01           xjbvs=1.0000e+00
+ pbs=6.1000e-01           cjs=1.0700e-03 + dcjs_n_10_sp
+ mjs=2.9000e-01           pbsws=9.9000e-01
+ cjsws=1.2600e-10 + dcjsws_n_10_sp                mjsws=1.0000e-01
+ pbswgs=6.0000e-01        cjswgs=2.3100e-10 + dcjswgs_n_10_sp
+ mjswgs=9.8900e-01        tnom=2.5000e+01          kt1=-3.8000e-01
+ kt1l=1.0000e-09          kt2=-4.0740e-02          ute=-1.0220e+00
+ ua1=4.3500e-09           ub1=-4.1040e-18          uc1=2.6360e-10
+ prt=0.0000e+00           at=1.0000e+05            njs=1.0560e+00
+ tpb=1.3000e-03           tcj=9.0000e-04           tpbsw=3.5150e-03
+ tcjsw=4.0000e-04         tpbswg=1.2470e-03        tcjswg=8.2290e-03
+ xtis=3.0000e+00          ll=4.3480e-16            wl=-4.0050e-15
+ lln=9.0000e-01           wln=1.0000e+00           lw=3.2080e-15
+ ww=-1.5010e-15           lwn=1.0000e+00           wwn=1.0000e+00
+ lwl=-1.6220e-21          wwl=1.7820e-22 + dwwl_n_10_sp
+ llc=-9.0100e-16          wlc=0.0000e+00           lwc=0.0000e+00
+ wwc=1.0000e-15           lwlc=0.0000e+00          wwlc=0.0000e+00
+ lmin=8.0000e-08          lmax=5.0000e-05          wmin=1.2000e-07
+ wmax=1.0000e-04          pvth0= - 1.2500e-03 + dpvth0_n_10_sp
+ lk3=7.2000e-01           wk3=-1.3000e-01          lk3b=-2.0000e-01
+ pk3b=2.0000e-02          ldsub=-1.2720e-03        wdsub=5.0000e-04
+ llpe0=3.8910e-08 + dllpe0_n_10_sp                lcit=7.0000e-05
+ wvoff=-1.3400e-03        leta0=1.3000e-05         weta0=3.7760e-05
+ letab=8.2510e-06         wu0=2.4000e-04 * (1 - p_u0_ma_n/sqrt(mf))
+ pu0= - 6.5000e-05 * (1 - p_u0_ma_n/sqrt(mf))              lub=-2.5220e-20
+ wub=-3.0000e-20          pub=-6.5270e-21          wuc=-5.5000e-12
+ pvsat=-7.3390e+02        lags=8.0000e-01          lketa=4.3920e-03
+ pketa=-5.0000e-04        ldelta=5.5800e-04        lvoffcv=-5.3220e-03
+ pkt1=1.0000e-03          lute=7.5240e-02          wute=2.5000e-02
+ pute=7.4000e-03          lub1=6.5000e-20          wuc1=-7.2000e-12
+ saref=1.7600e-06         sbref=1.7600e-06         wlod=0.0000e+00
+ kvth0=5.0000e-08         lkvth0=3.9000e-06        wkvth0=9.0000e-08
+ pkvth0=0.0000e+00        llodvth=1.0000e+00       wlodvth=1.0000e+00
+ stk2=0.0000e+00          lodk2=1.0000e+00         lodeta0=1.0000e+00
+ ku0=-1.5200e-08          lku0=-6.2900e-09         wku0=-1.0000e-08
+ pku0=1.2800e-15          llodku0=1.0500e+00       wlodku0=1.0000e+00
+ kvsat=9.9000e-01         steta0=-2.8000e-11       tku0=0.0000e+00
```

C.2 PMOS Model

(i) Nominal model

```
// DEVICE 2

model p_10_sp bsim4 type=p
+ version=4.3000e+00       binunit=1.0000e+00       paramchk=1.0000e+00
+ mobmod=0.0000e+00        capmod=2.0000e+00        igcmod=1.0000e+00
+ igbmod=1.0000e+00        geomod=0.0000e+00        diomod=2.0000e+00
+ rdsmod=0.0000e+00        rbodymod=0.0000e+00      permod=1.0000e+00
+ acnqsmod=0.0000e+00      rgeomod=1.0000e+00       fnoimod=1.0000e+00
+ tnoimod=0.0000e+00       toxe=2.4500e-09 + dtoxe_p_10_sp
+ toxp=1.9110e-09 + dtoxp_p_10_sp                  toxm=2.4500e-09
+ epsrox=3.9000e+00        xj=1.2000e-07            ngate=1.0000e+20
+ ndep=3.6000e+16          nsd=1.0000e+20           rsh=8.0000e+00
+ wint=8.0090e-09 + dwint_p_10_sp                  lint=-2.1220e-08
+ vth0= - 5.8100e-02 + dvth0_p_10_sp               k1=2.2500e-01
+ k2=-2.4750e-02           k3= - 8.8950e+00 + dk3_p_10_sp
+ k3b=3.9000e+00           w0=2.1220e-06            dvt0=4.6860e+00
+ dvt1=8.7290e-01          dvt2=1.2770e-02          dvt0w=3.0000e-01
+ dvt1w=3.9660e+06         dvt2w=2.4940e-01         dsub=1.0160e+00
+ minv=2.8230e-01          voffl= - 2.5000e-09 + dvoffl_p_10_sp
+ dvtp0=6.0620e-06         dvtp1=4.4890e-01
+ lpe0= - 1.2670e-07 + dlpe0_p_10_sp               lpeb=6.2500e-08
+ phin=0.0000e+00          cdsc=0.0000e+00          cdscb=-8.0000e-03
+ cdscd=0.0000e+00         cit=2.7750e-04           voff=-1.2000e-01
+ nfactor=2.0000e+00       eta0=3.0000e-02          etab=-5.0310e-01
+ vfb=-1.0000e+00          u0=9.2600e-03 + du0_p_10_sp
+ ua=4.2790e-10            ub=1.1290e-18            uc=8.5910e-11
+ eu=1.0000e+00            vsat=1.3670e+05          a0=1.8600e+00
+ ags=1.4670e+00           a1=0.0000e+00            a2=1.0000e+00
+ b0=7.0000e-07            b1=6.0000e-07            keta=-5.1120e-02
+ dwg=-1.7240e-08          dwb=0.0000e+00           pclm=2.9400e-01
+ pdiblc1=5.1850e-08       pdiblc2=4.0800e-03       pdiblcb=-5.0000e-01
+ drout=4.6980e-04         pvag=1.2960e+00          delta=2.3890e-03
+ pscbe1=6.3370e+09        pscbe2=3.0000e-03        fprout=3.0000e+02
+ pdits=2.9810e-01         pditsd=7.1760e-01        pditsl=5.0000e+05
+ rdsw=2.2500e+02 + drdsw_p_10_sp                  rdswmin=8.0000e+01
+ prwg=0.0000e+00          prwb=2.0000e-01          wr=1.0000e+00
+ alpha0=2.1400e-08        alpha1=7.0000e-02        beta0=1.2000e+01
+ agidl=4.4320e-09         bgidl=4.8080e+09         cgidl=9.1730e-03
+ egidl=-2.1800e+00        toxref=2.4500e-09        dlcig=3.2000e-08
+ aigbacc=1.1030e-02       bigbacc=6.7610e-03       cigbacc=5.7700e-01
+ nigbacc=4.3960e+00       aigbinv=9.4660e-03       bigbinv=2.3400e-03
+ cigbinv=1.8320e-03       eigbinv=1.6330e+00       nigbinv=3.1240e+00
+ aigc=6.7900e-03          bigc=8.8750e-04          cigc=6.3430e-04
+ aigsd=5.6520e-03         bigsd=7.8050e-05         cigsd=1.8030e-02
+ nigc=7.9250e-01          poxedge=1.0000e+00       pigcd=2.0000e+00
+ ntox=1.0000e+00          dlc=3.4200e-08           dwc=-3.0000e-08
+ xpart=1.0000e+00         cgso=4.2000e-11 + dcgso_p_10_sp
+ cgdo=4.2000e-11 + dcgdo_p_10_sp                  cgbo=0.0000e+00
+ cgdl=2.0000e-10 + dcgdl_p_10_sp
+ cgsl=2.0000e-10 + dcgsl_p_10_sp                  clc=1.0000e-07
+ cle=6.0000e-01           cf=9.0800e-11 + dcf_p_10_sp
+ ckappas=7.3000e-01       ckappad=7.3000e-01       acde=3.5090e-01
+ moin=6.7000e+00          noff=2.9360e+00          voffcv=-5.2570e-02
+ ef=1.103336
+ noia=1.0635922e+41       noib=6.9613951e+26       noic=5.2897264e+09
+ em=4.1000000e+07         ntnoi=1.0
+ xl= - 1.0000e-08 + dxl_p_10_sp
+ xw=0.0000e+00 + dxw_p_10_sp                      dmcg=1.6000e-07
+ dmci=1.0000e-07          dwj=0.0000e+00           jss=1.9950e-07
```

```
+ jsws=1.0920e-13          jswgs=1.0000e-13          ijthsfwd=3.5000e-03
+ ijthsrev=2.1750e-03      bvs=8.9640e+00            xjbvs=1.0000e+00
+ pbs=7.3000e-01           cjs=1.2600e-03 + dcjs_p_10_sp
+ mjs=3.1000e-01           pbsws=9.9000e-01
+ cjsws=1.2900e-10 + dcjsws_p_10_sp                  mjsws=1.0000e-01
+ pbswgs=6.0000e-01        cjswgs=2.4500e-10 + dcjswgs_p_10_sp
+ mjswgs=9.8900e-01        tnom=2.5000e+01           kt1=-3.4000e-01
+ kt1l=-9.5660e-09         kt2=-1.0000e-02           ute=-1.9620e+00
+ ua1=-8.3500e-10          ub1=-1.3400e-18           uc1=0.0000e+00
+ prt=-1.6750e+02          at=1.0340e+05             njs=1.0540e+00
+ tpb=1.4000e-03           tcj=8.0000e-04            tpbsw=1.0000e-04
+ tcjsw=4.0000e-04         tpbswg=1.5050e-03         tcjswg=7.6180e-03
+ xtis=3.0000e+00          ll=5.5440e-16             wl=7.1650e-16
+ lln=1.0500e+00           wln=9.7350e-01            lw=-2.1170e-15
+ ww=-4.3920e-15           lwn=1.0000e+00            wwn=9.9400e-01
+ lwl=2.3760e-23           wwl= - 1.4950e-22 + dwwl_p_10_sp
+ llc=-6.7780e-16          wlc=0.0000e+00            lwc=0.0000e+00
+ wwc=1.0000e-15           lwlc=0.0000e+00           wwlc=0.0000e+00
+ lmin=8.0000e-08          lmax=5.0000e-05           wmin=1.2000e-07
+ wmax=1.0000e-04          pvth0=0.0000e+00 + dpvth0_p_10_sp
+ lk3=1.0000e+00           pk3=-1.4700e-01           lk3b=-7.6900e-01
+ wk3b=2.1290e+00          wdsub=1.1010e-02          pdvtp1=-2.0000e-02
+ llpe0=2.9370e-08 + dllpe0_p_10_sp                  llpeb=1.3590e-08
+ lnfactor=2.6600e-01      letab=3.9610e-02          lags=1.8360e+00
+ pags=-8.4000e-02         lb0=-5.5000e-08           lb1=-5.8900e-08
+ lketa=-1.9200e-03        ldelta=2.4950e-03         wrdsw=1.0000e+01
+ lvoffcv=4.1250e-04       wkt1=5.0000e-03           wua1=3.9440e-11
+ saref=1.7600e-06         sbref=1.7600e-06          wlod=0.0000e+00
+ kvth0=-8.0000e-10        lkvth0=-1.5000e-06        wkvth0=6.0000e-07
+ pkvth0=0.0000e+00        llodvth=8.0000e-01        wlodvth=1.0000e+00
+ stk2=0.0000e+00          lodk2=1.0000e+00          lodeta0=1.0000e+00
+ ku0=5.3000e-07           lku0=5.8000e-04           wku0=-1.1000e-09
+ pku0=-2.5000e-10         llodku0=6.8000e-01        wlodku0=8.5000e-01
+ kvsat=1.0000e+00         steta0=3.8000e-10         tku0=0.0000e+00
```

(ii) Typical Process Corner

```
// DEVICE 2    pmos

parameters dwwl_p_10_sp=0.000000e+000
parameters dtoxp_p_10_sp=0.000000e+000
parameters dwint_p_10_sp=0.000000e+000
parameters dllpe0_p_10_sp=0.000000e+000
parameters dxl_p_10_sp=0.000000e+000
parameters drdsw_p_10_sp=0.000000e+000
parameters dcgdo_p_10_sp=0.000000e+000
parameters dcgsl_p_10_sp=0.000000e+000
parameters dcf_p_10_sp=0.000000e+000
parameters dcjsws_p_10_sp=0.000000e+000
parameters dtoxe_p_10_sp=0.000000e+000
parameters dcjs_p_10_sp=0.000000e+000
parameters dcjswgs_p_10_sp=0.000000e+000
parameters dvth0_p_10_sp=0.000000e+000
parameters du0_p_10_sp=0.000000e+000
parameters dcgso_p_10_sp=0.000000e+000
parameters dvoffl_p_10_sp=0.000000e+000
parameters dcgdl_p_10_sp=0.000000e+000
parameters dpvth0_p_10_sp=0.000000e+000
parameters dxw_p_10_sp=0.000000e+000
parameters dk3_p_10_sp=0.000000e+000
parameters dlpe0_p_10_sp=0.000000e+000
```

(iii) Slow Process Corner

```
// DEVICE 2   pmos

parameters dwwl_p_10_sp=5.720000e-023
parameters dtoxp_p_10_sp=1.000000e-010
parameters dwint_p_10_sp=2.500000e-009
parameters dllpe0_p_10_sp=2.600000e-009
parameters dxl_p_10_sp=1.000000e-009
parameters drdsw_p_10_sp=1.000000e+001
parameters dcgdo_p_10_sp=-4.200000e-012
parameters dcgsl_p_10_sp=-2.000000e-011
parameters dcf_p_10_sp=-9.080000e-012
parameters dcjsws_p_10_sp=1.290000e-011
parameters dtoxe_p_10_sp=1.000000e-010
parameters dcjs_p_10_sp=1.260000e-004
parameters dcjswgs_p_10_sp=2.450000e-011
parameters dvth0_p_10_sp=-3.000000e-002
parameters du0_p_10_sp=0.000000e+000
parameters dcgso_p_10_sp=-4.200000e-012
parameters dvoffl_p_10_sp=-1.170000e-009
parameters dcgdl_p_10_sp=-2.000000e-011
parameters dpvth0_p_10_sp=0.000000e+000
parameters dxw_p_10_sp=-3.000000e-009
parameters dk3_p_10_sp=3.051000e+001
parameters dlpe0_p_10_sp=0.000000e+000
```

(iv) Fast Process Corner

```
// DEVICE 2 pmos

parameters dwwl_p_10_sp=-4.200000e-023
parameters dtoxp_p_10_sp=-1.000000e-010
parameters dwint_p_10_sp=0.000000e+000
parameters dllpe0_p_10_sp=0.000000e+000
parameters dxl_p_10_sp=-3.337000e-009
parameters drdsw_p_10_sp=0.000000e+000
parameters dcgdo_p_10_sp=4.200000e-012
parameters dcgsl_p_10_sp=2.000000e-011
parameters dcf_p_10_sp=9.080000e-012
parameters dcjsws_p_10_sp=-1.290000e-011
parameters dtoxe_p_10_sp=-1.000000e-010
parameters dcjs_p_10_sp=-1.260000e-004
parameters dcjswgs_p_10_sp=-2.450000e-011
parameters dvth0_p_10_sp=2.360000e-002
parameters du0_p_10_sp=0.000000e+000
parameters dcgso_p_10_sp=4.200000e-012
parameters dvoffl_p_10_sp=-2.565000e-009
parameters dcgdl_p_10_sp=2.000000e-011
parameters dpvth0_p_10_sp=1.215000e-004
parameters dxw_p_10_sp=3.000000e-009
parameters dk3_p_10_sp=-3.100000e+001
parameters dlpe0_p_10_sp=1.600000e-009
```

(v) Monte Carlo Model

```
// DEVICE 2  pmos

model p bsim4 type=p
+ version=4.3000e+00       binunit=1.0000e+00       paramchk=1.0000e+00
+ mobmod=0.0000e+00        capmod=2.0000e+00        igcmod=1.0000e+00
+ igbmod=1.0000e+00        geomod=0.0000e+00        diomod=2.0000e+00
+ rdsmod=0.0000e+00        rbodymod=0.0000e+00      permod=1.0000e+00
+ acnqsmod=0.0000e+00      rgeomod=1.0000e+00       fnoimod=1.0000e+00
+ tnoimod=0.0000e+00       toxe=2.4500e-09 + dtoxe_p_10_sp
+ toxp=1.9110e-09 + dtoxp_p_10_sp                  toxm=2.4500e-09
+ epsrox=3.9000e+00        xj=1.2000e-07            ngate=1.0000e+20
+ ndep=3.6000e+16          nsd=1.0000e+20           rsh=8.0000e+00
+ wint=8.0090e-09 + dwint_p_10_sp                  lint=-2.1220e-08
+ vth0= - 5.8100e-02 + dvth0_p_10_sp + p_vth0_ma_p/sqrt(mf)
k1=2.2500e-01
+ k2=-2.4750e-02          k3= - 8.8950e+00 + dk3_p_10_sp
+ k3b=3.9000e+00          w0=2.1220e-06            dvt0=4.6860e+00
+ dvt1=8.7290e-01         dvt2=1.2770e-02          dvt0w=3.0000e-01
+ dvt1w=3.9660e+06        dvt2w=2.4940e-01         dsub=1.0160e+00
+ minv=2.8230e-01         voffl= - 2.5000e-09 + dvoffl_p_10_sp
+ dvtp0=6.0620e-06        dvtp1=4.4890e-01
+ lpe0= - 1.2670e-07 + dlpe0_p_10_sp               lpeb=6.2500e-08
+ phin=0.0000e+00         cdsc=0.0000e+00          cdscb=-8.0000e-03
+ cdscd=0.0000e+00        cit=2.7750e-04           voff=-1.2000e-01
+ nfactor=2.0000e+00      eta0=3.0000e-02          etab=-5.0310e-01
+ vfb=-1.0000e+00         u0=(9.2600e-03 + du0_p_10_sp) * (1 -
p_u0_ma_p/sqrt(mf))
+ ua=4.2790e-10           ub=1.1290e-18            uc=8.5910e-11
+ eu=1.0000e+00           vsat=1.3670e+05          a0=1.8600e+00
+ ags=1.4670e+00          a1=0.0000e+00            a2=1.0000e+00
+ b0=7.0000e-07           b1=6.0000e-07            keta=-5.1120e-02
+ dwg=-1.7240e-08         dwb=0.0000e+00           pclm=2.9400e-01
+ pdiblc1=5.1850e-08      pdiblc2=4.0800e-03       pdiblcb=-5.0000e-01
+ drout=4.6980e-04        pvag=1.2960e+00          delta=2.3890e-03
+ pscbe1=6.3370e+09       pscbe2=3.0000e-03        fprout=3.0000e+02
+ pdits=2.9810e-01        pditsd=7.1760e-01        pditsl=5.0000e+05
+ rdsw=2.2500e+02 + drdsw_p_10_sp                  rdswmin=8.0000e+01
+ prwg=0.0000e+00         prwb=2.0000e-01          wr=1.0000e+00
+ alpha0=2.1400e-08       alpha1=7.0000e-02        beta0=1.2000e+01
+ agidl=4.4320e-09        bgidl=4.8080e+09         cgidl=9.1730e-03
+ egidl=-2.1800e+00       toxref=2.4500e-09        dlcig=3.2000e-08
+ aigbacc=1.1030e-02      bigbacc=6.7610e-03       cigbacc=5.7700e-01
+ nigbacc=4.3960e+00      aigbinv=9.4660e-03       bigbinv=2.3400e-03
+ cigbinv=1.8320e-03      eigbinv=1.6330e+00       nigbinv=3.1240e+00
+ aigc=6.7900e-03         bigc=8.8750e-04          cigc=6.3430e-04
+ aigsd=5.6520e-03        bigsd=7.8050e-05         cigsd=1.8030e-02
+ nigc=7.9250e-01         poxedge=1.0000e+00       pigcd=2.0000e+00
+ ntox=1.0000e+00         dlc=3.4200e-08           dwc=-3.0000e-08
+ xpart=1.0000e+00        cgso=4.2000e-11 + dcgso_p_10_sp
+ cgdo=4.2000e-11 + dcgdo_p_10_sp                  cgbo=0.0000e+00
+ cgdl=2.0000e-10 + dcgdl_p_10_sp
+ cgsl=2.0000e-10 + dcgsl_p_10_sp                  clc=1.0000e-07
+ cle=6.0000e-01          cf=9.0800e-11 + dcf_p_10_sp
+ ckappas=7.3000e-01      ckappad=7.3000e-01       acde=3.5090e-01
+ moin=6.7000e+00         noff=2.9360e+00          voffcv=-5.2570e-02
+ ef=1.103336             noia=1.0635922e+41       noib=6.9613951e+26
+ noic=5.2897264e+09      em=4.1000000e+07         ntnoi=1.0
+ xl= - 1.0000e-08 + dxl_p_10_sp
+ xw=0.0000e+00 + dxw_p_10_sp                      dmcg=1.6000e-07
+ dmci=1.0000e-07         dwj=0.0000e+00           jss=1.9950e-07
+ jsws=1.0920e-13         jswgs=1.0000e-13         ijthsfwd=3.5000e-03
```

```
+ ijthsrev=2.1750e-03      bvs=8.9640e+00          xjbvs=1.0000e+00
+ pbs=7.3000e-01           cjs=1.2600e-03 + dcjs_p_10_sp
+ mjs=3.1000e-01           pbsws=9.9000e-01
+ cjsws=1.2900e-10 + dcjsws_p_10_sp                mjsws=1.0000e-01
+ pbswgs=6.0000e-01        cjswgs=2.4500e-10 + dcjswgs_p_10_sp
+ mjswgs=9.8900e-01        tnom=2.5000e+01         kt1=-3.4000e-01
+ kt1l=-9.5660e-09         kt2=-1.0000e-02         ute=-1.9620e+00
+ ua1=-8.3500e-10          ub1=-1.3400e-18         uc1=0.0000e+00
+ prt=-1.6750e+02          at=1.0340e+05           njs=1.0540e+00
+ tpb=1.4000e-03           tcj=8.0000e-04          tpbsw=1.0000e-04
+ tcjsw=4.0000e-04         tpbswg=1.5050e-03       tcjswg=7.6180e-03
+ xtis=3.0000e+00          ll=5.5440e-16           wl=7.1650e-16
+ lln=1.0500e+00           wln=9.7350e-01          lw=-2.1170e-15
+ ww=-4.3920e-15           lwn=1.0000e+00          wwn=9.9400e-01
+ lwl=2.3760e-23           wwl= - 1.4950e-22 + dwwl_p_10_sp
+ llc=-6.7780e-16          wlc=0.0000e+00          lwc=0.0000e+00
+ wwc=1.0000e-15           lwlc=0.0000e+00         wwlc=0.0000e+00
+ lmin=8.0000e-08          lmax=5.0000e-05         wmin=1.2000e-07
+ wmax=1.0000e-04          pvth0=0.0000e+00 + dpvth0_p_10_sp
+ lk3=1.0000e+00           pk3=-1.4700e-01         lk3b=-7.6900e-01
+ wk3b=2.1290e+00          wdsub=1.1010e-02        pdvtp1=-2.0000e-02
+ llpe0=2.9370e-08 + dllpe0_p_10_sp                llpeb=1.3590e-08
+ lnfactor=2.6600e-01      letab=3.9610e-02        lags=1.8360e+00
+ pags=-8.4000e-02         lb0=-5.5000e-08         lb1=-5.8900e-08
+ lketa=-1.9200e-03        ldelta=2.4950e-03       wrdsw=1.0000e+01
+ lvoffcv=4.1250e-04       wkt1=5.0000e-03         wua1=3.9440e-11
+ saref=1.7600e-06         sbref=1.7600e-06        wlod=0.0000e+00
+ kvth0=-8.0000e-10        lkvth0=-1.5000e-06      wkvth0=6.0000e-07
+ pkvth0=0.0000e+00        llodvth=8.0000e-01      wlodvth=1.0000e+00
+ stk2=0.0000e+00          lodk2=1.0000e+00        lodeta0=1.0000e+00
+ ku0=5.3000e-07           lku0=5.8000e-04         wku0=-1.1000e-09
+ pku0=-2.5000e-10         llodku0=6.8000e-01      wlodku0=8.5000e-01
+ kvsat=1.0000e+00         steta0=3.8000e-10       tku0=0.0000e+00
```

C.3 UMC 90nm Technology Overview [6]

UMC L90N 1P9M 1.0V/2.5V lowK Logic/MixedMode							
Process technology specifications	**units**	**Standard Performance (SP)**			**Low Leakage(LL)**		
Application		**ASIC - Consumer - Network**			**Portable - Wireless**		
		LVT	**RVT**	**HVT**	**LVT**	**RVT**	**HVT**
Substrate Type		P-substrate					
Nwell-Non salicide (N+P+N+PolyP+Poly)	Ohm/sq	370-8295100240					
Wafer size (6) - available die thicknesses		12 Inch-29 Mils or 11 Mils					
Core devices		SP_Lvt (1.0V)	SP_Rvt (1.0V)	SP_Hvt (1.0V)	LL_Lvt (1.2V)	LL_Rvt (1.2V)	LL_Hvt (1.2V)
Core devices Tox-Min gate length	Å-µm	15.5 - 0.08	15.5 - 0.08	15.5 -0.08	22-0.09	22-0.09	22-0.09
Core devices Ioff	Amp/um	50n	10n	400p	400p	30p	10p
Core devices Delay	ps/stage	9.8	10.6	16.1	15.5	20.5	21.3
Core devices VtON N/P	V	0.26/-0.22	0.33/-0.277	0.457/-0.39	0.49/-0.394	0.562/-0.502	0.648/-0.54
Core device overdrive (OD) feasibility	V	1.2V	1.2V	1.2V	-	-	-
Core device overdrive (OD) Ioff	Amp/um N/P	60n/100n	5n/12n	400p/600p	-	-	-
Core device overdrive (OD) Delay	ps/stage	7.7	8.6	11.9	-	-	-
Core device overdrive (OD) VtSAT N/P	V	0.137/-0.09	0.227/-0.167	0.362/-0.287	-	-	-
IO devices	V	1.8V2.5V(default)3.3V					
IO devices Tox_gl (VG=-2V, VB=0V)-Min gate length	Å-µm	31-0.1852-0.2465-0.34					
IO devices Ioff	Amp/um N/P	10p/400p15p/15p10p/10p					
IO devices Delay	ps/stage	2624.739.4					
IO devices VtON N/P	V	0.527/-0.4130.548/-0.50.57/-0.566					
IO device underdrive (UD) feasibilty		1.8V at Gox52Ioff N/P 8p/8pdelay 34.5 ps/stageVtsat N/P 0.462/-0.432min gate length 0.4µ					
IO device overdrive (OD) feasibility		3.3V at Gox52Ioff N/P 15p/52pdelay 70 ps/stageVtsat N/P0.45/-0.436min gate length 0.7µ					
High Ohmic Resistor (HR)	Ohm/sq	1012					
Metal Metal Cap (MiM cap)	fF/µm²	1.544					
NCAP	fF/µm²	15.3 @ 1.0V thin oxide-11.7 @1.2V medium oxide-8.9 @1.8V thick oxide-5.8 @2.5V thick oxide-4.8 @3.3V thick oxide					
Native threshold voltage NFET		SP_NVT 1.0_1.2V ODLL_NVT 1.2VNVT 1.8VNVT 2.5VNVT 3.3V					
Number of Poly/Metal Layers	#	1 Poly - 9 Metals :M1M2->M6(1X)-M7->M8(2X)-M9(4X)					
Metal pitch	µm	M1(0.12)M2->M6(0.14)-M7->M8(0.28)-M9(0.56)					
Metal Resistivity	mOhm/sq	M1(115)M2->M6(105)-M7->M8(44)-M9(27)					
Cadence Design Kit		Yes					

Appendix D MUTEX Circuits Simulation Results

This appendix presents the simulation results of the circuits in Chapter 4.

D.1 Latch Size

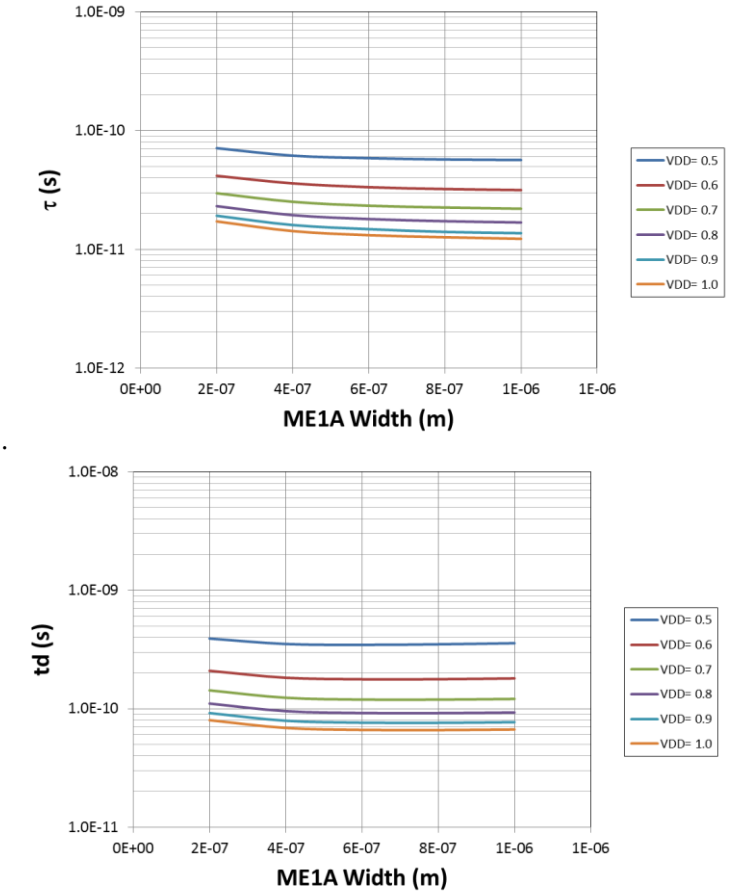


Figure D.1 Impact of Latch size on τ and t_d (ME1A)

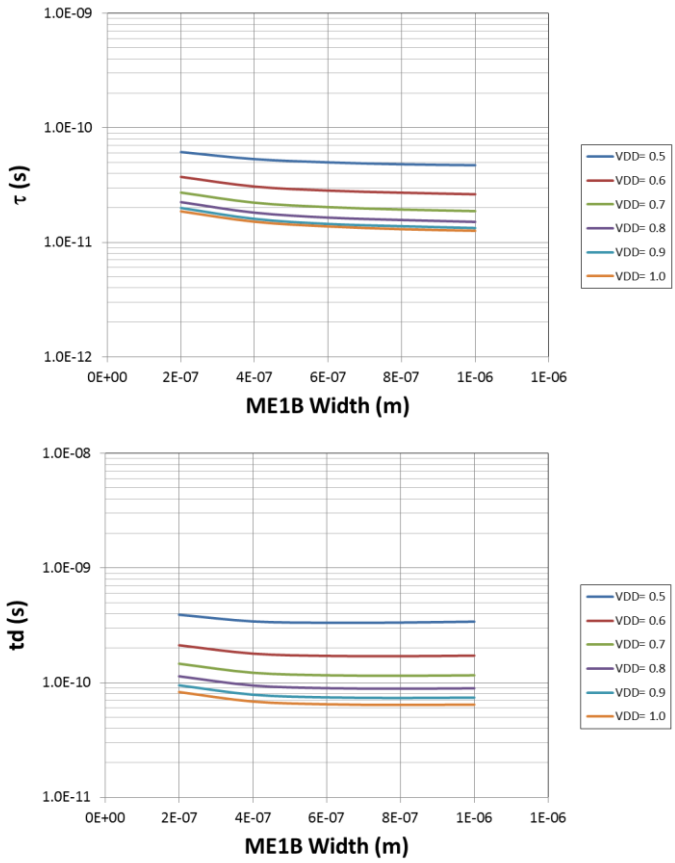


Figure D.2 Impact of Latch size on τ and t_d (ME1B)

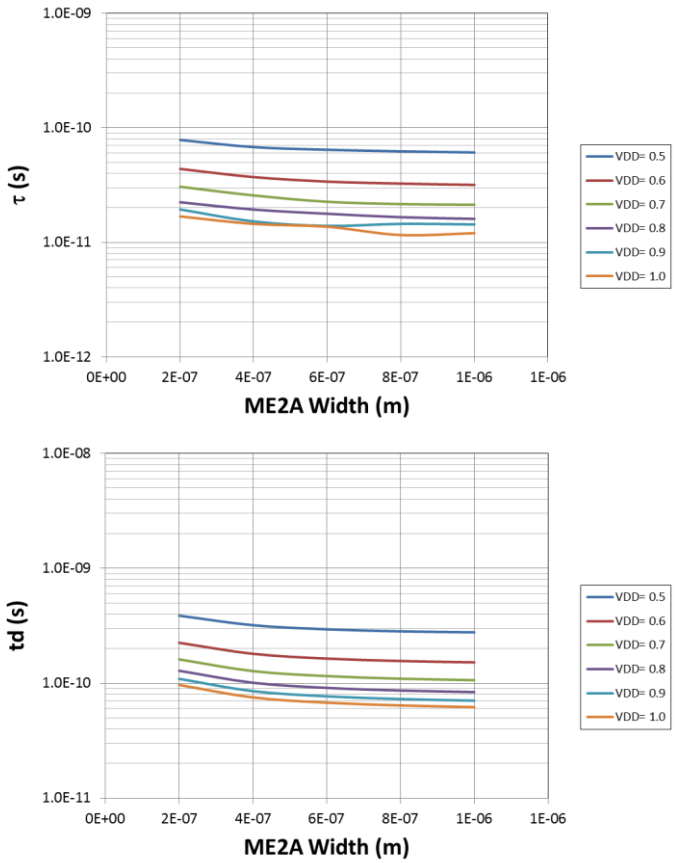


Figure D.3 Impact of Latch size on τ and t_d (ME2A)

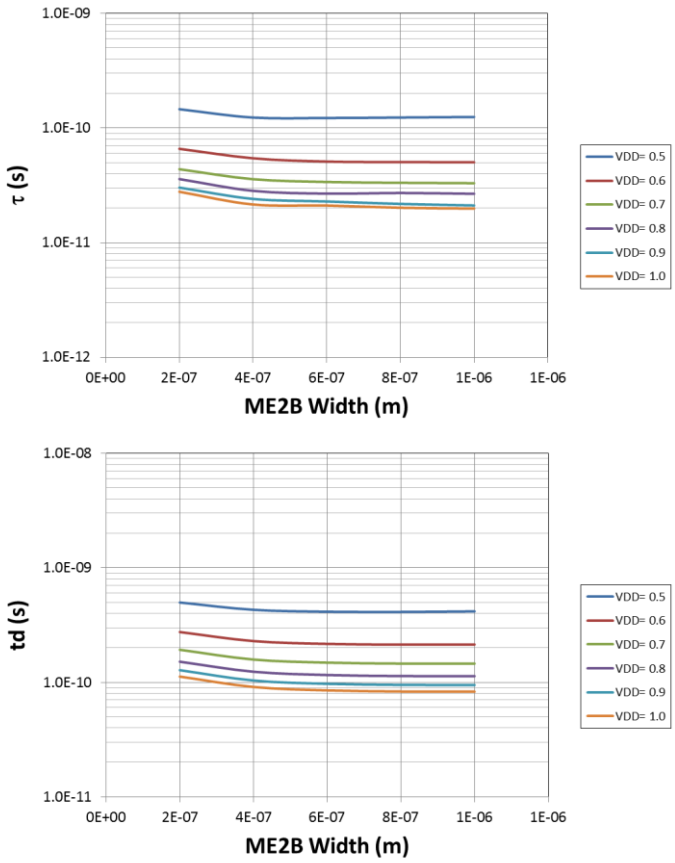


Figure D.4 Impact of Latch size on τ and t_d (ME2B)

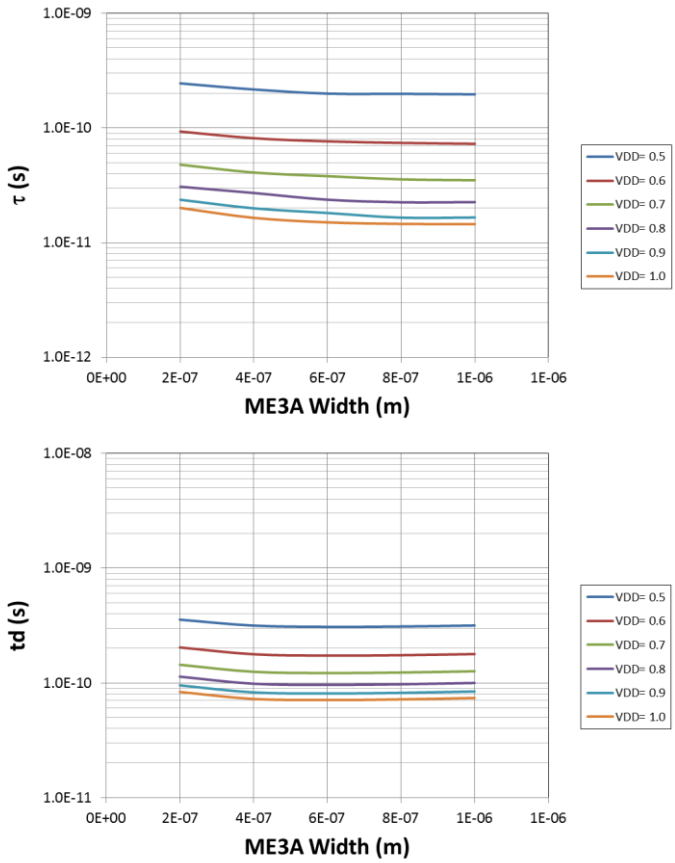


Figure D.5 Impact of Latch size on τ and t_d (ME3A)

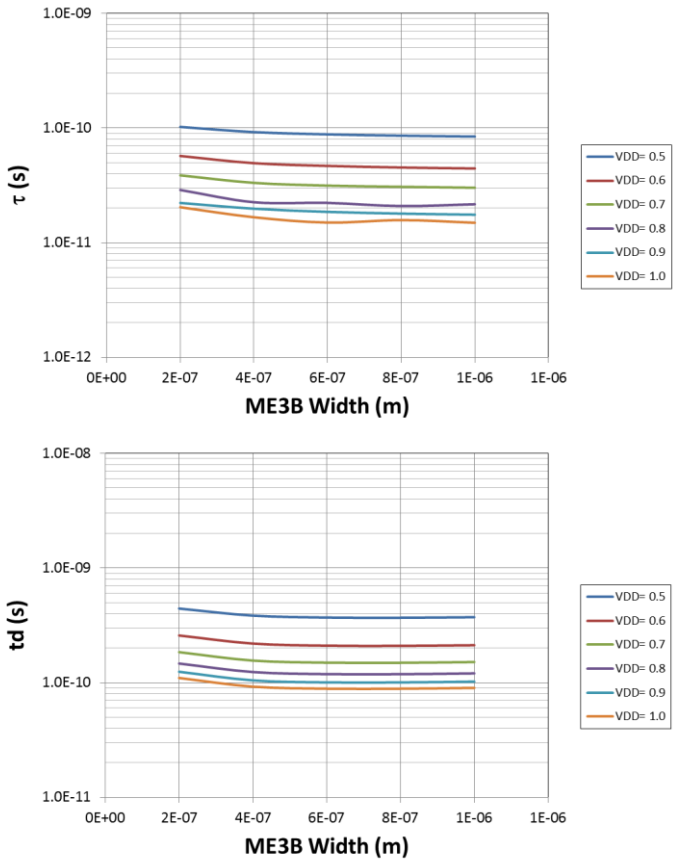


Figure D.6 Impact of Latch size on τ and t_d (ME3B)

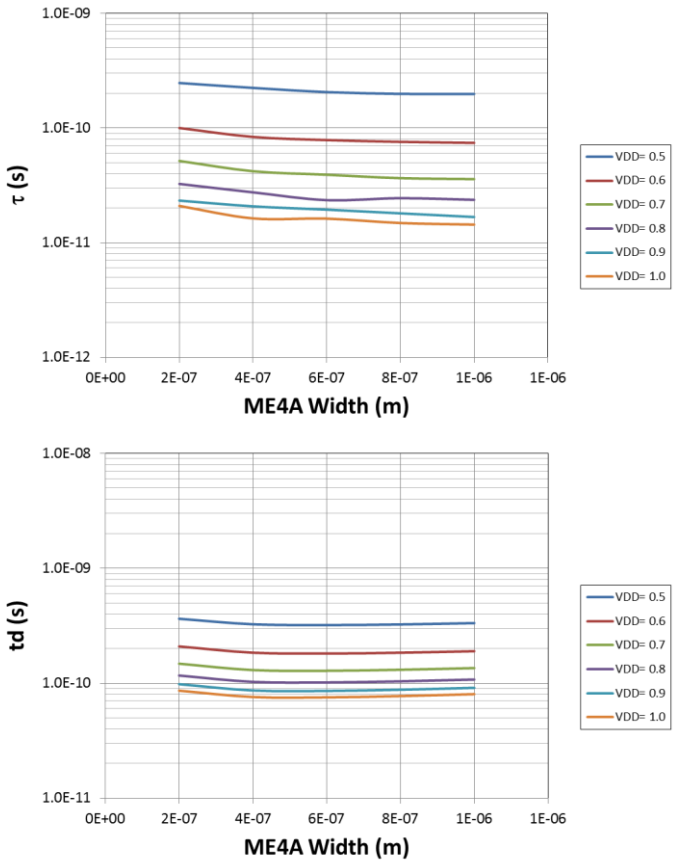


Figure D.7 Impact of Latch size on τ and t_d (ME4A)

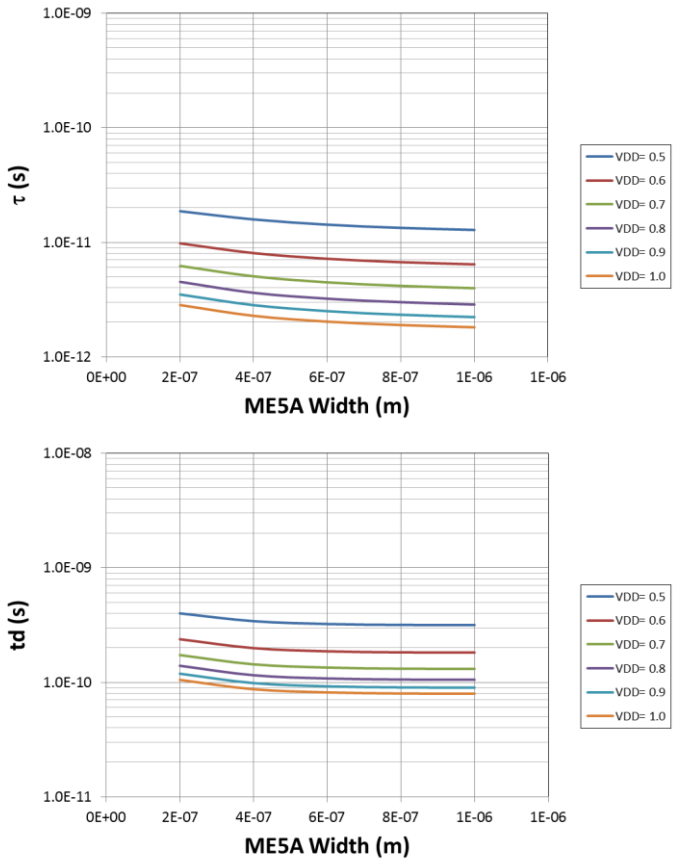


Figure D.8 Impact of Latch size on τ and t_d (ME5A)

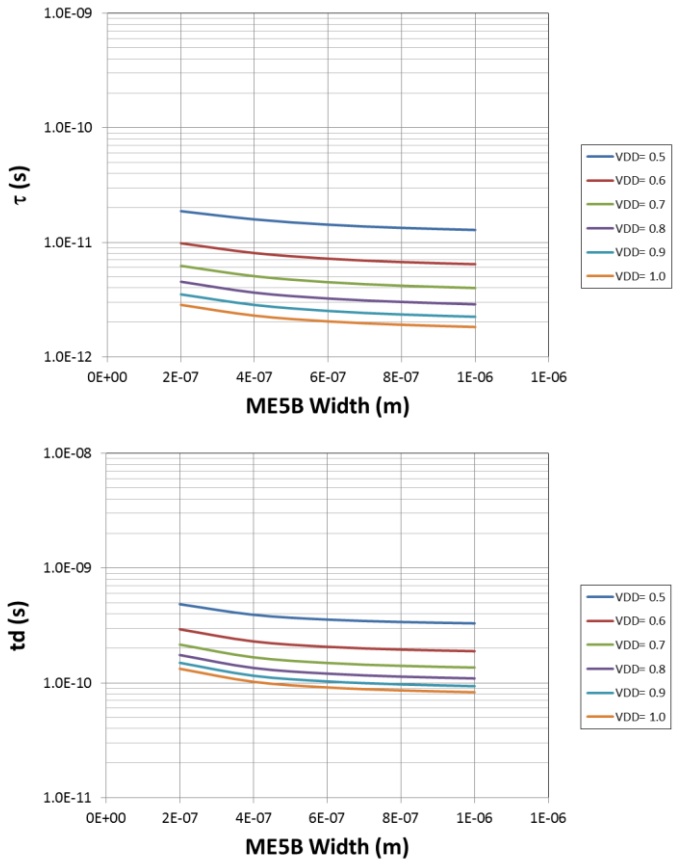


Figure D.9 Impact of Latch size on τ and t_d (ME5A)

D.2 Load Size

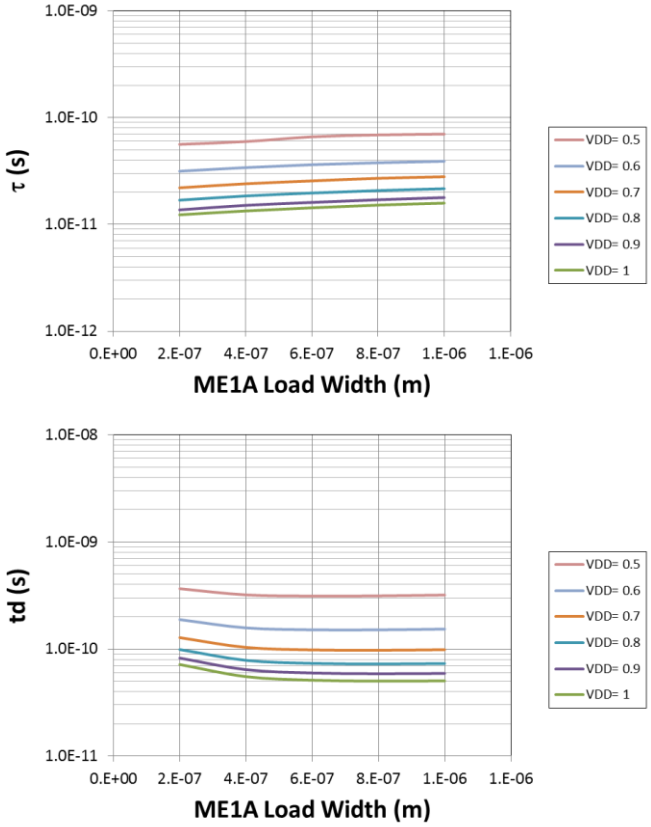


Figure D.10 Impact Load size on τ and t_d (ME1A)

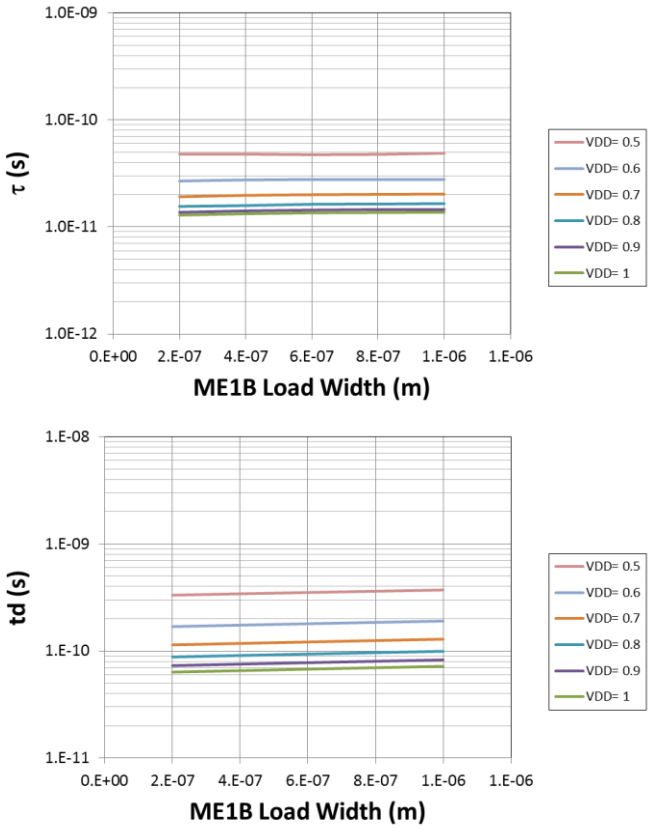
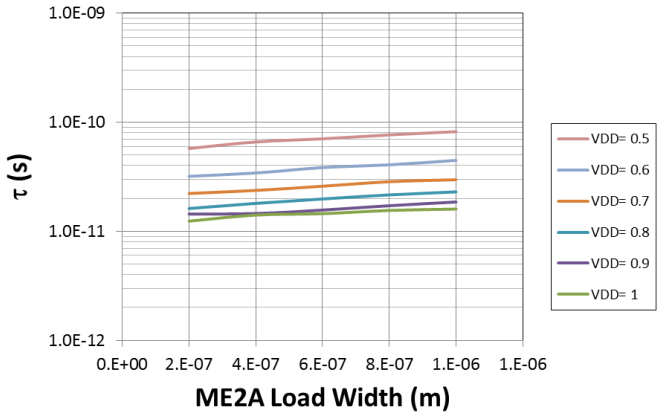


Figure D.11 Impact Load size on τ and t_d (ME1B)



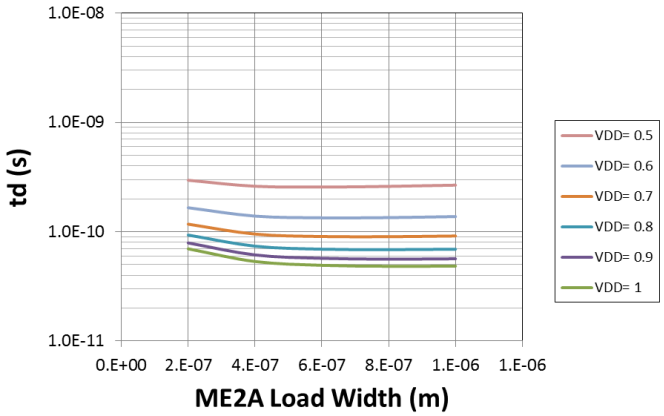
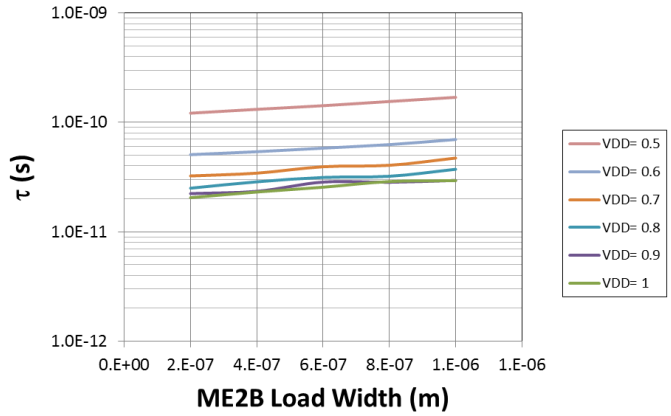


Figure D.12 Impact Load size on τ and t_d (ME2A)



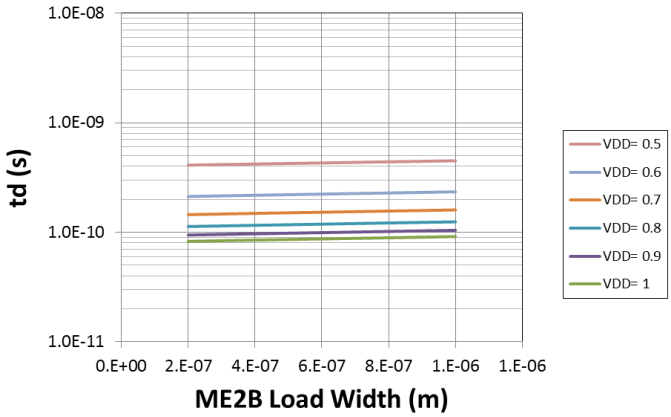
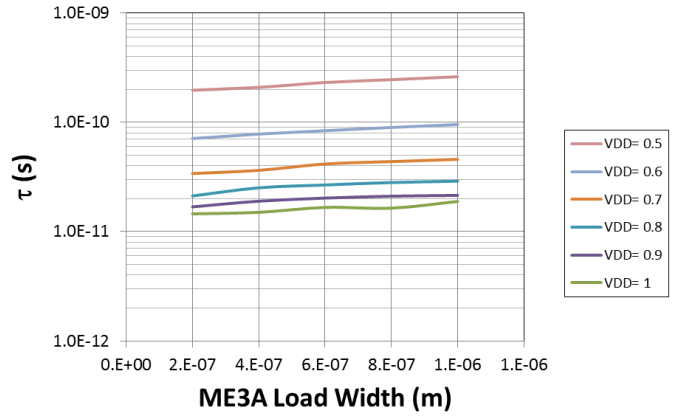


Figure D.13 Impact Load size on τ and t_d (ME2B)



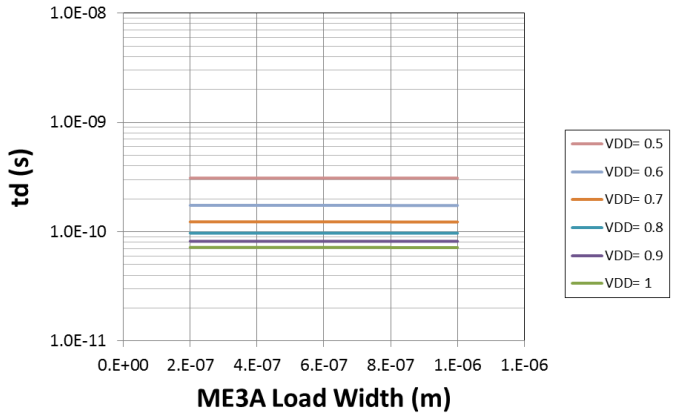
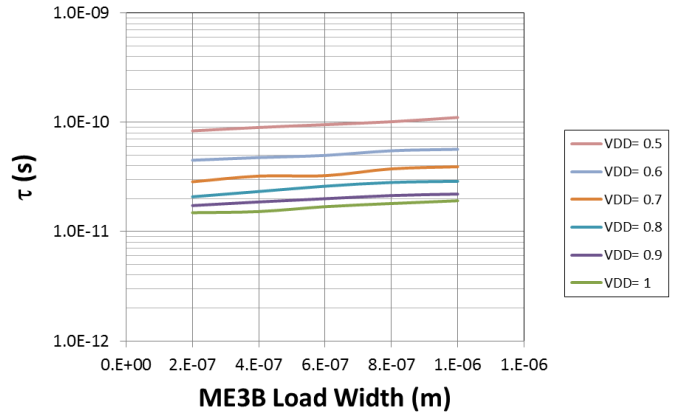


Figure D.14 Impact Load size on τ and t_d (ME3A)



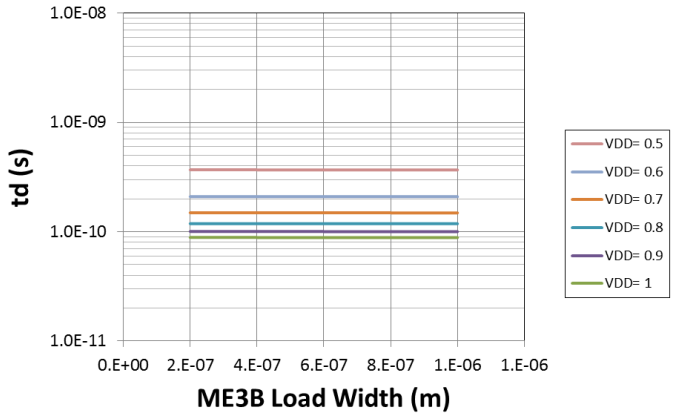
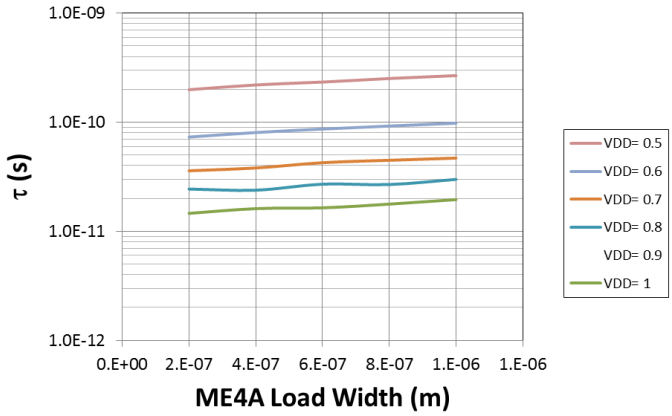


Figure D.15 Impact Load size on τ and t_d (ME3B)



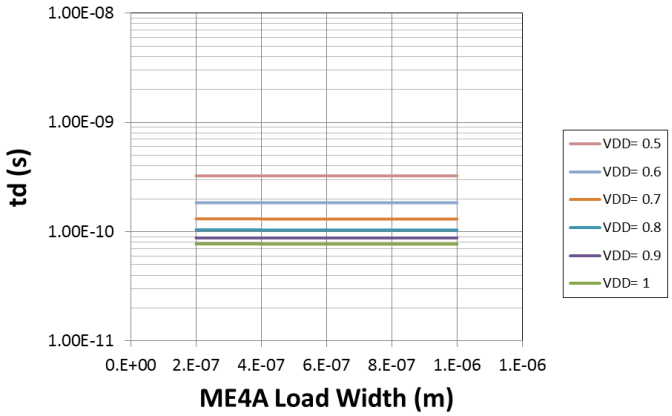
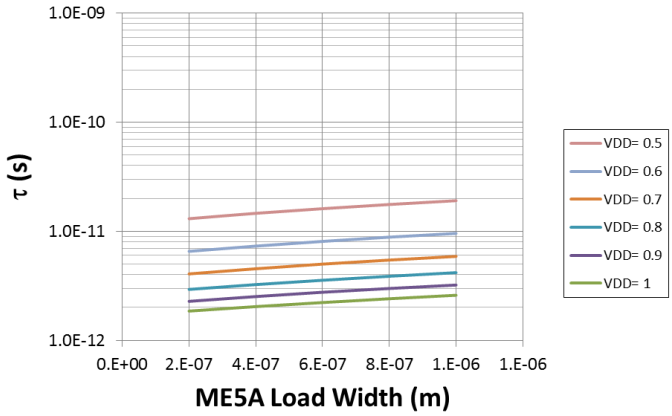


Figure D.16 Impact Load size on τ and t_d (ME4A)



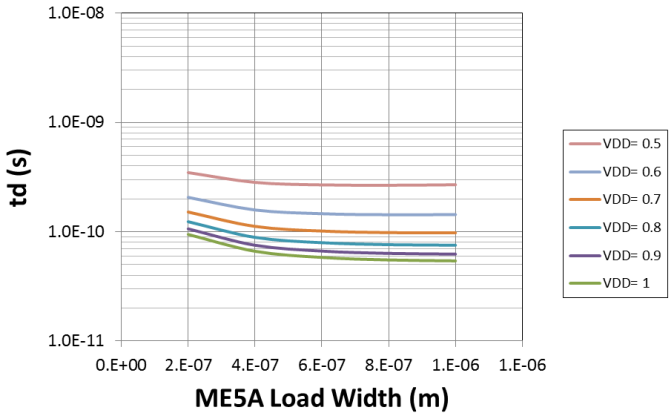


Figure D.17 Impact Load size on τ and t_d (ME5A)

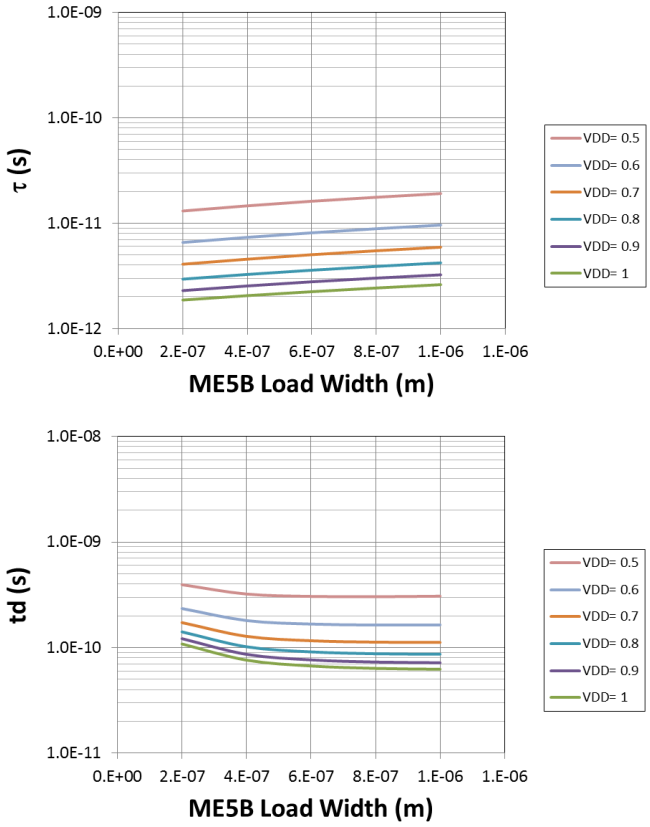


Figure D.18 Impact Load size on τ and t_d (ME5B).

D.3 Temperature Effect

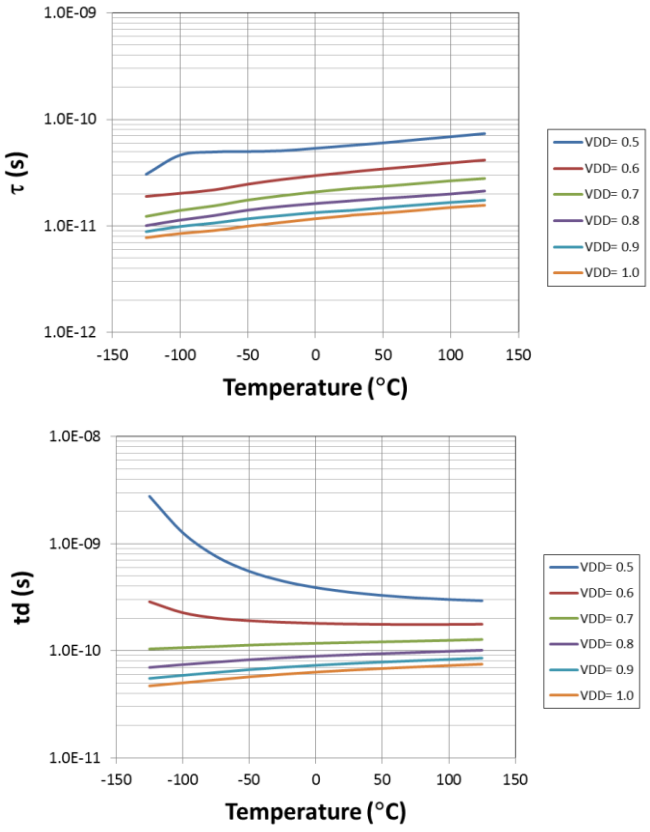


Figure D.19 Impact Voltage and Temperature on τ and t_d (ME1A)

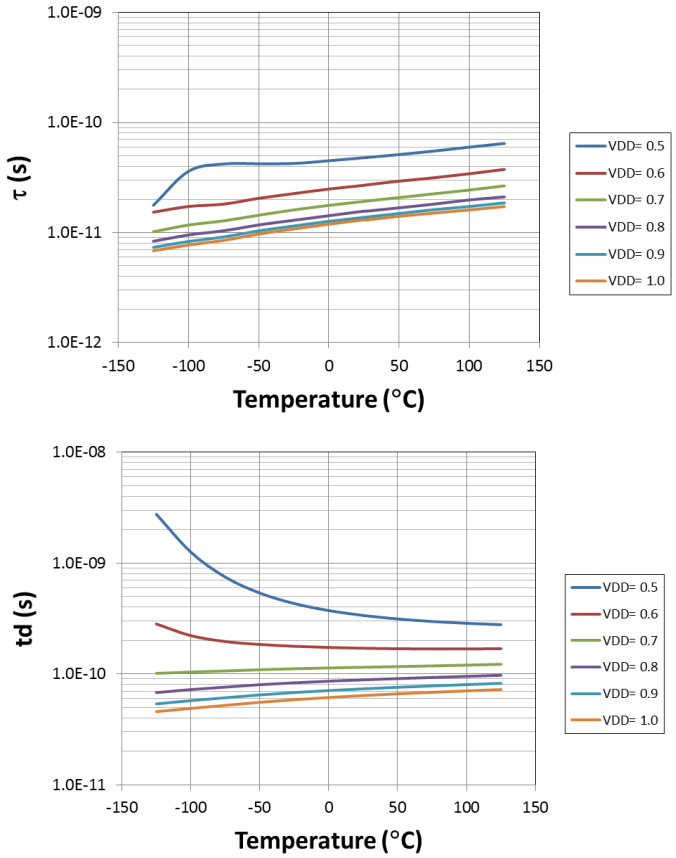


Figure D.20 Impact Voltage and Temperature on τ and t_d (ME1B)

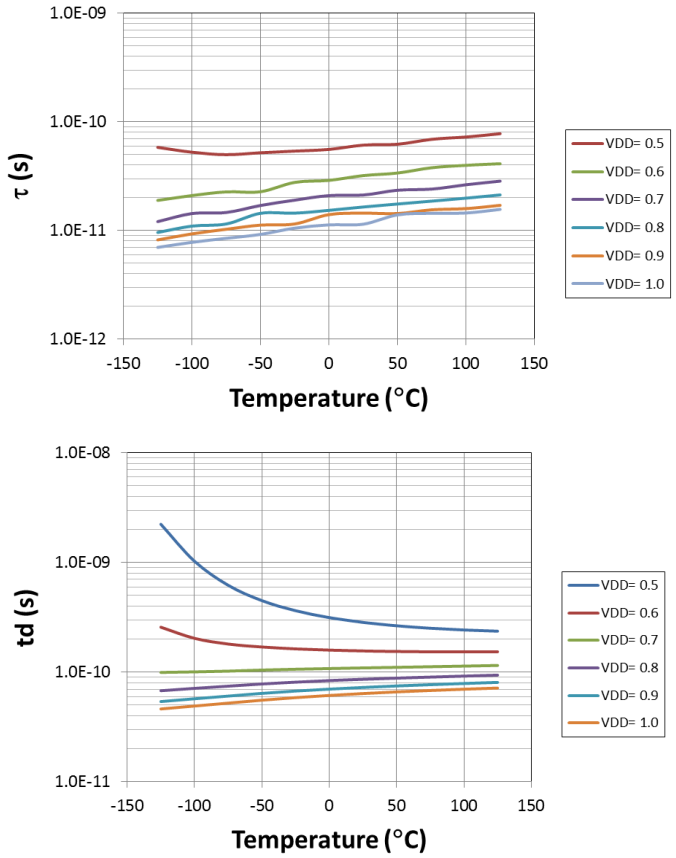


Figure D.21 Impact Voltage and Temperature on τ and t_d (ME2A)

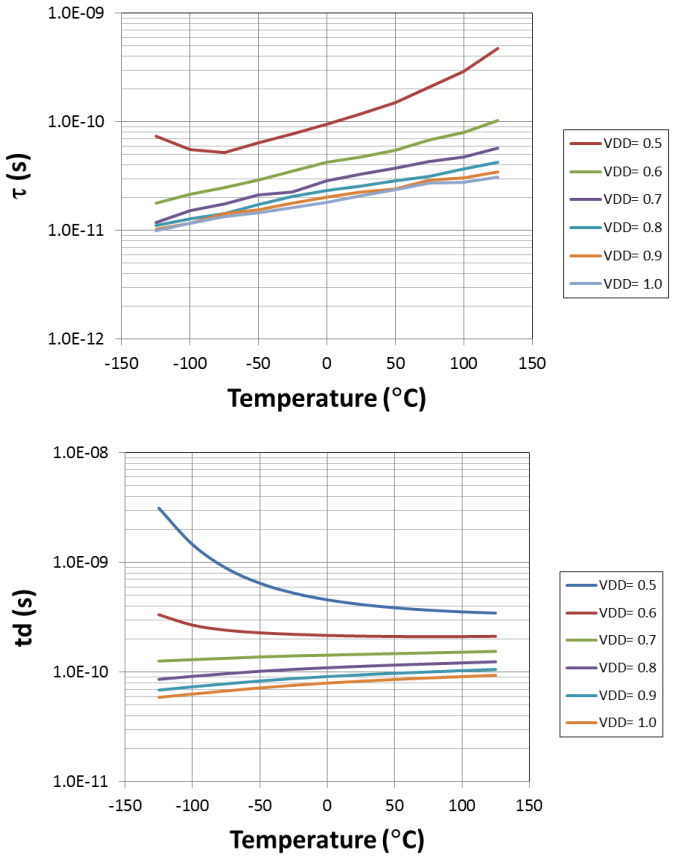


Figure D.22 Impact Voltage and Temperature on τ and t_d (ME2B)

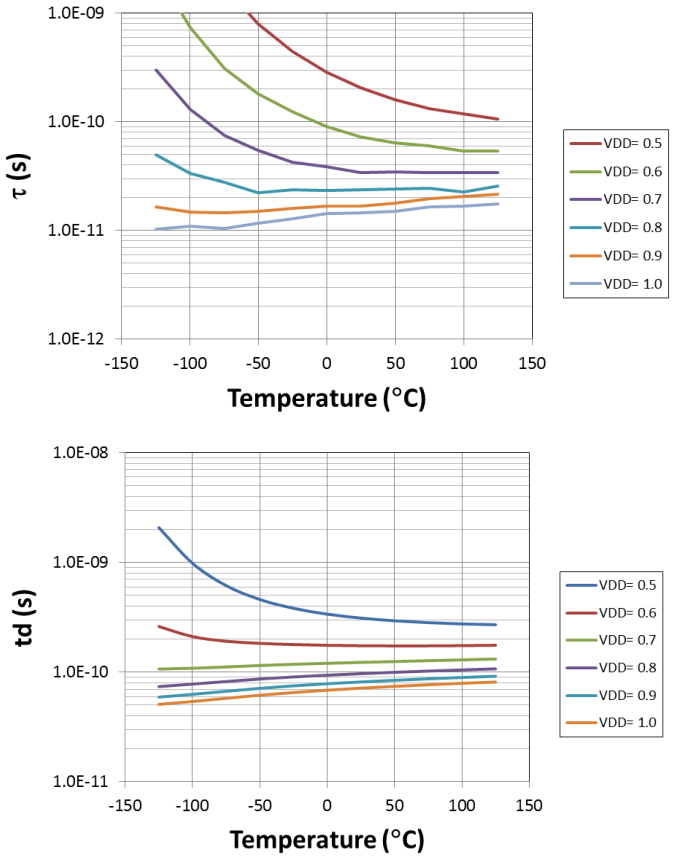


Figure D.23 Impact Voltage and Temperature on τ and t_d (ME3A)

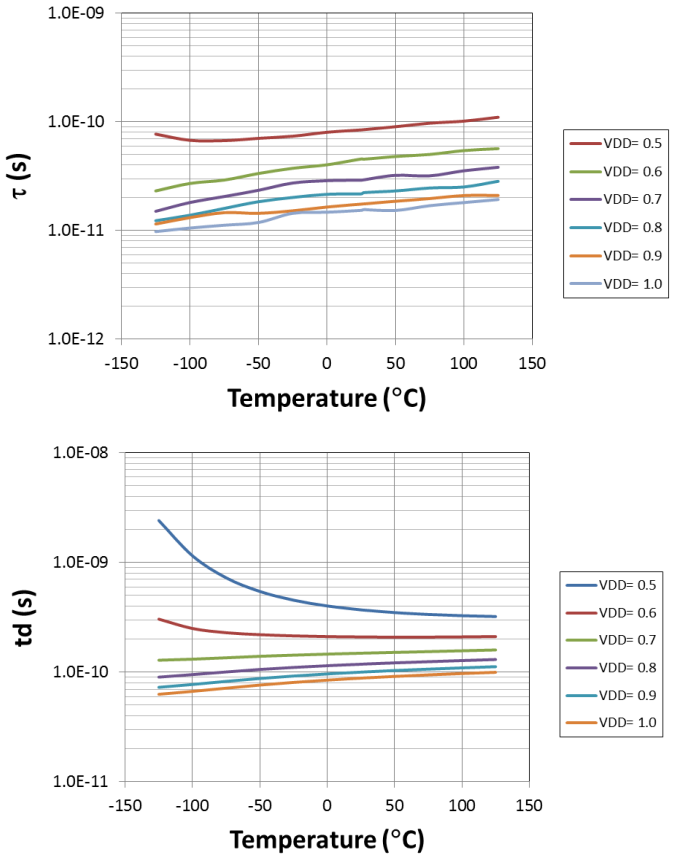


Figure D.24 Impact Voltage and Temperature on τ and t_d (ME3B)

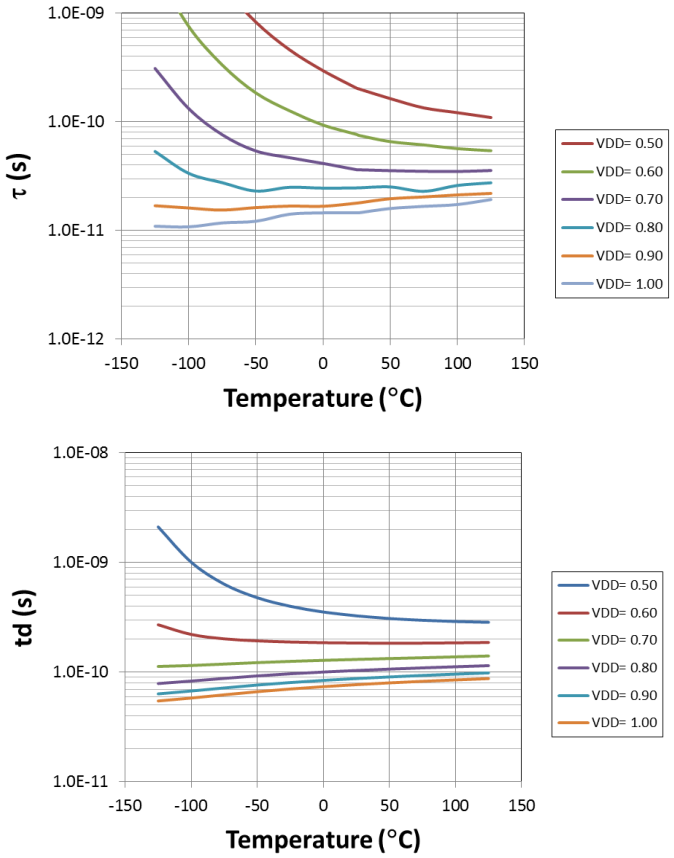


Figure D.25 Impact Voltage and Temperature on τ and t_d (ME4A)

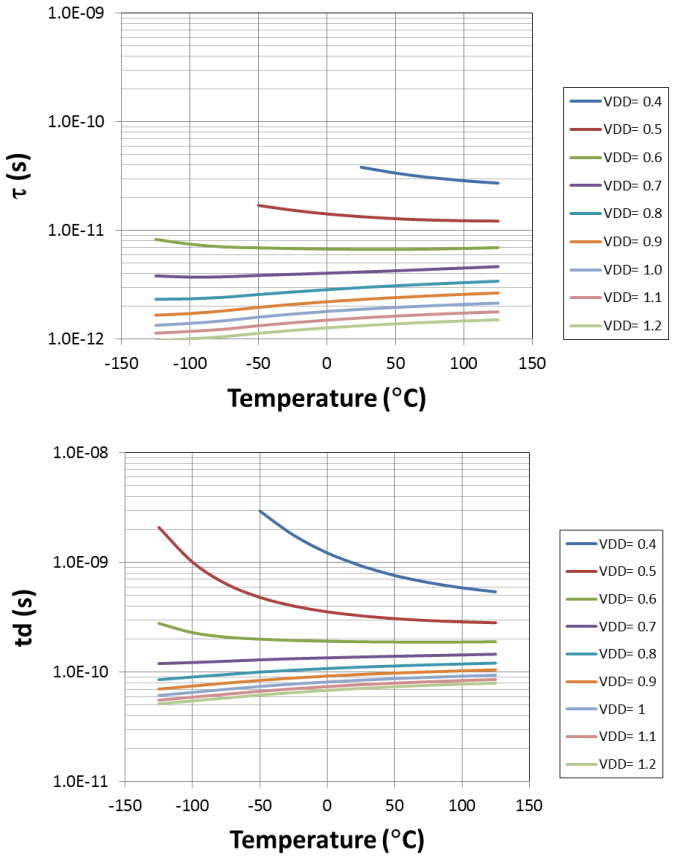


Figure D.26 Impact Voltage and Temperature on τ and t_d (ME5A)

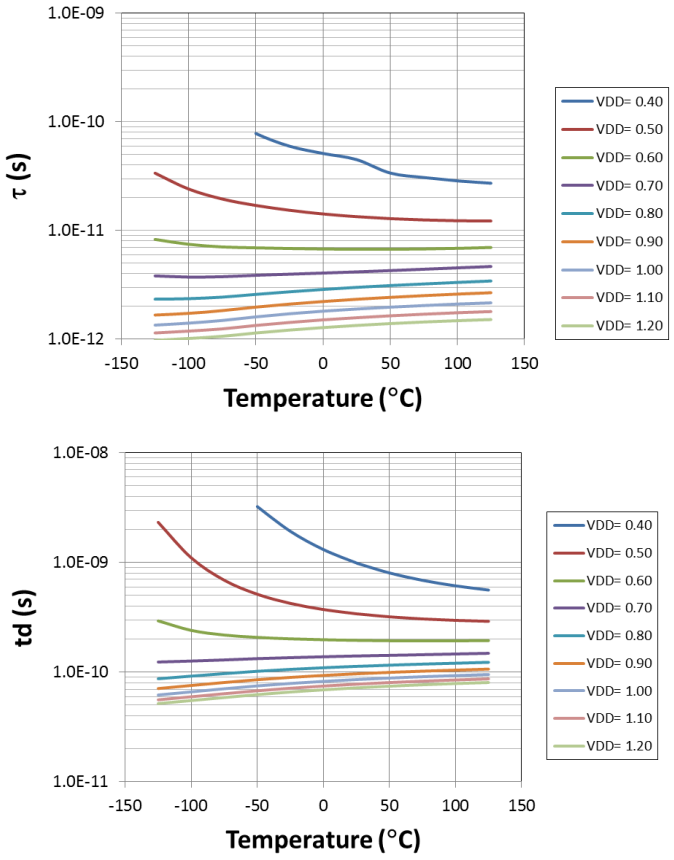
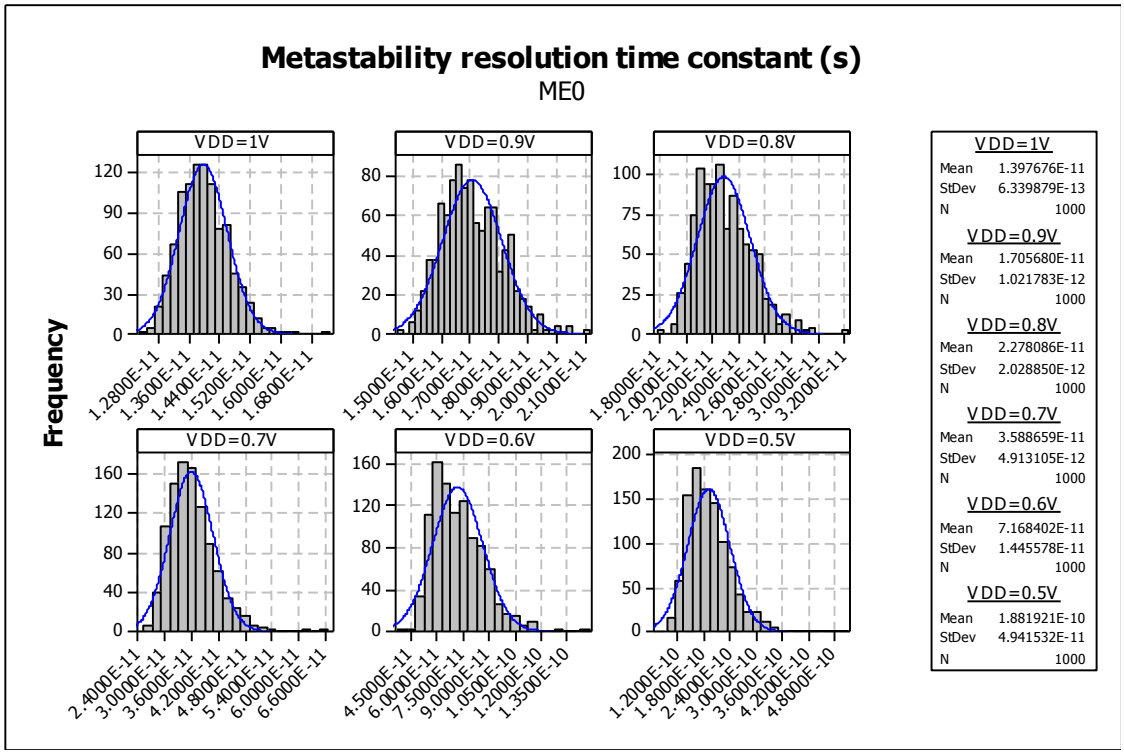


Figure D.27 Impact Voltage and Temperature on τ and t_d (ME5B)

D.4 Process Variations Effect



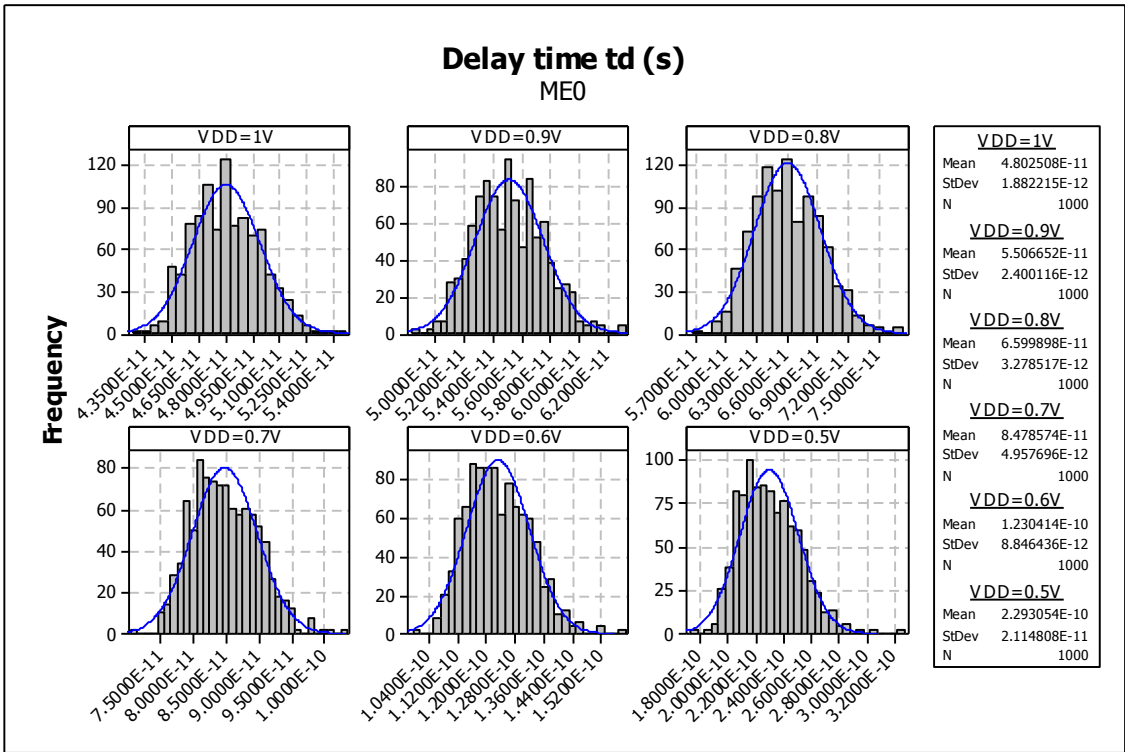


Figure D.28 Histograms of the impact Process Variation and Voltage on τ and t_d (ME0)

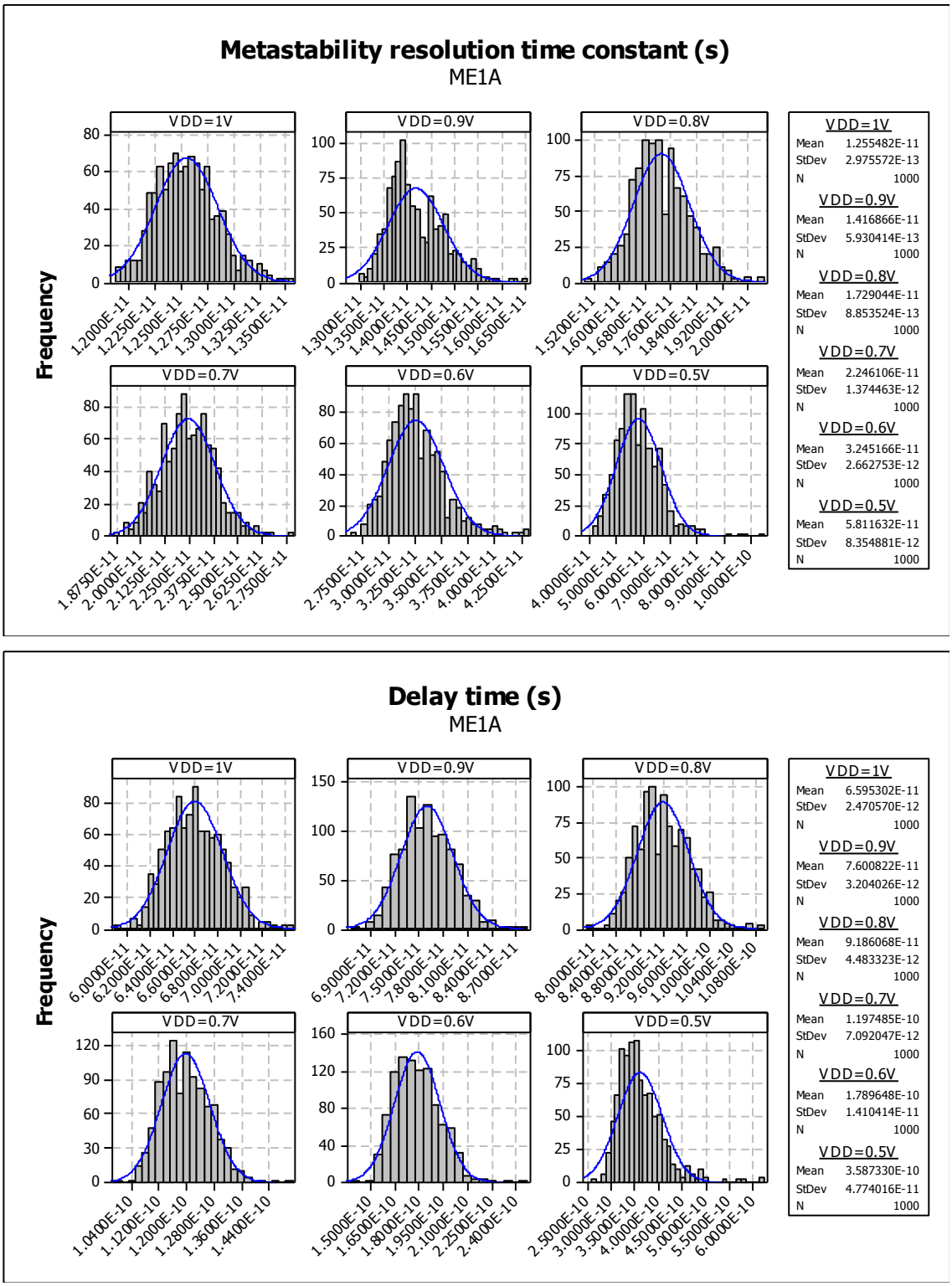


Figure D.29 Histograms of the impact Process Variation and Voltage on τ and t_d (ME1A)

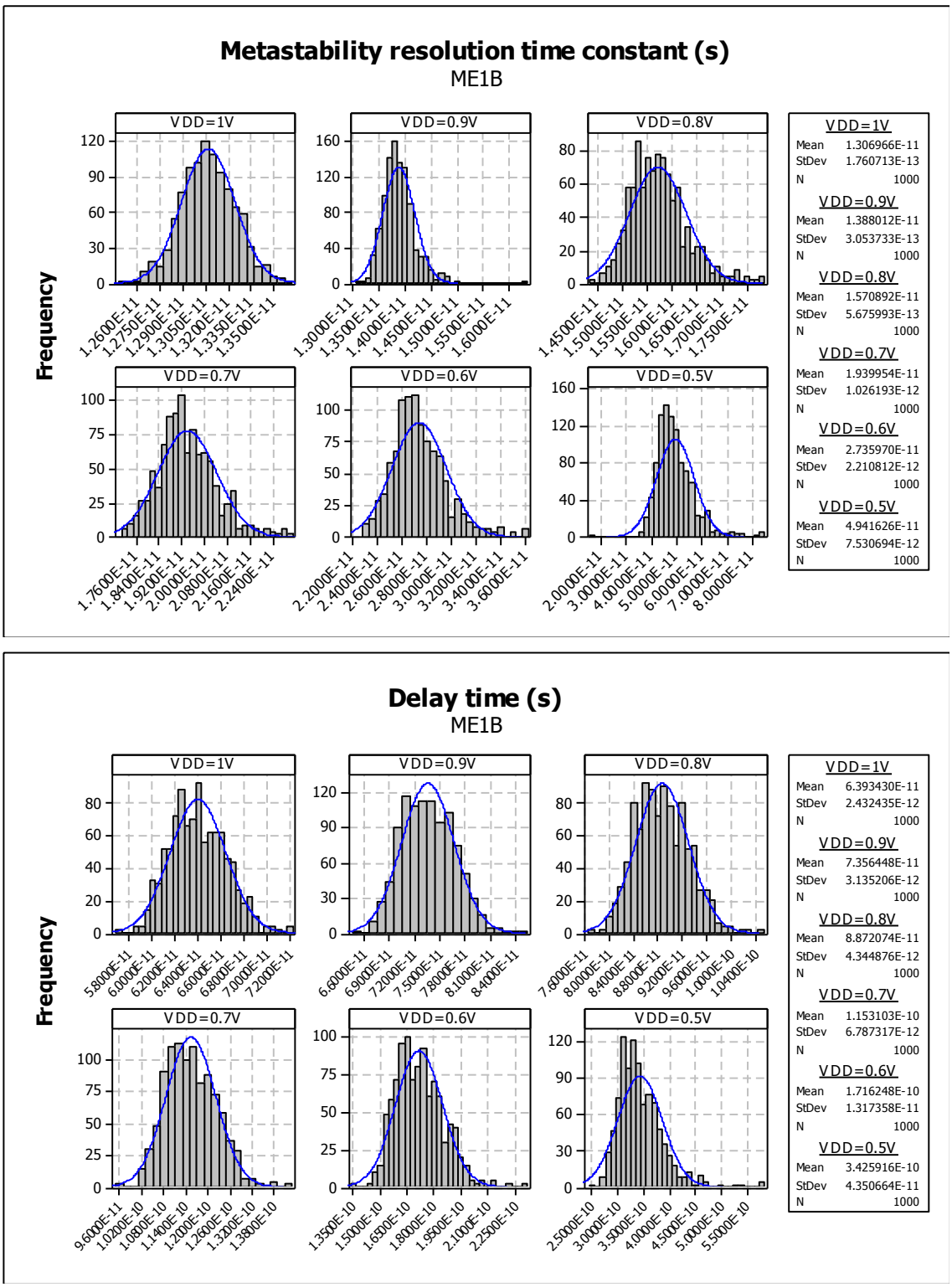


Figure D.30 Histograms of the impact Process Variation and Voltage on τ and t_d (ME1B)

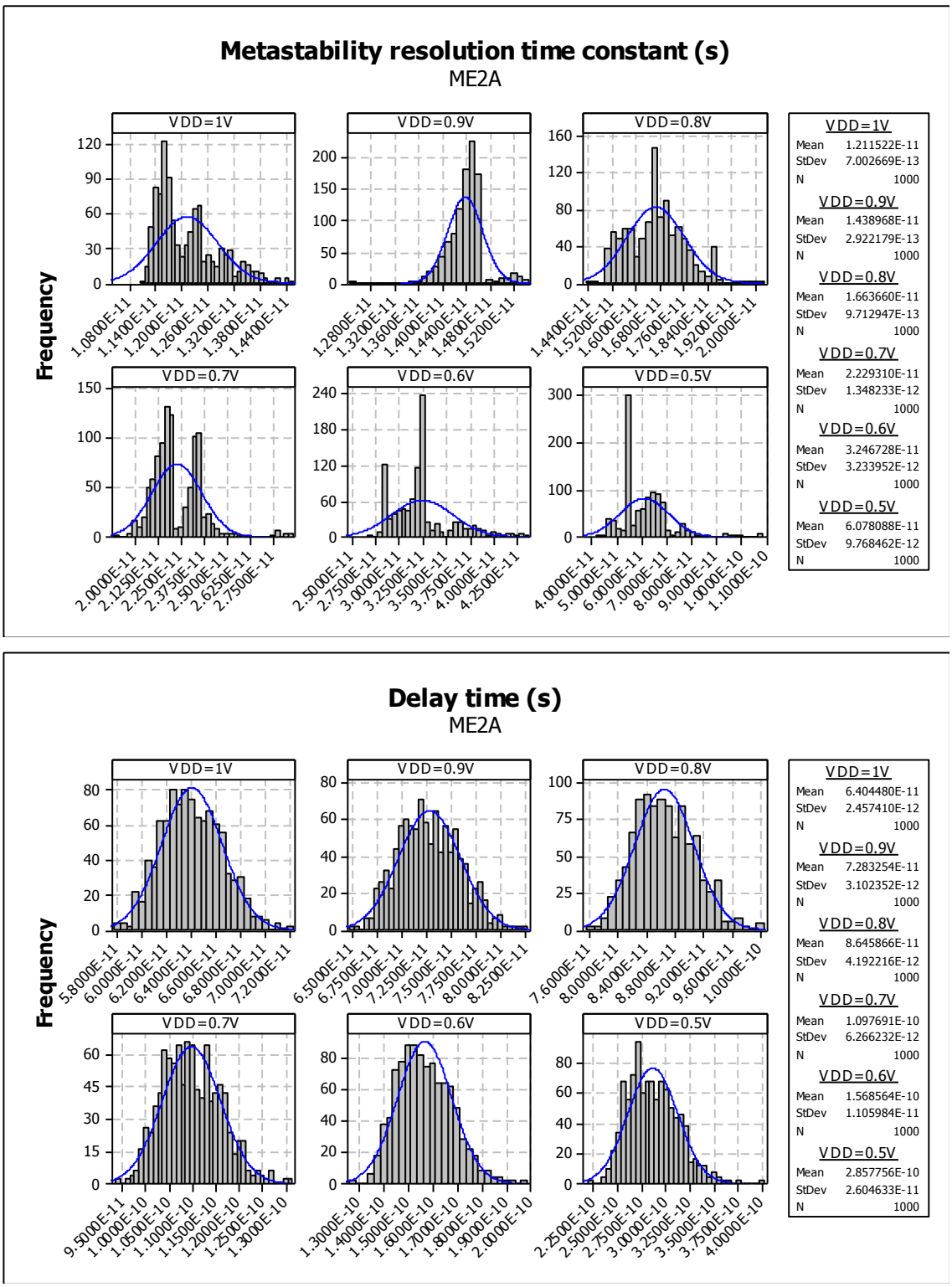


Figure D.31 Histograms of the impact Process Variation and Voltage on τ and t_d (ME1A)

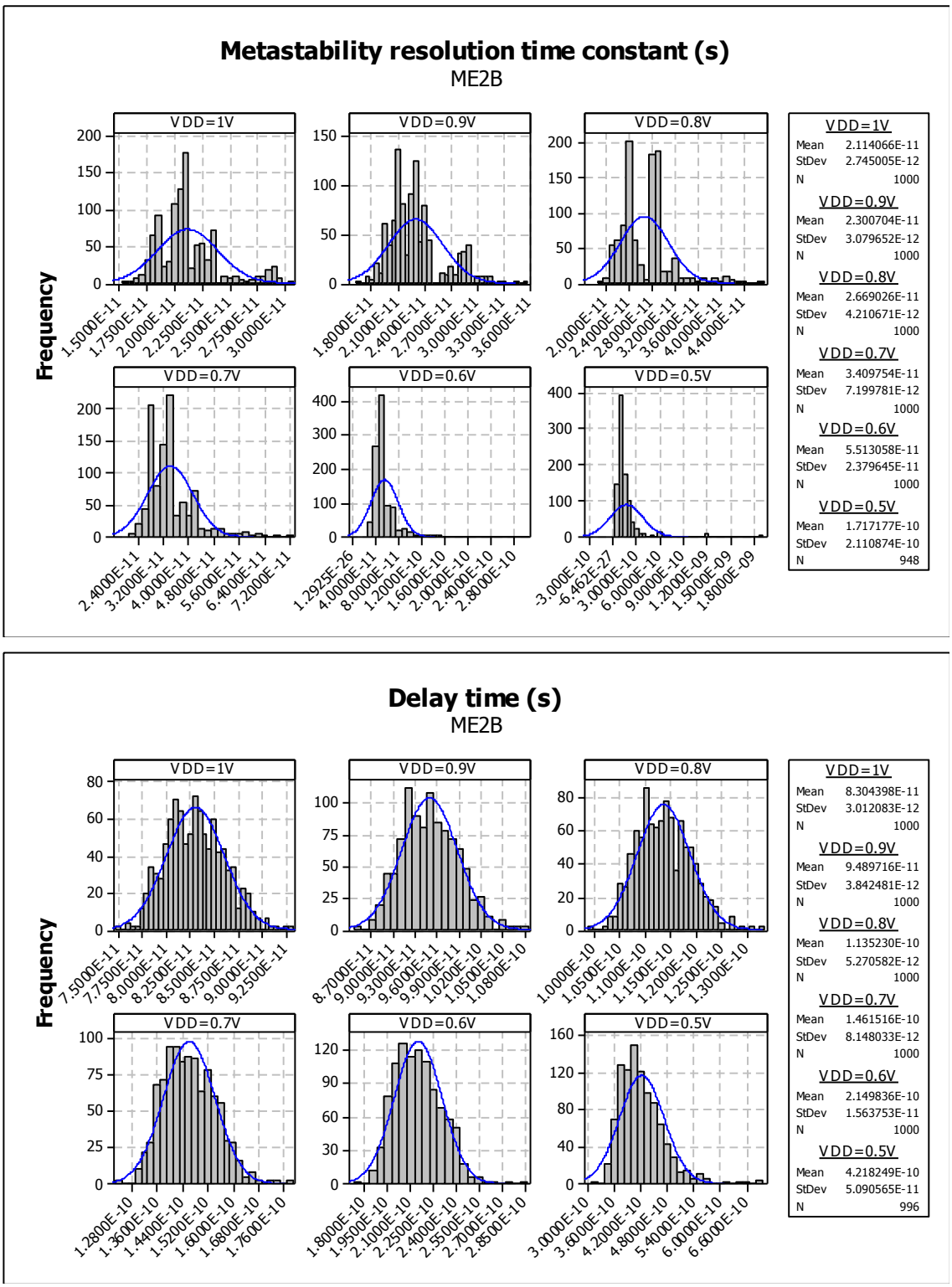
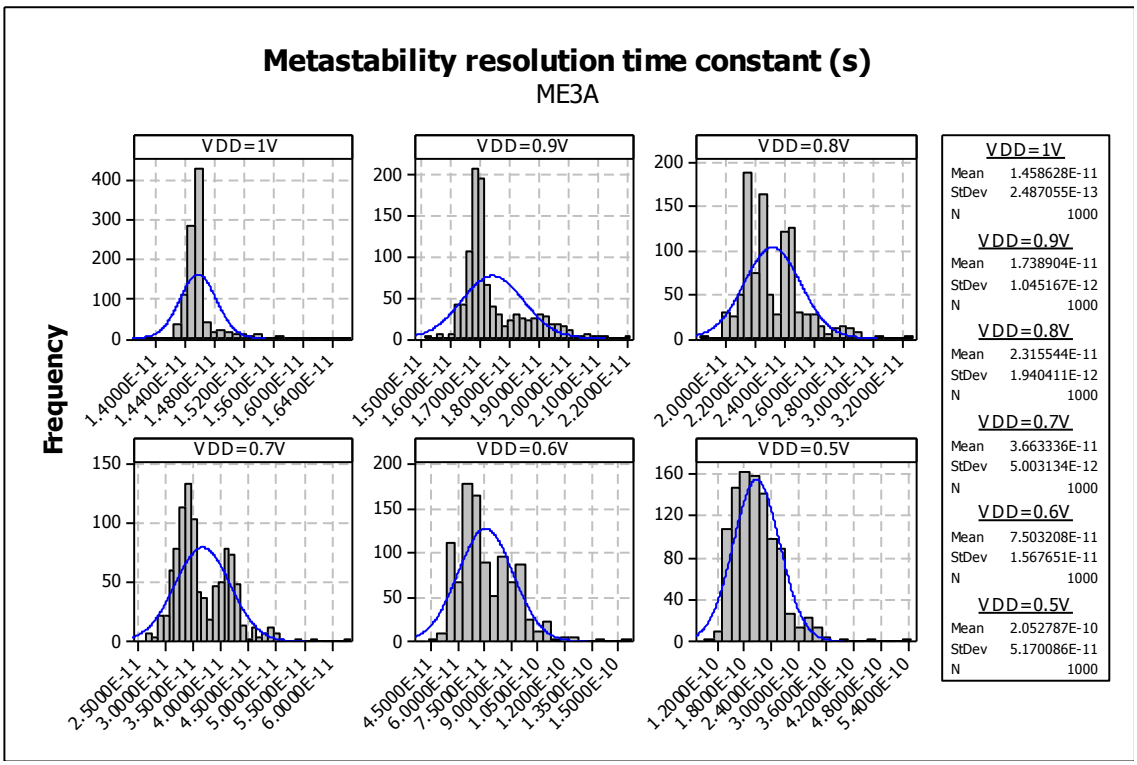


Figure D.32 Histograms of the impact Process Variation and Voltage on τ and t_d (ME2B)



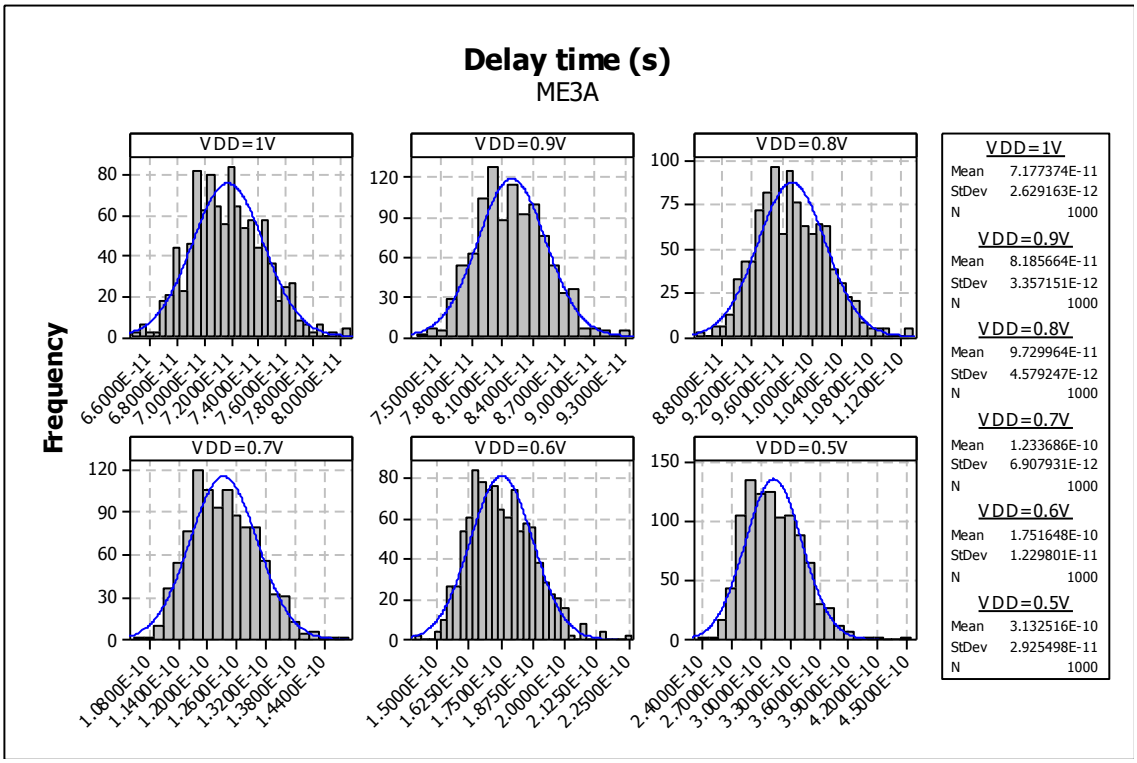


Figure D.33 Histograms of the impact Process Variation and Voltage on τ and t_d (ME3A)

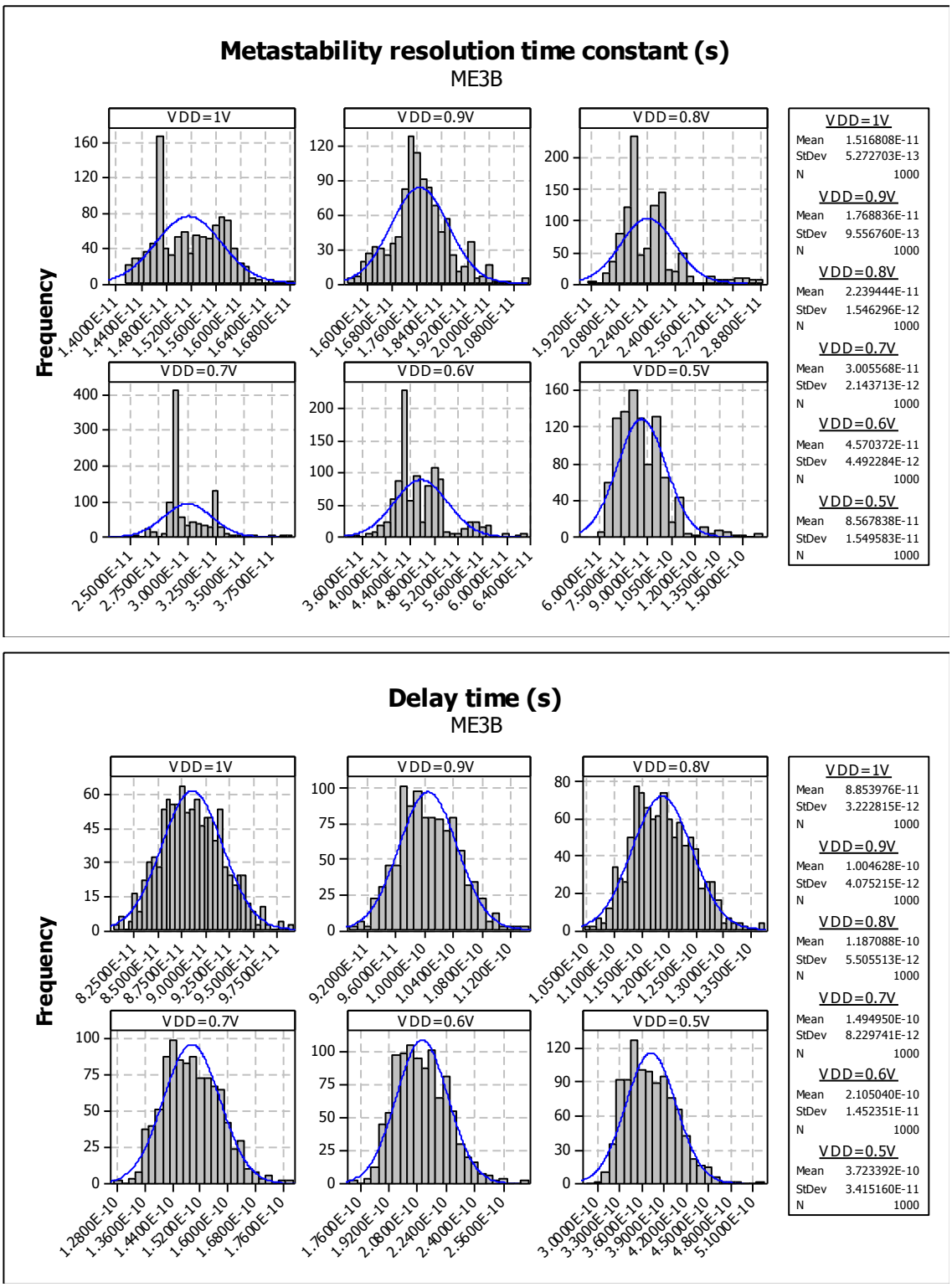


Figure D.34 Histograms of the impact Process Variation and Voltage on τ and t_d (ME3B)

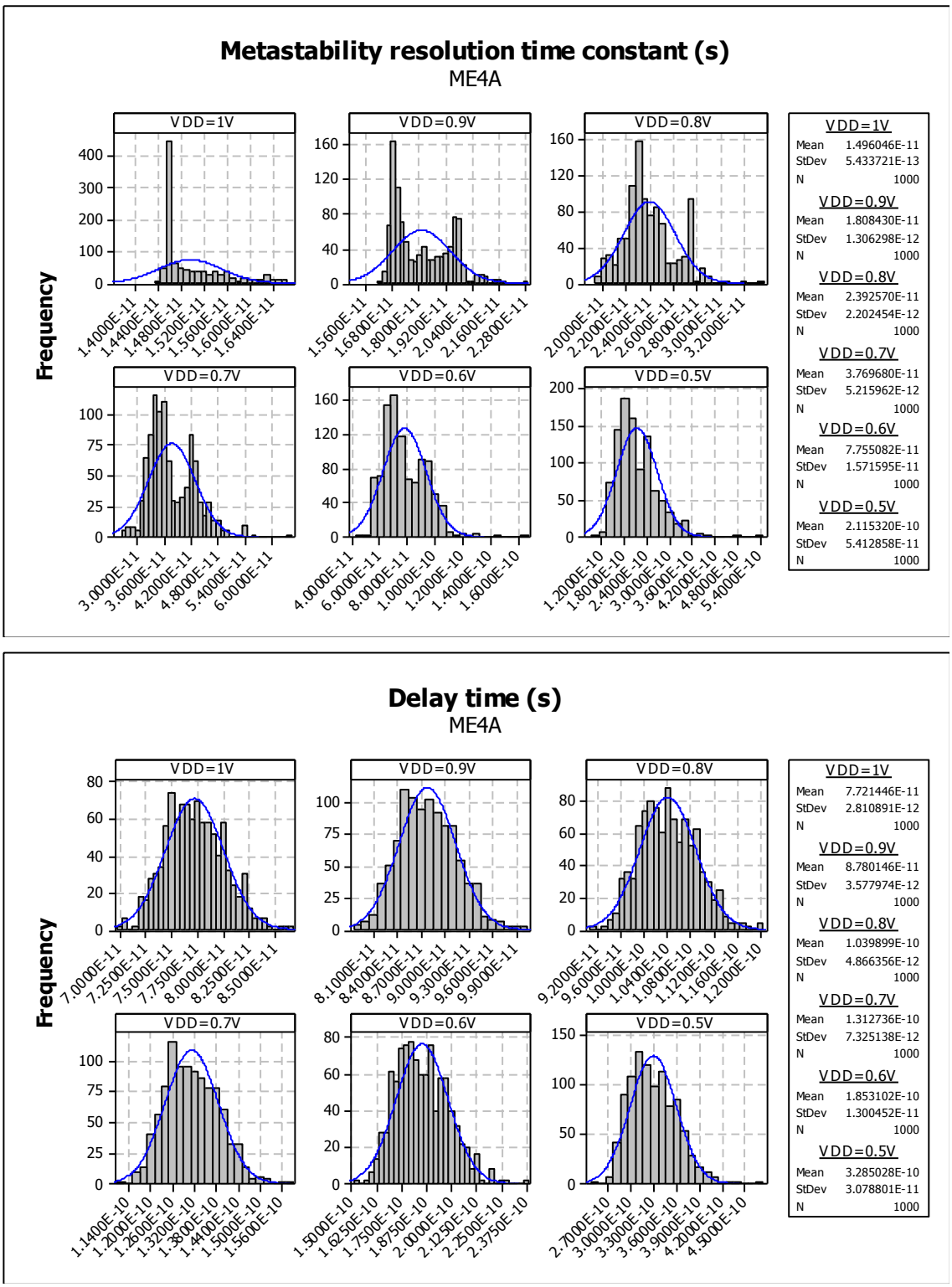


Figure D.35 Histograms of the impact Process Variation and Voltage on τ and t_d (ME4A)

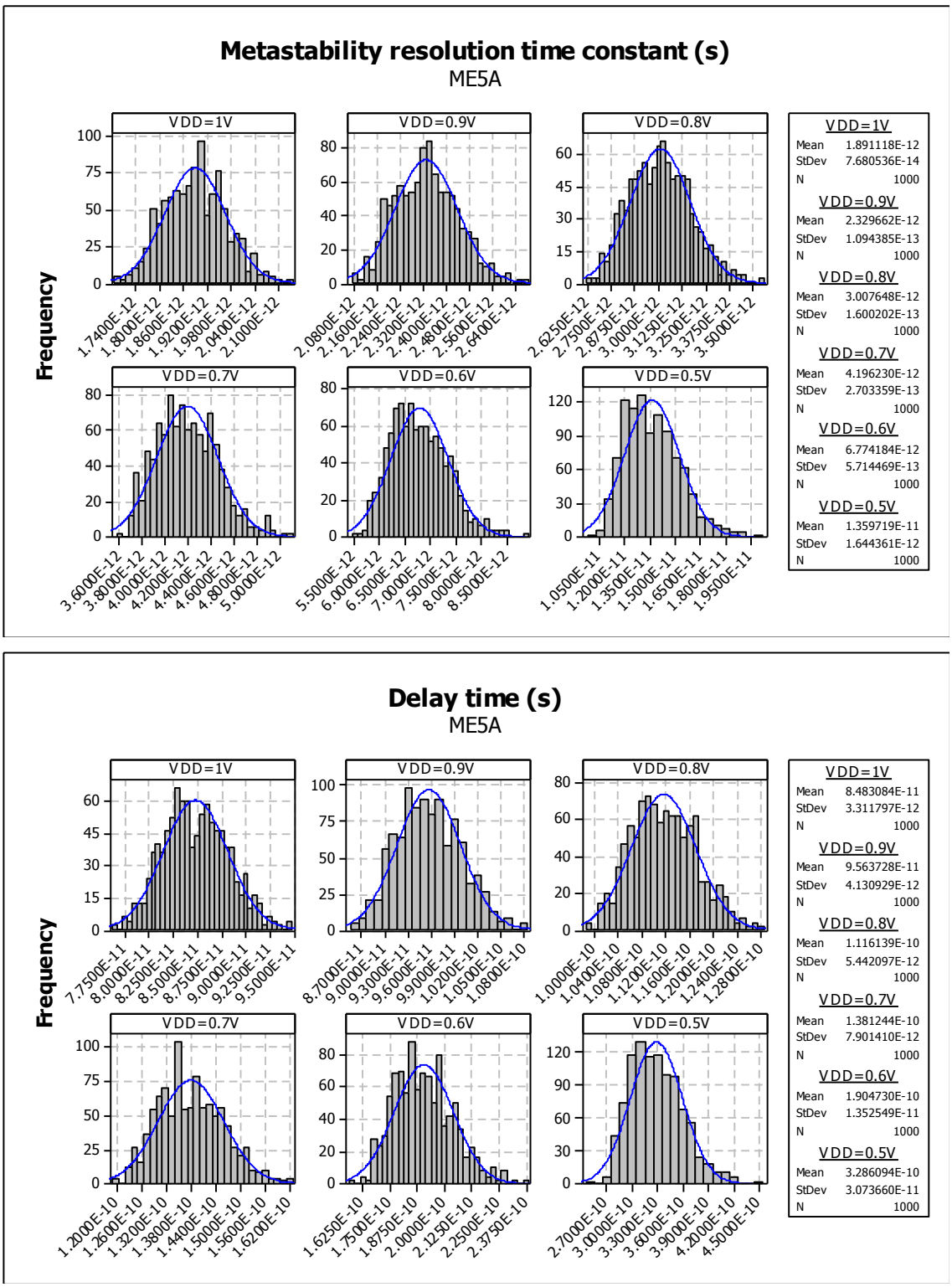


Figure D.36 Histograms of the impact Process Variation and Voltage on τ and t_d (ME5A)

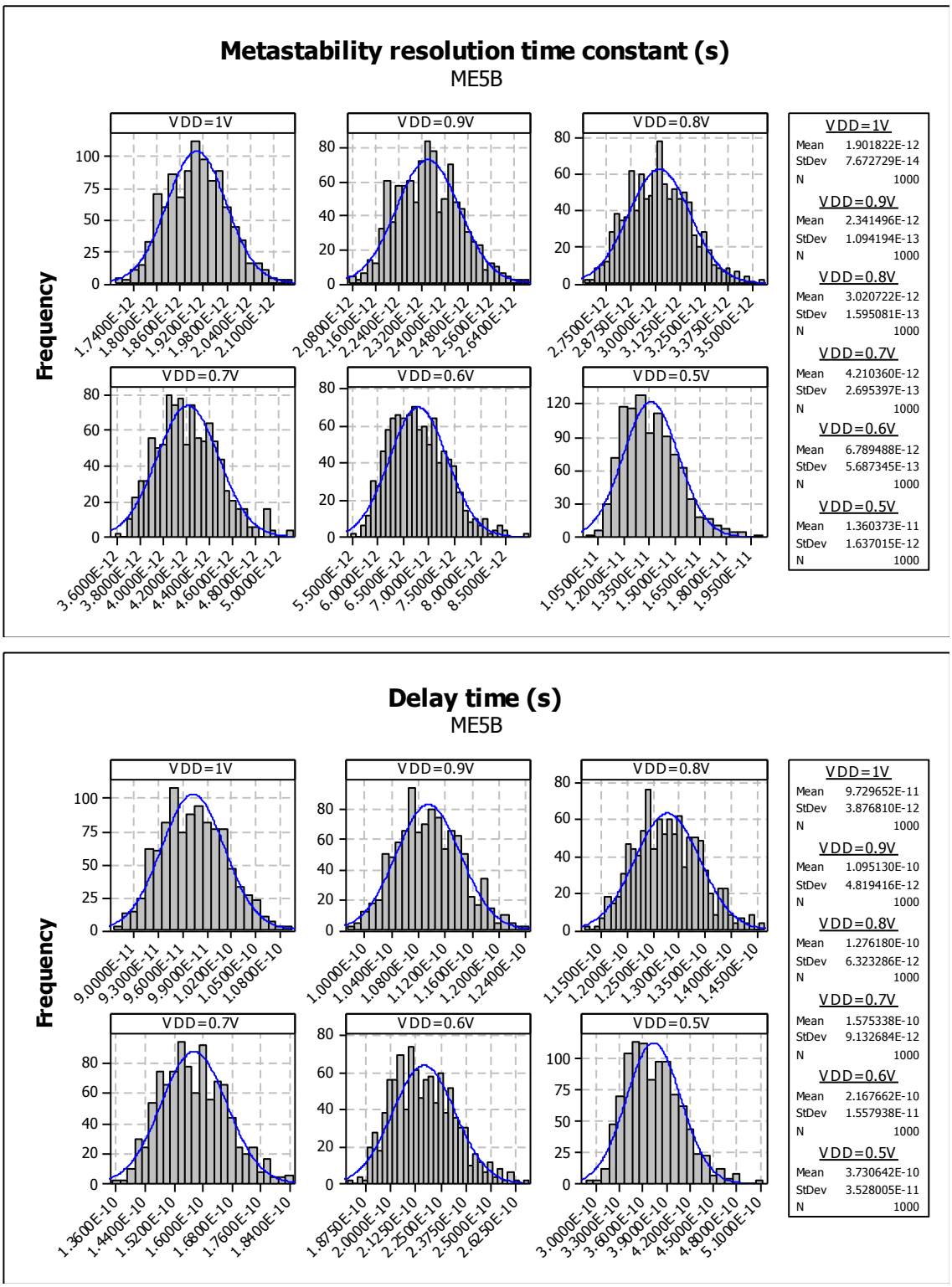


Figure D.37 Histograms of the impact Process Variation and Voltage on τ and t_d (ME5B)

D.5 Mean and Standard-Deviation

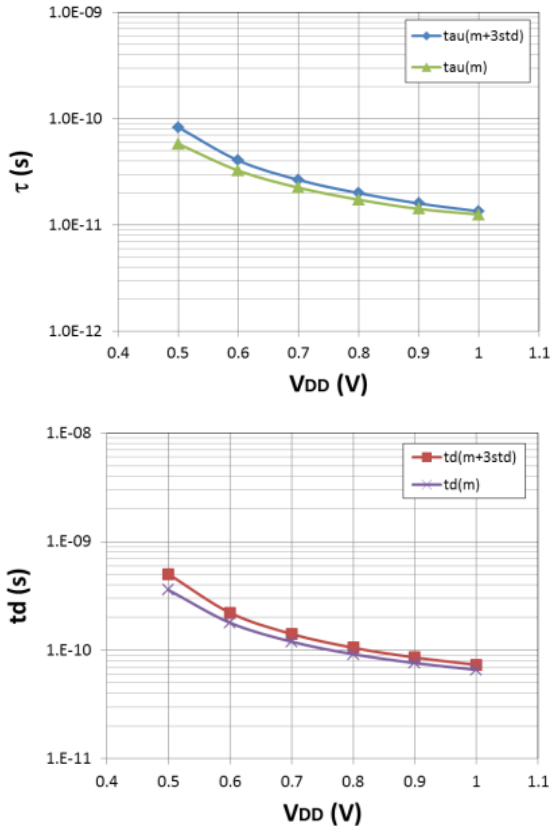


Figure D.38 Impact Process Variation and Voltage on τ and t_d (ME1A)

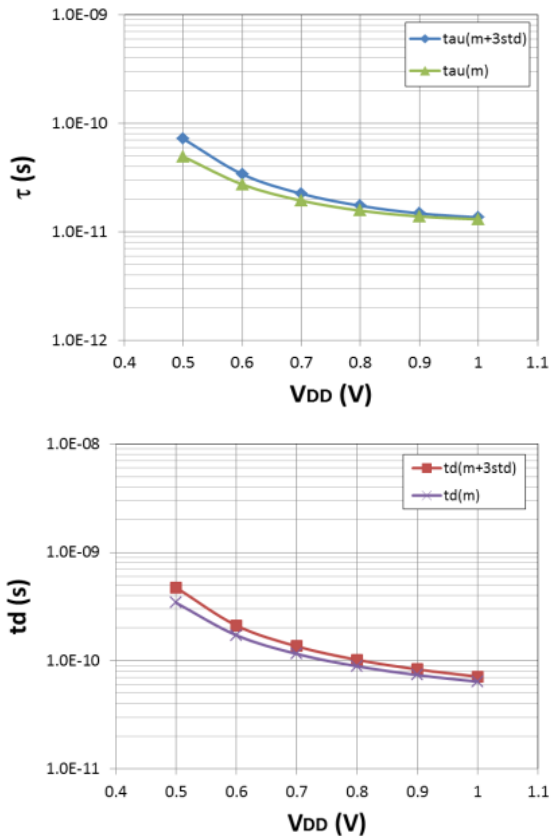


Figure D.39 Impact Process Variation and Voltage on τ and t_d (ME1B)

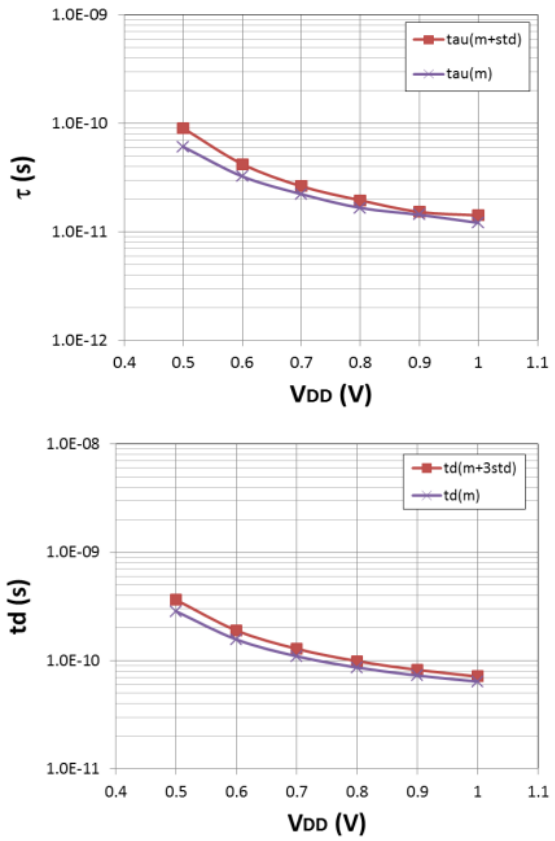


Figure D.40 Impact Process Variation and Voltage on τ and t_d (ME2A)

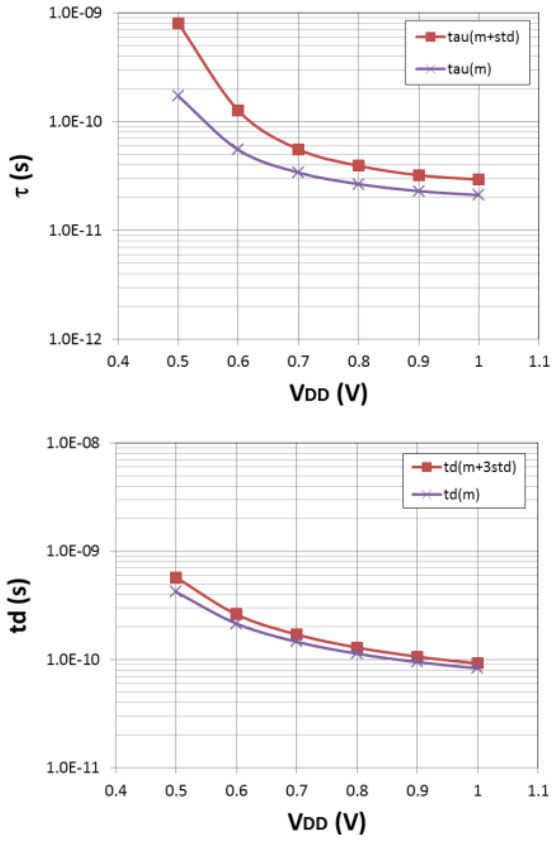


Figure D.41 Impact Process Variation and Voltage on τ and t_d (ME2B)

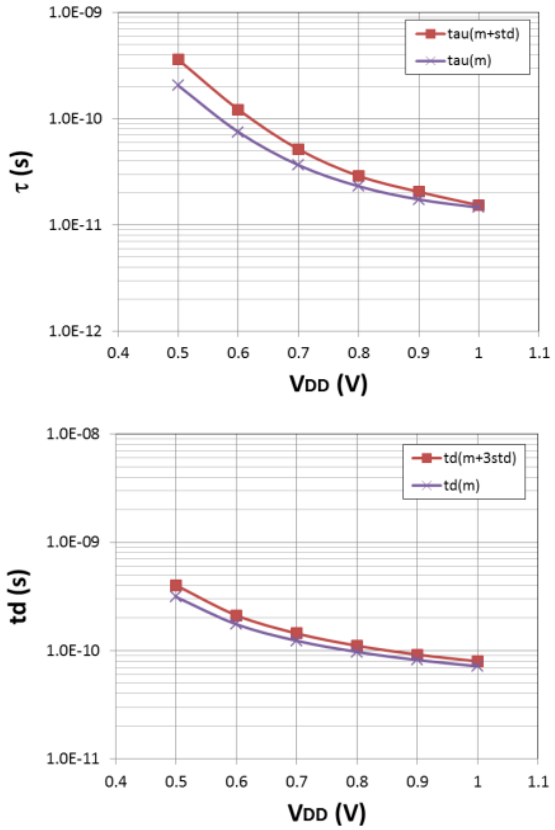


Figure D.42 Impact Process Variation and Voltage on τ and t_d (ME3A)

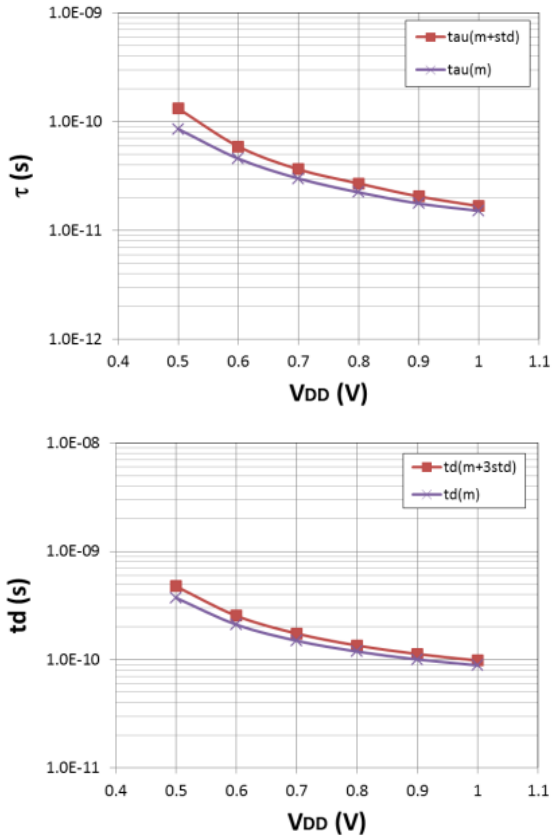


Figure D.43 Impact Process Variation and Voltage on τ and t_d (ME3B)

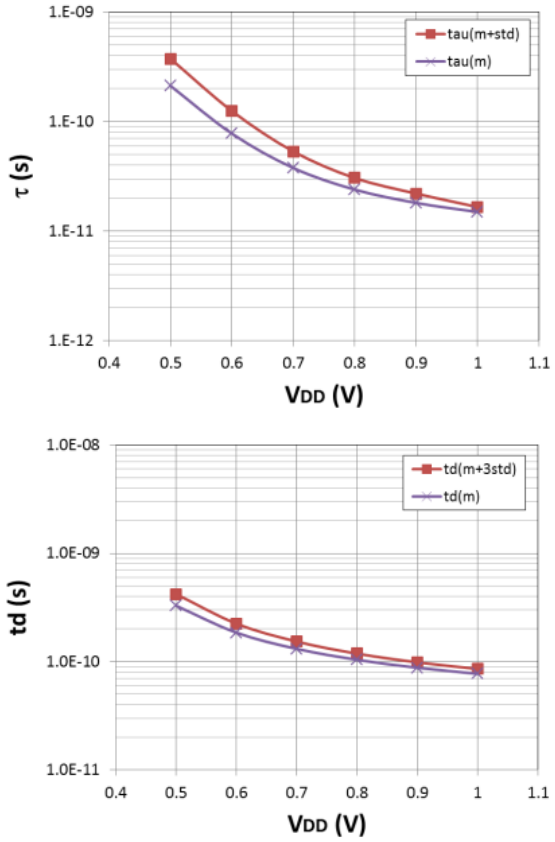


Figure D.44 Impact Process Variation and Voltage on τ and t_d (ME4A)

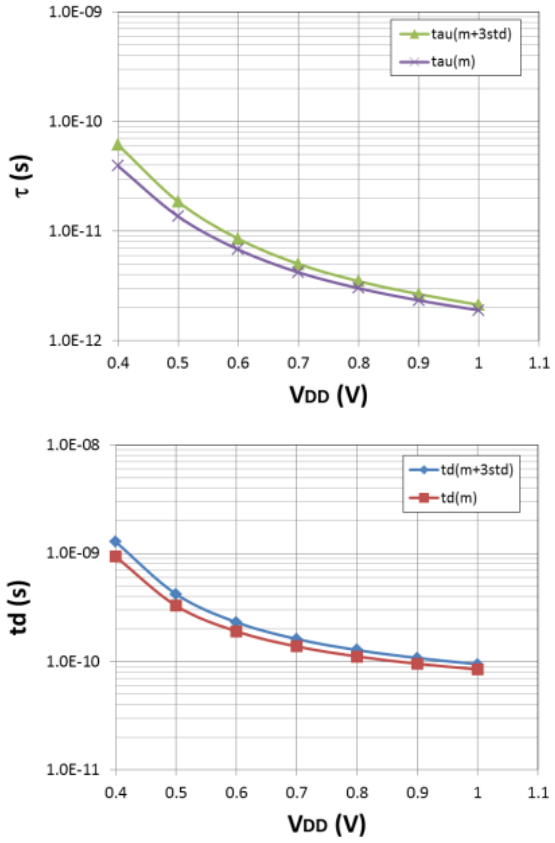
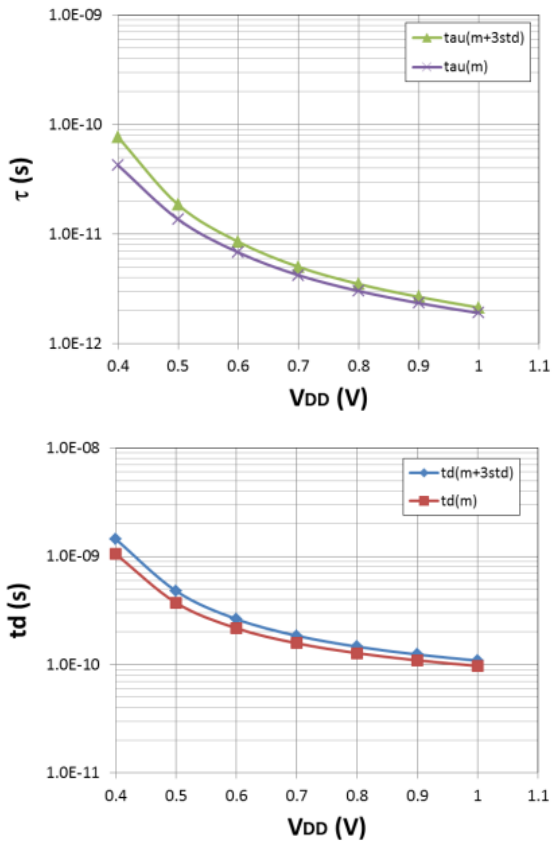
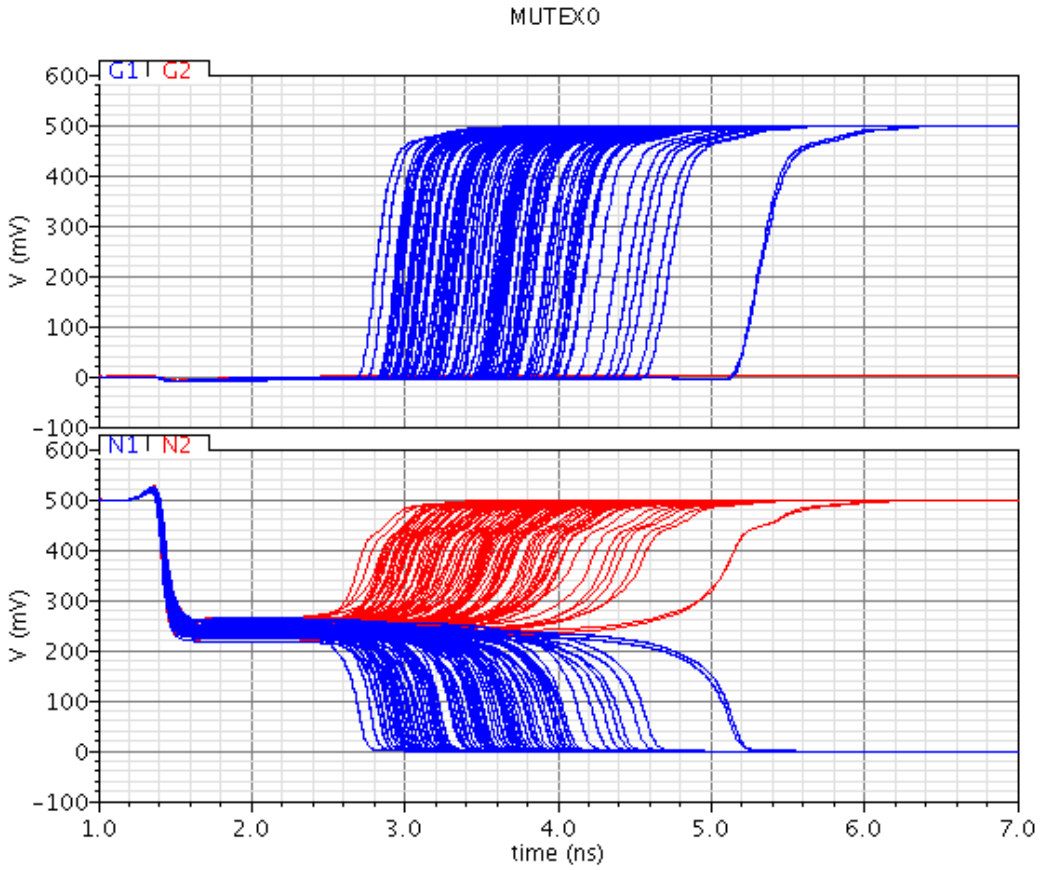


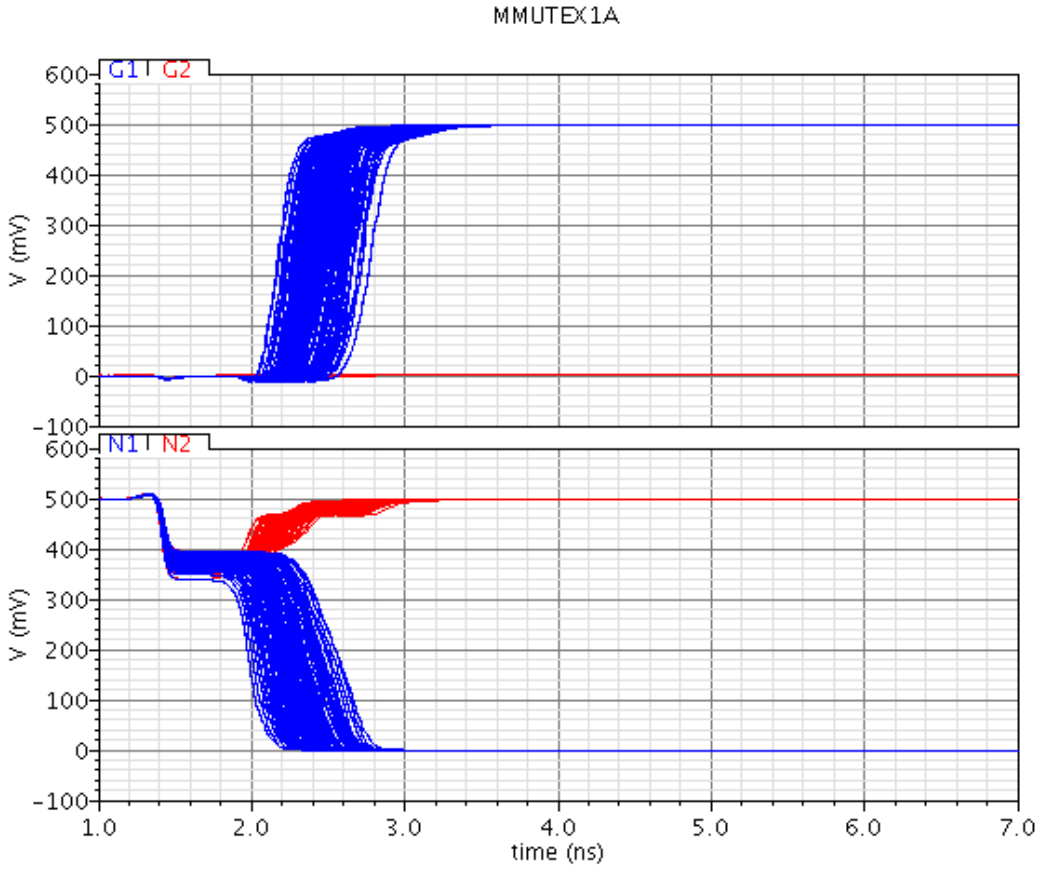
Figure D.45 Impact Process Variation and Voltage on τ and t_d (ME5A)



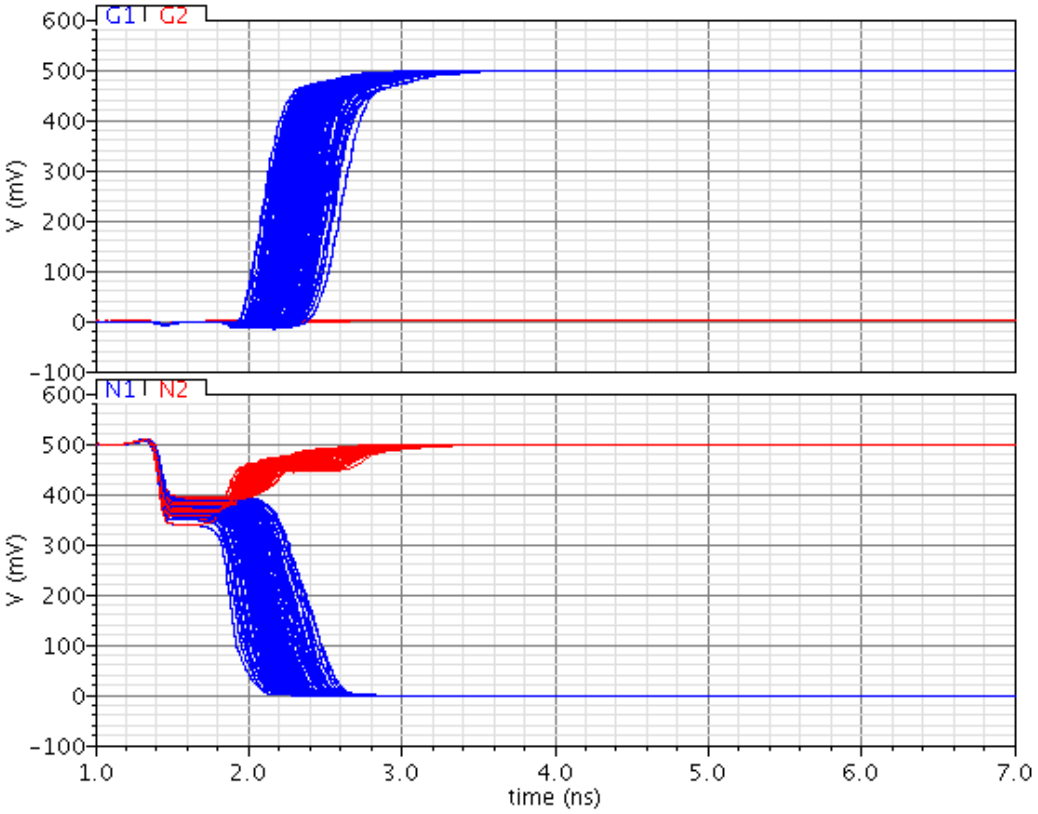
Impact Process Variation and Voltage on τ and t_d (ME5B)

D.6 Metastability Monte Carlo Simulation

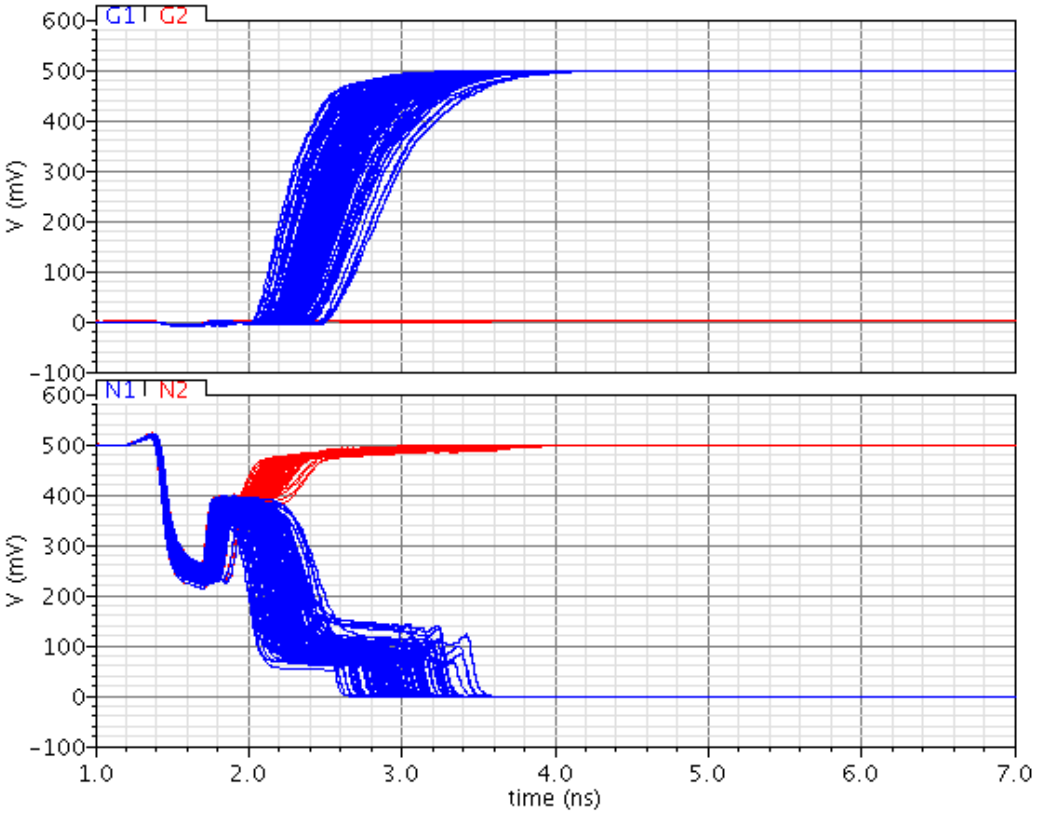




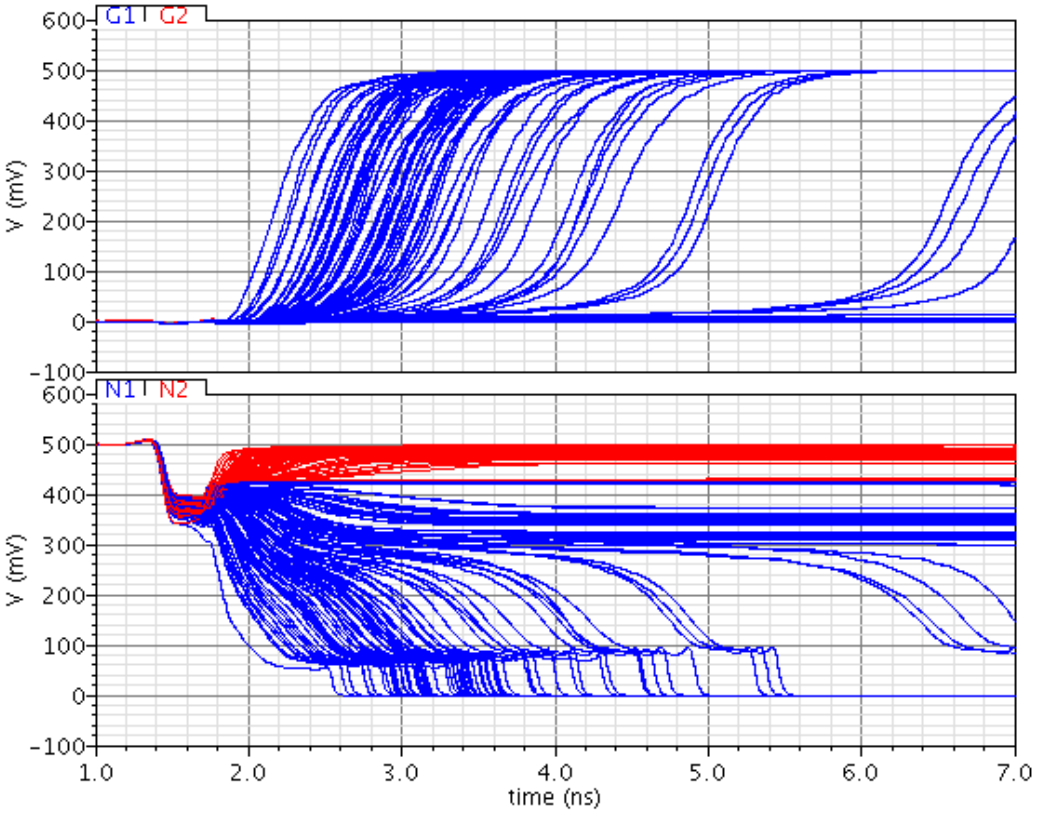
MMUTEX1B



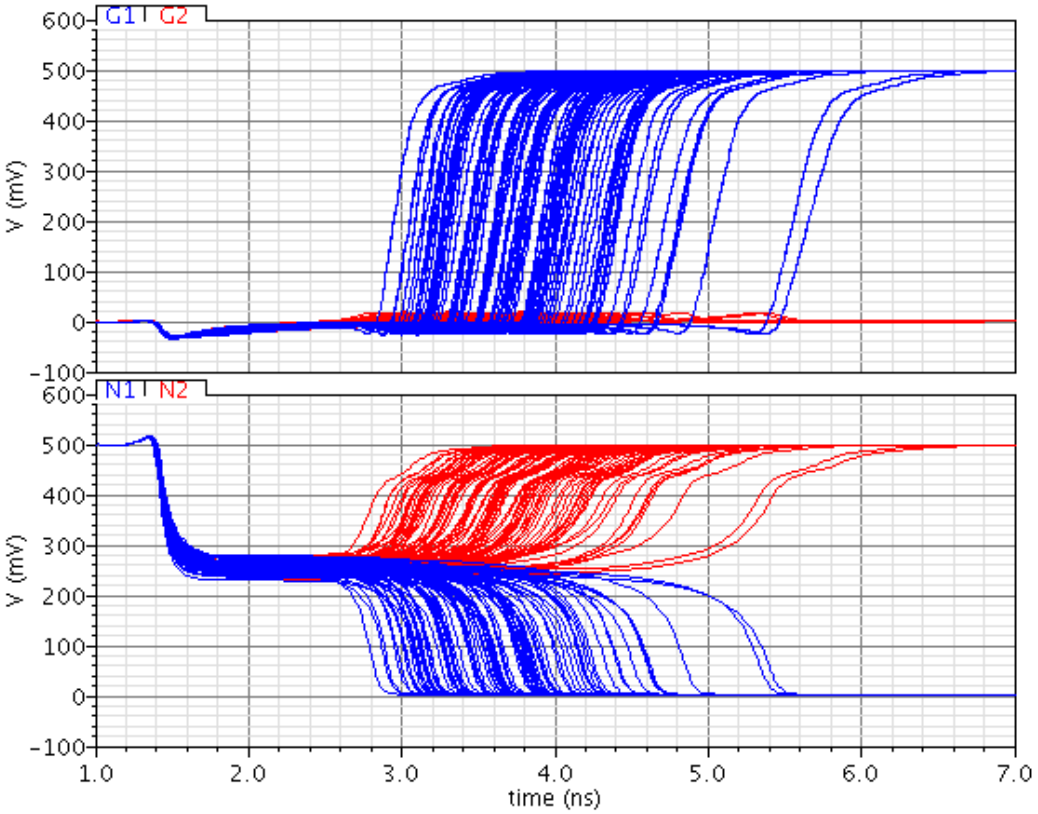
MMUTEX2A



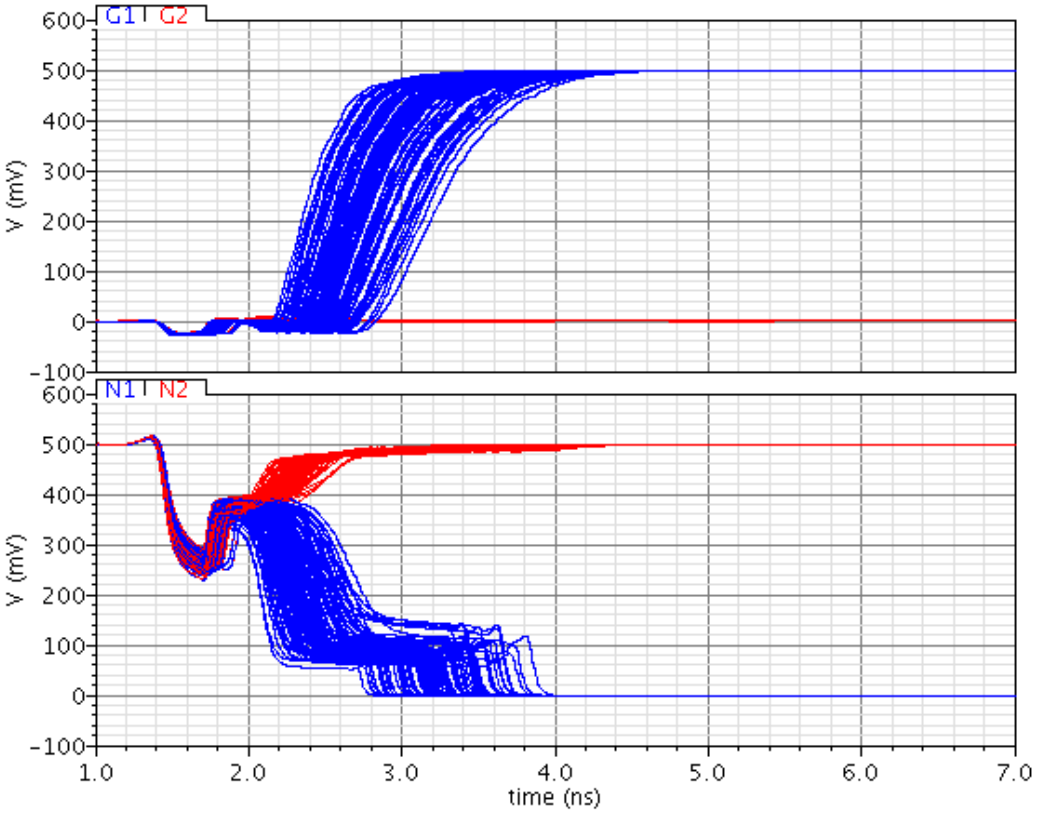
MMUTEX2B



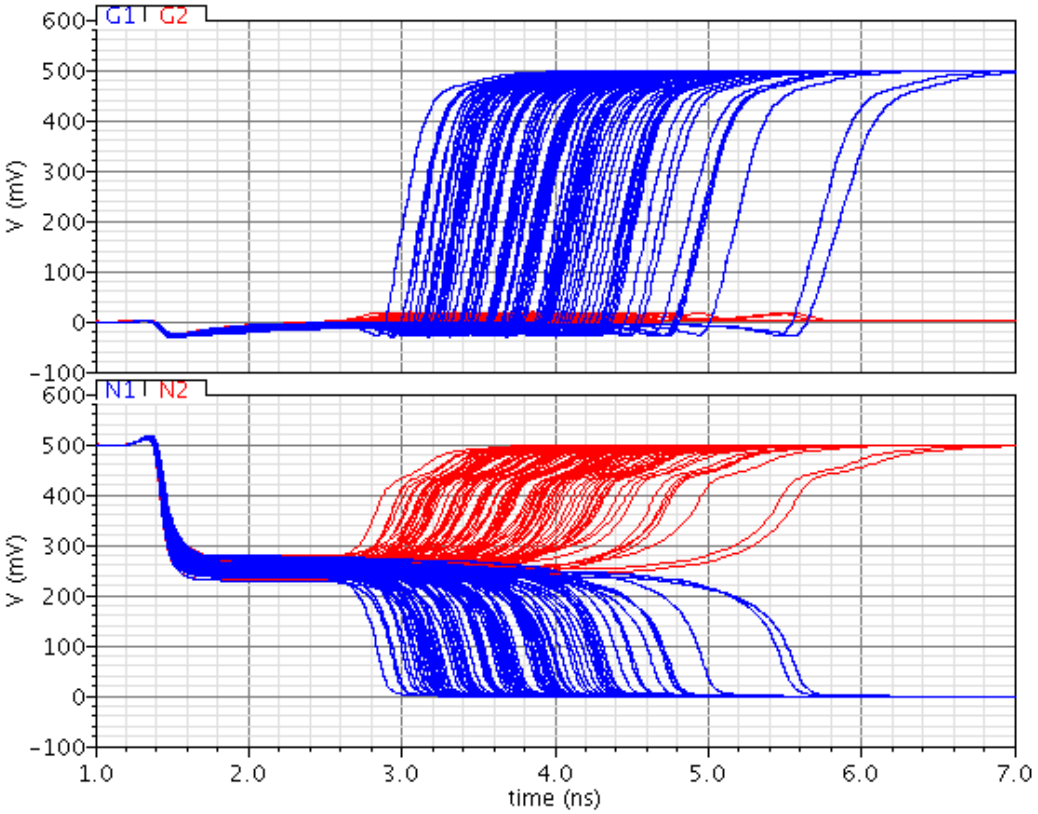
MMUTEX3A



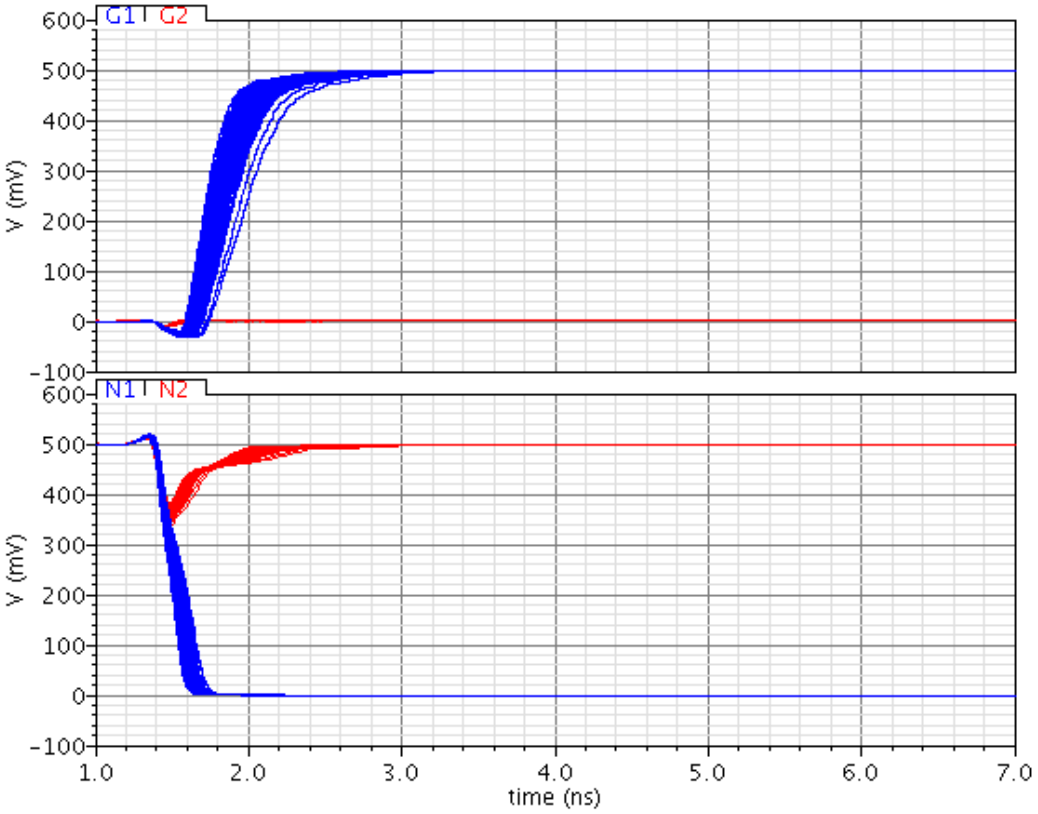
MMUTEX3B



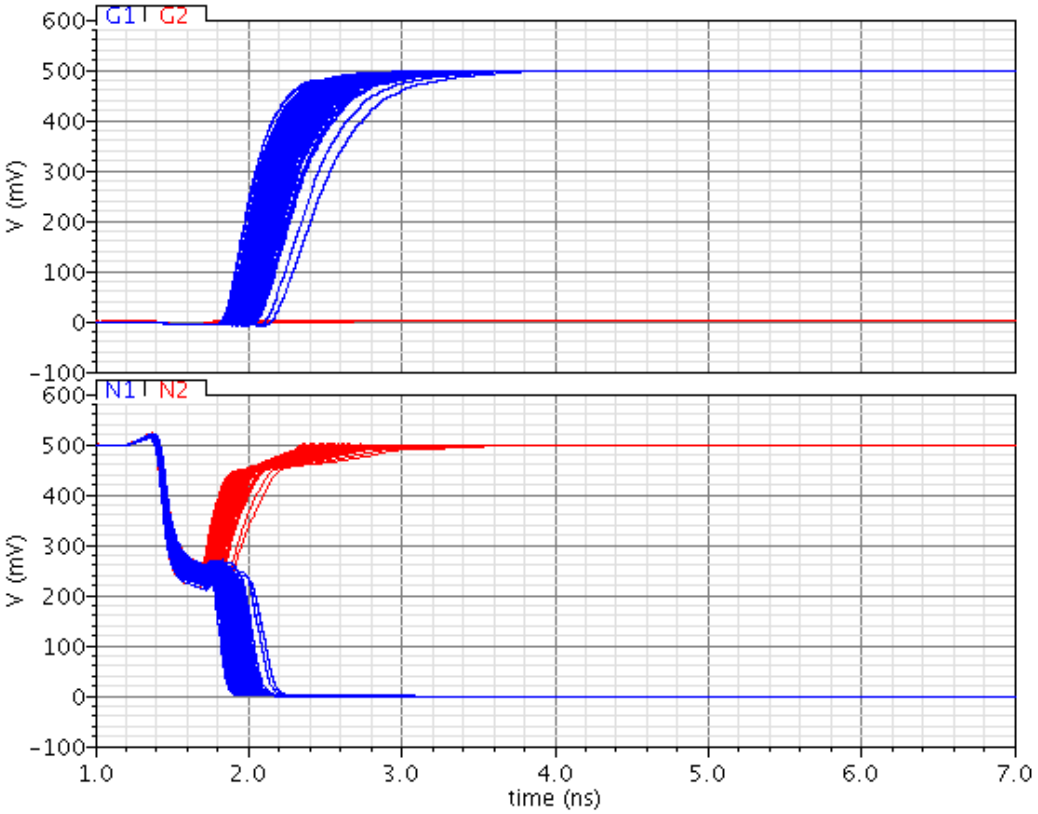
MMUTEX4A



MMUTEX5A



MMUTEX5B



Appendix E MADAC Technique Simulation Results

This appendix presents the simulation results of the circuits in Chapter 5.

E.1 Impact of Supply Voltage Reduction on τ

Table E.1 Voltage supply reduction impact on τ (ps)

τ (ps)	Without MADAC				With MADAC				
V_{DD}	C2MOS	TGFF	JLFF	RLFF	C2MOS	TGFF	JLFF	RLFF	RML
0.4	543.40	510.80	432.65	127.95	540.60	469.05	374.50	140.10	62.63
0.5	167.25	157.15	156.00	53.25	157.00	161.65	124.40	54.28	24.20
0.6	62.08	60.87	58.81	29.10	54.24	58.82	45.17	27.60	12.77
0.7	31.40	28.25	27.25	19.81	23.52	27.10	20.27	15.62	7.90
0.8	20.95	18.66	16.95	14.38	12.65	15.10	11.09	9.79	5.59
0.9	17.27	13.54	13.92	11.56	8.02	9.73	6.99	6.67	4.16
1.0	15.93	10.08	11.36	9.62	5.55	6.93	4.91	4.98	3.31
1.1	16.54	8.56	9.66	8.81	4.06	5.25	3.64	3.89	2.72
1.2	7.64	7.07	8.49	7.61	3.12	4.11	2.83	3.19	2.29

Table E.2 τ MADAC improvments

V_{DD}	C2MOS	TGFF	JLFF	RLFF	RML
0.4	-0.5%	-8.2%	-13.4%	9.5%	-51.1%
0.5	-6.1%	2.9%	-20.3%	1.9%	-54.6%
0.6	-12.6%	-3.4%	-23.2%	-5.2%	-56.1%
0.7	-25.1%	-4.1%	-25.6%	-21.2%	-60.1%
0.8	-39.6%	-19.1%	-34.6%	-31.9%	-61.1%
0.9	-53.6%	-28.1%	-49.8%	-42.3%	-64.0%
1.0	-65.2%	-31.2%	-56.8%	-48.2%	-65.6%
1.1	-75.5%	-38.6%	-62.3%	-55.8%	-69.1%
1.2	-59.1%	-41.9%	-66.7%	-58.1%	-69.9%

E.2 Impact of Supply Voltage Reduction on Delay Time

Table E.3 Voltage supply reduction impact on D to Q delay time (ps)

	D to Q delay time (ps)								
	Without MADAC				With MADAC				
V_{DD}	C2MOS	TGFF	JLFF	RLFF	C2MOS	TGFF	JLFF	RLFF	RML
0.4	1313.90	957.80	2310.90	2253.90	1408.10	1128.30	2811.00	2937.50	2395.90
0.5	542.30	394.90	693.95	865.75	583.20	467.20	850.30	1114.55	934.45
0.6	319.40	230.21	373.65	507.80	345.00	274.06	459.30	650.80	554.40
0.7	225.39	160.55	257.56	362.20	244.49	192.35	316.98	463.55	397.60
0.8	176.06	123.85	200.19	286.37	191.76	149.29	247.09	367.30	315.71
0.9	146.45	101.80	166.78	240.89	159.92	123.33	206.54	309.85	266.16
1.0	127.22	87.41	145.35	210.83	139.14	106.39	180.78	271.25	233.49
1.1	113.76	77.42	130.59	189.65	124.65	94.60	162.97	244.22	210.31
1.2	104.09	70.13	120.01	173.77	114.18	86.02	150.14	224.48	192.99

Table E.4 D to Q delay time MADAC impact

V_{DD}	C2MOS	TGFF	JLFF	RLFF	RML
0.4	7.2%	17.8%	21.6%	30.3%	6.3%
0.5	7.5%	18.3%	22.5%	28.7%	7.9%
0.6	8.0%	19.0%	22.9%	28.2%	9.2%
0.7	8.5%	19.8%	23.1%	28.0%	9.8%
0.8	8.9%	20.5%	23.4%	28.3%	10.2%
0.9	9.2%	21.1%	23.8%	28.6%	10.5%
1.0	9.4%	21.7%	24.4%	28.7%	10.7%
1.1	9.6%	22.2%	24.8%	28.8%	10.9%
1.2	9.7%	22.7%	25.1%	29.2%	11.1%

E.3 Impact of Supply Voltage Reduction on Setup plus Hold Time

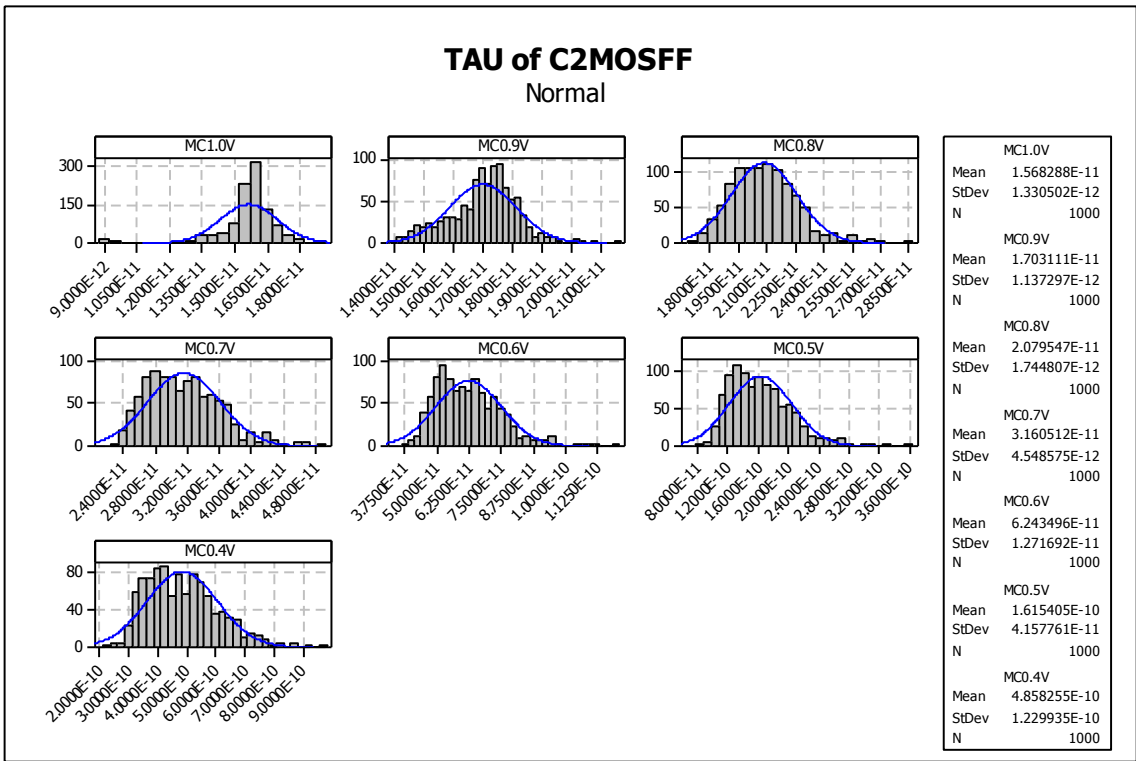
Table E.5 Voltage supply reduction impact on Setup plus Hold time (ps)

	Setup plus Hold time (ps)								
	Without MADAC				With MADAC				
V_{DD}	C2MOS	TGFF	JLFF	RLFF	C2MOS	TGFF	JLFF	RLFF	RML
0.4	322.30	124.00	970.97	972.72	549.00	94.50	1320.50	1492.44	322.30
0.5	131.86	51.89	266.67	362.73	223.10	39.80	357.71	538.12	131.86
0.6	76.53	29.26	133.59	209.53	129.22	21.82	177.67	305.72	76.53
0.7	53.28	19.48	88.75	148.39	90.39	14.07	116.78	214.58	53.28
0.8	41.19	14.27	67.52	117.02	70.43	9.95	88.52	169.01	41.19
0.9	33.93	11.12	55.53	98.45	59.14	7.53	72.79	142.32	33.93
1.0	29.19	9.02	48.04	86.44	52.15	5.94	63.20	124.32	29.19
1.1	25.86	7.55	43.04	77.77	47.54	4.86	56.80	111.67	25.86
1.2	23.46	6.42	39.48	71.17	44.46	4.04	52.20	102.63	23.46

Table E.6 Setup plus Hold time MADAC impact

V_{DD}	C2MOS	TGFF	JLFF	RLFF	RML
0.4	70.3%	-23.8%	36.0%	53.4%	-2.3%
0.5	69.2%	-23.3%	34.1%	48.4%	-1.3%
0.6	68.8%	-25.4%	33.0%	45.9%	-0.2%
0.7	69.7%	-27.8%	31.6%	44.6%	0.1%
0.8	71.0%	-30.3%	31.1%	44.4%	0.2%
0.9	74.3%	-32.3%	31.1%	44.6%	0.1%
1.0	78.7%	-34.1%	31.6%	43.8%	0.0%
1.1	83.8%	-35.6%	32.0%	43.6%	0.0%
1.2	89.5%	-37.0%	32.2%	44.2%	0.0%

E.4 Impact of Process Variations on τ



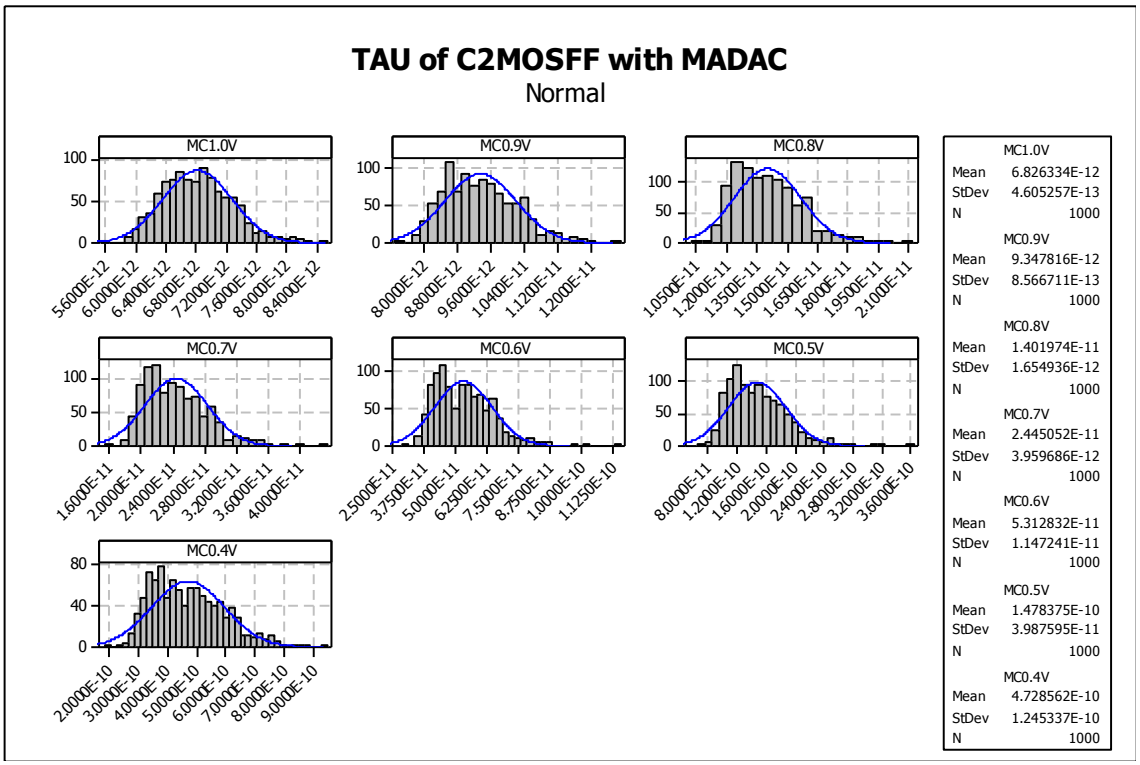
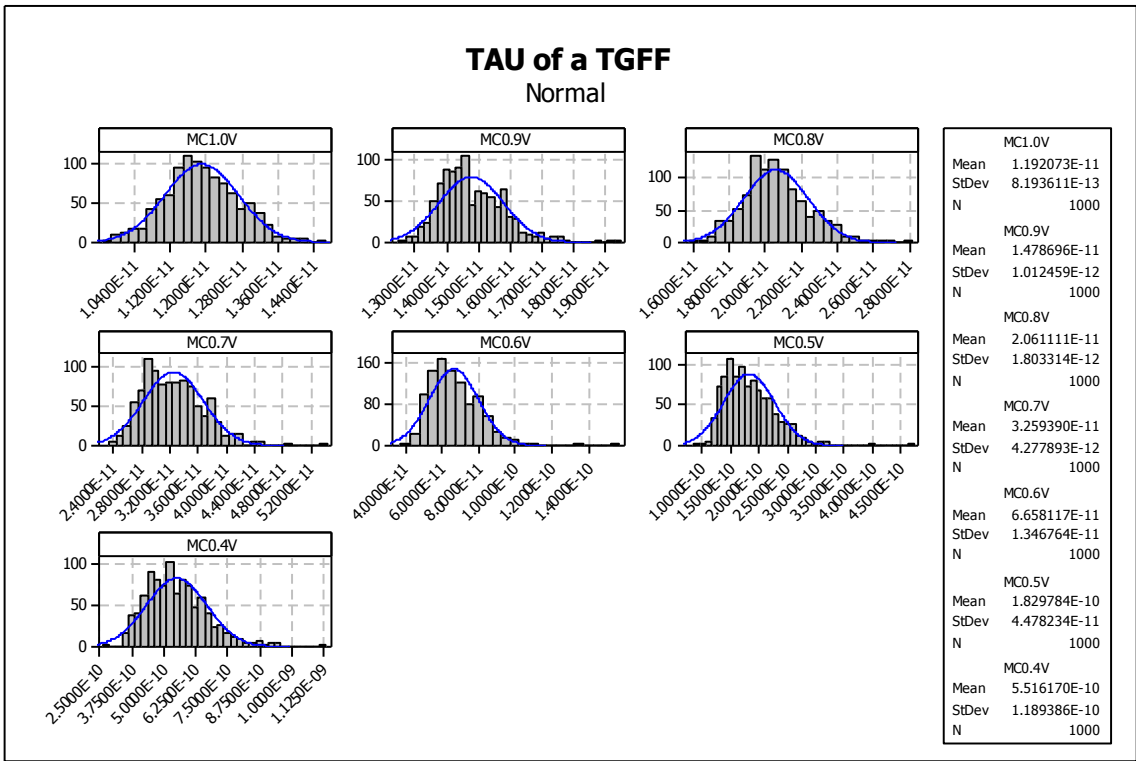


Figure E.1 Histograms of the impact Process Variation and Voltage on τ of the C2MOSFF without and with MADAC



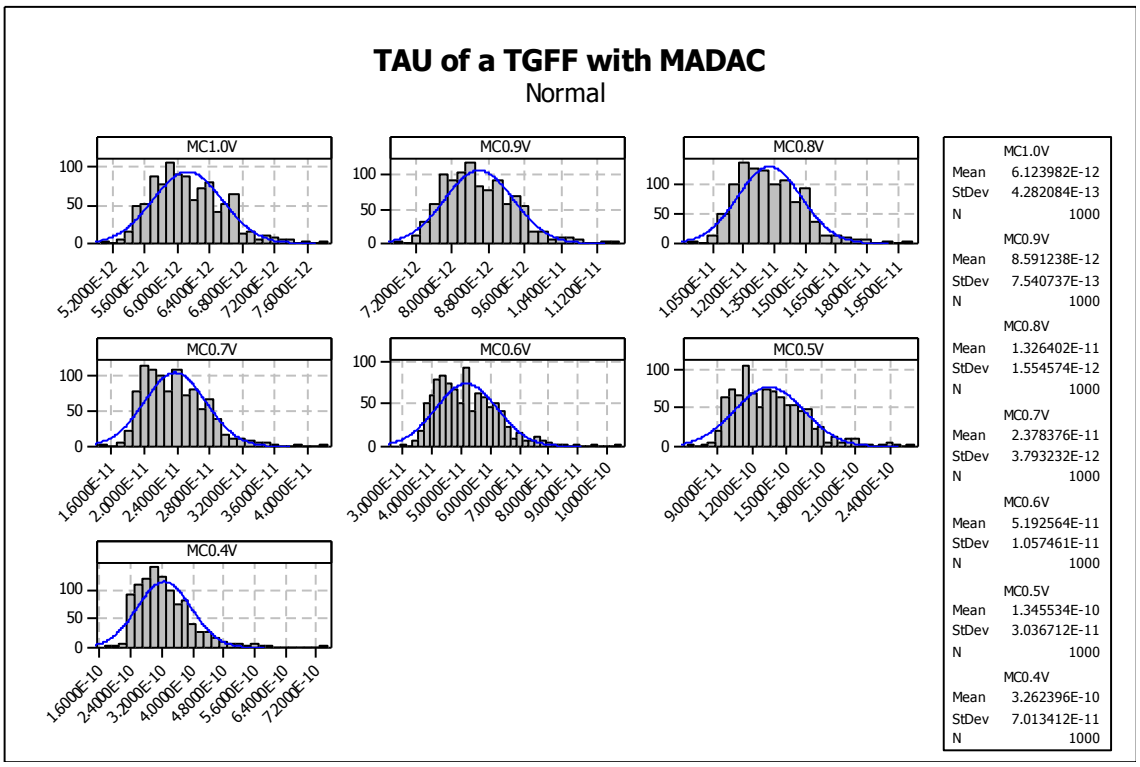
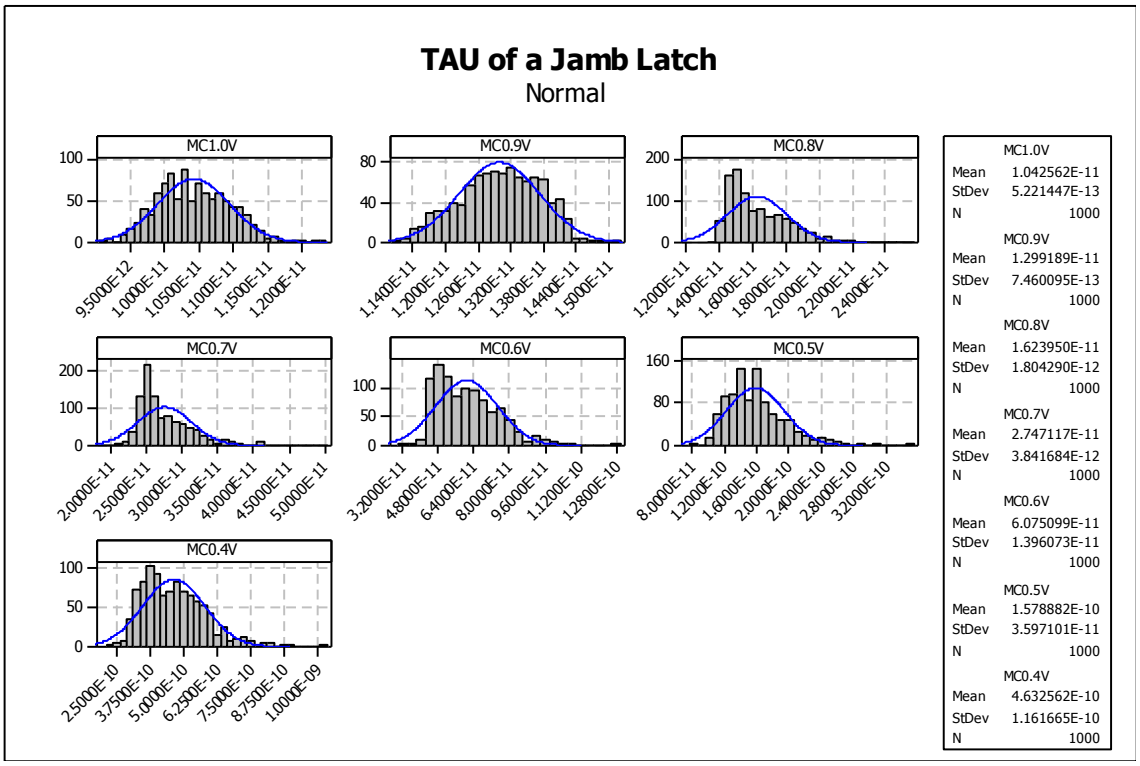


Figure E.2 Histograms of the impact Process Variation and Voltage on τ of the TGFF without and with MADAC



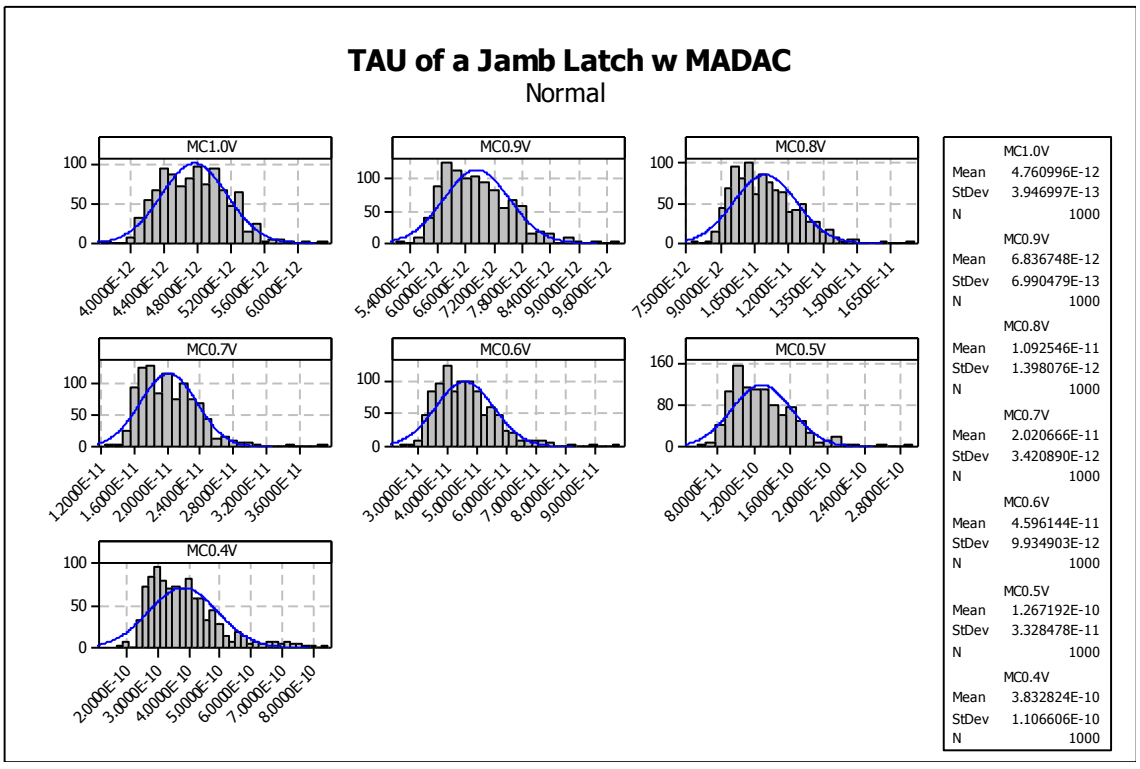
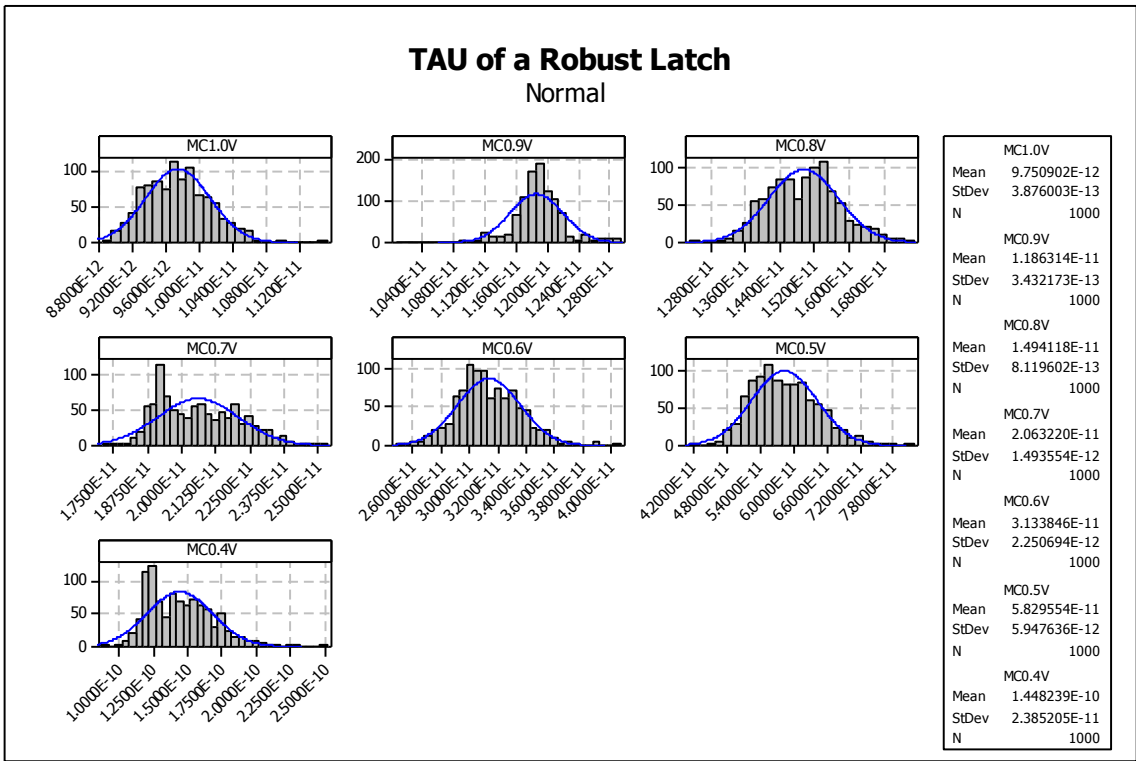


Figure E.3 Histograms of the impact Process Variation and Voltage on τ of the Jamb Latch without and with MADAC



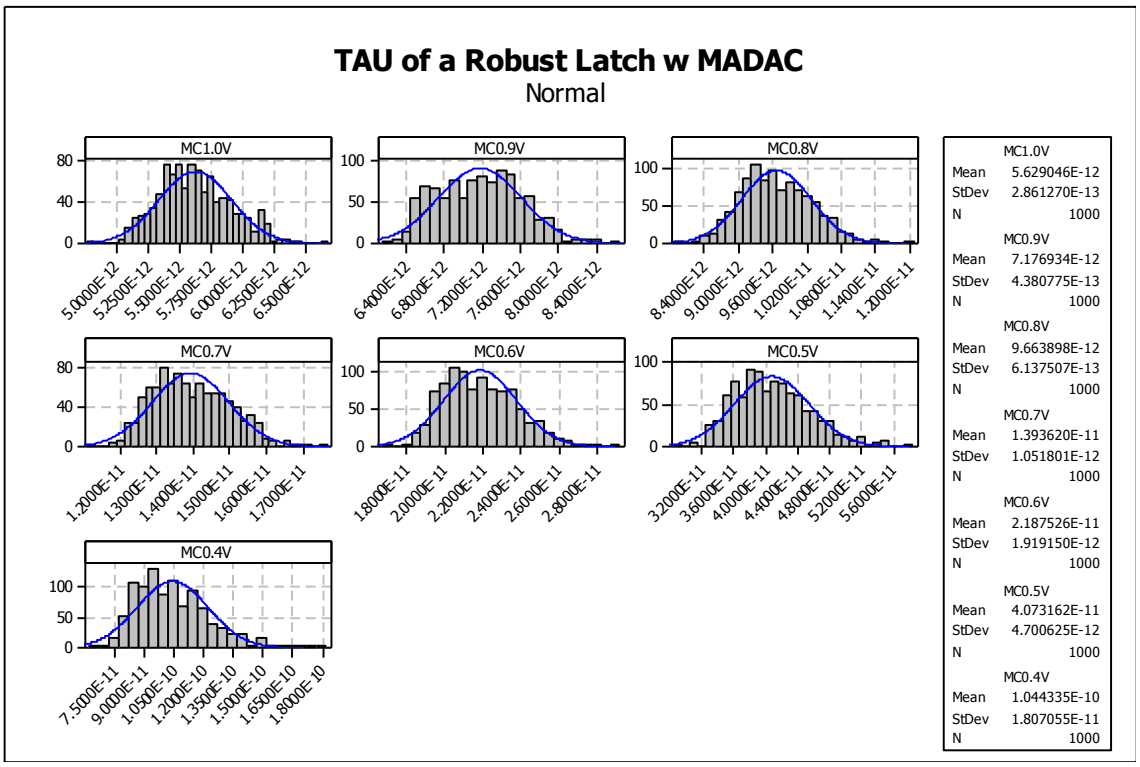


Figure E.4 Histograms of the impact Process Variation and Voltage on τ of the Robust Latch without and with MADAC

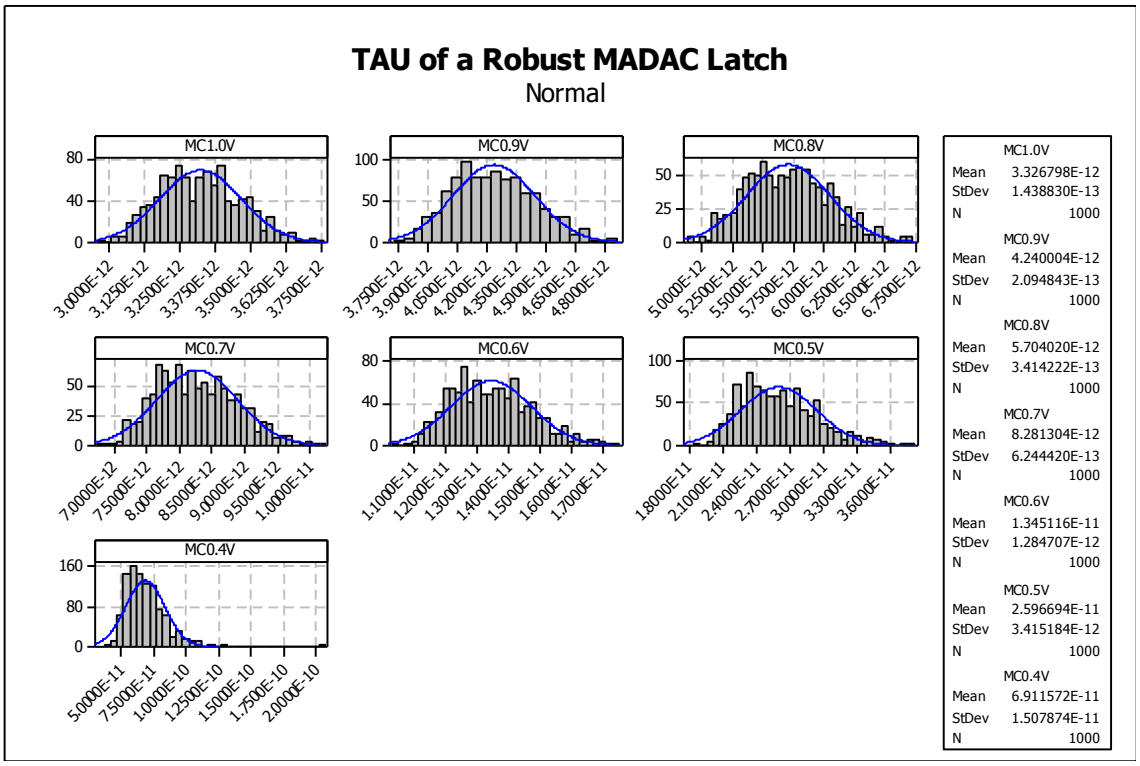


Figure E.5 Histograms of the impact Process Variation and Voltage on τ of the Robust MADAC Latch

Appendix F LSHS 2 Simulations

Simulations waveforms of LSHS 2: Upshifting 0.8V to 1.2V

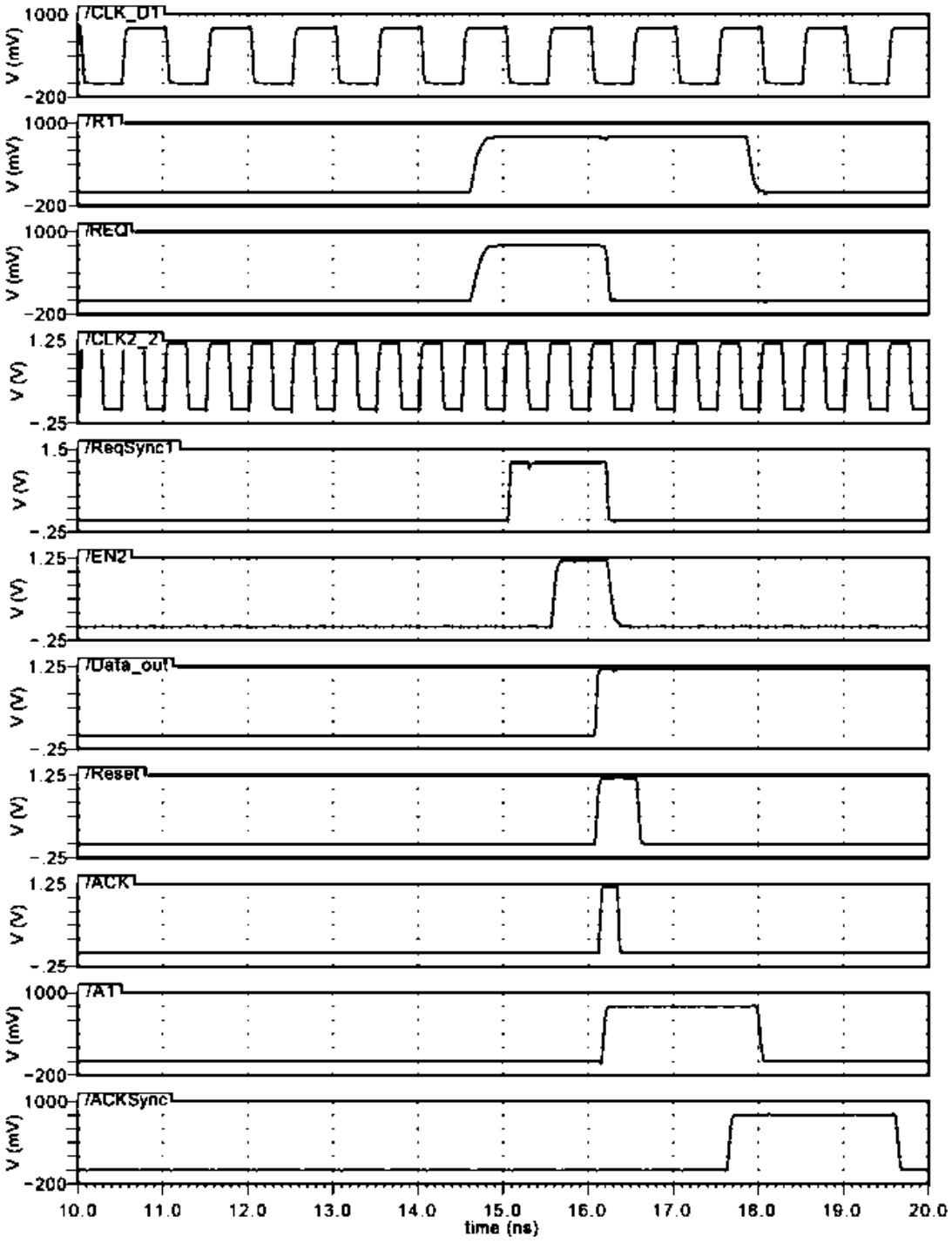


Figure F.1 LSHS2 waveforms: upshifting 0.8V to 1.2V

Simulations waveforms of LSHS 2:

Downshifting 1.2V to 0.8V

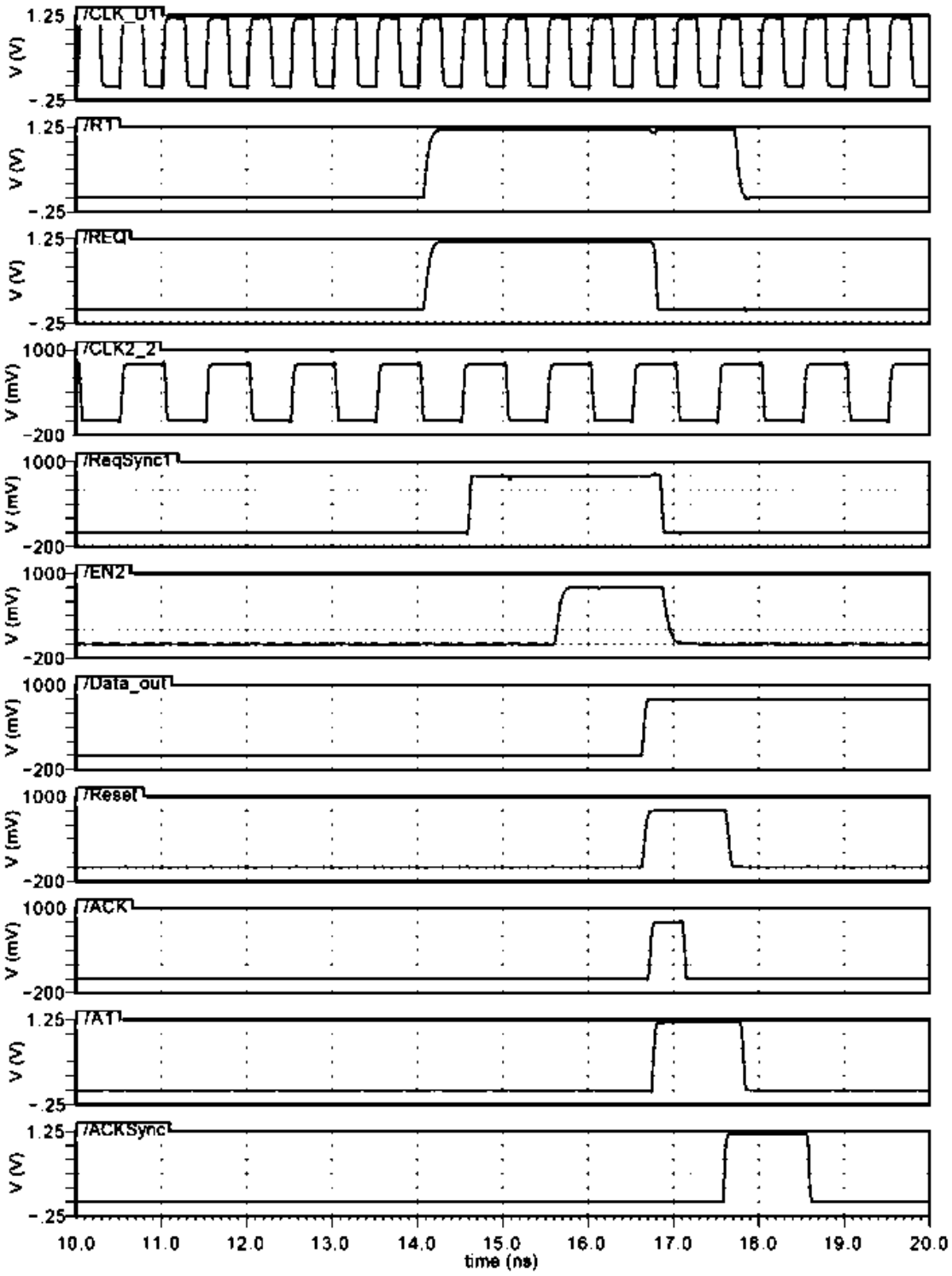


Figure F.2 PLSHS2 waveforms downshifting 1.2V to 0.8V

Simulations waveforms of LSHS 2:

Slow-to-fast transfer: Tclk1=5ns and Tclk2=500ps

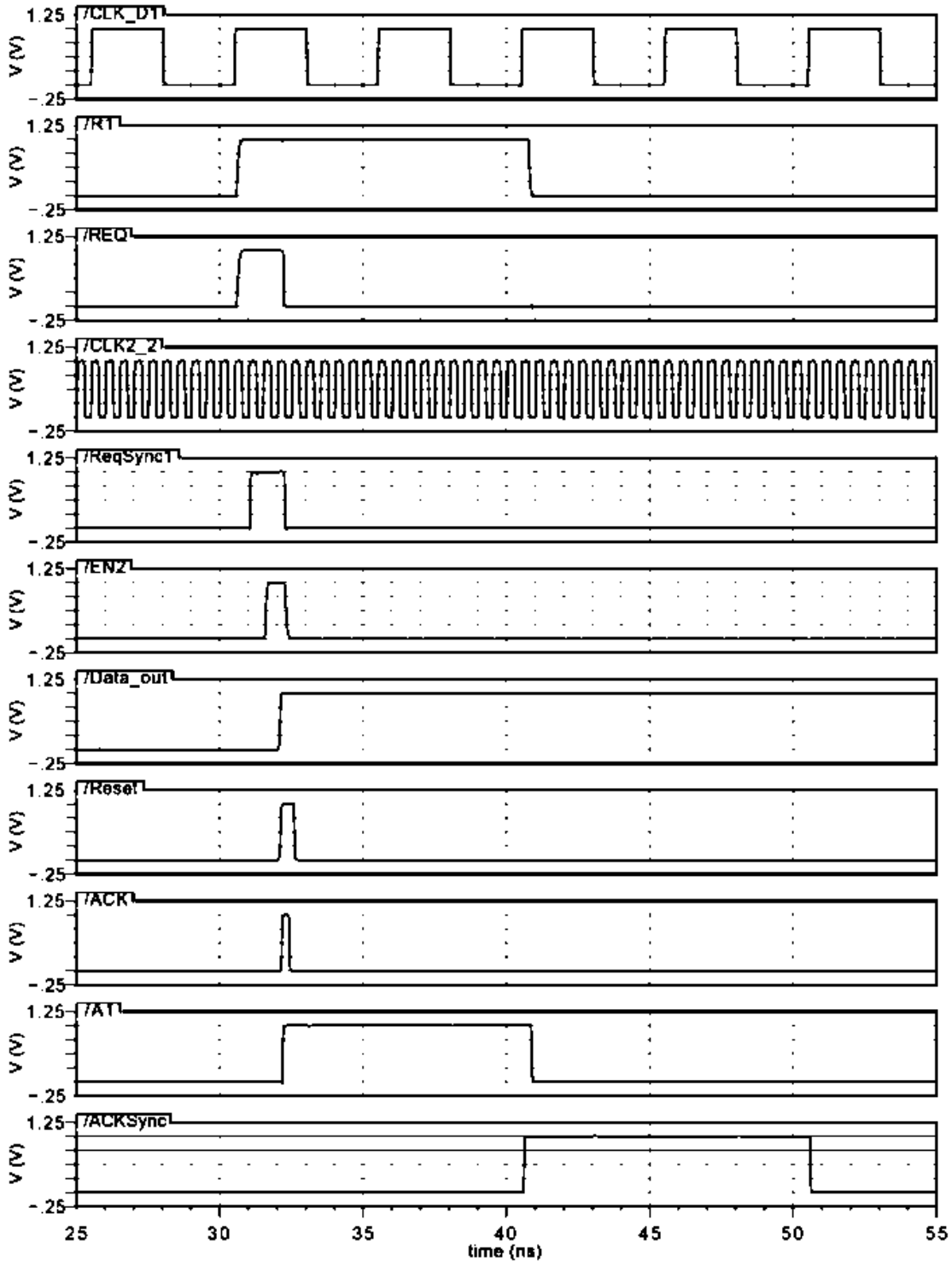


Figure F.3 PLSHS2 slow-to-fast transfer: 5ns to 500ps

Simulations waveforms of LSHS 2:

Slow-to-fast transfer: Tclk1=10ns and Tclk2=500ps

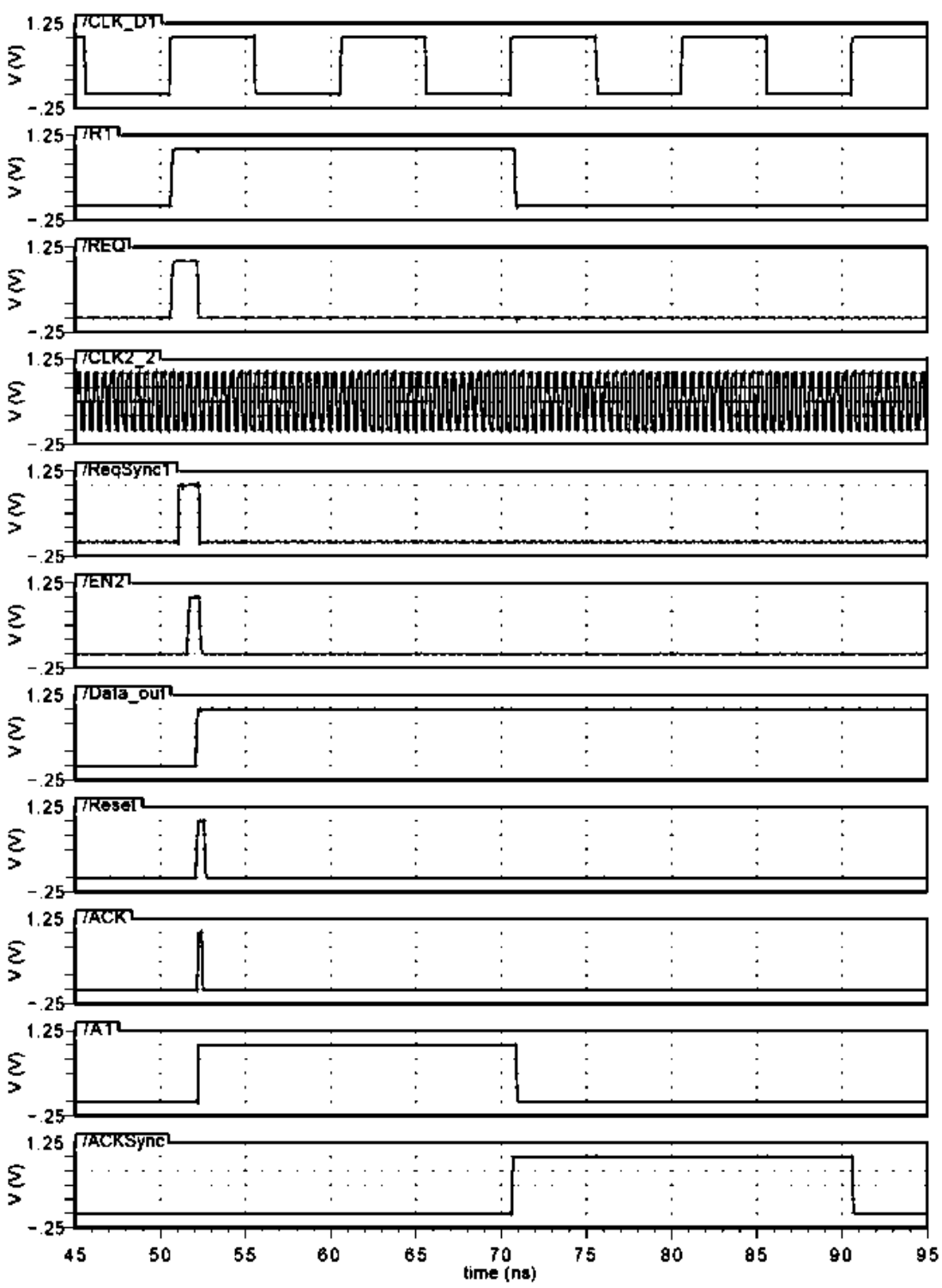


Figure F.4 PLSHS2 slow-to-fast transfer: 10ns to 500ps

Simulations waveforms of LSHS 2:

Fast-to-slow transfer: Tclk1=500ps and Tclk2=5ns

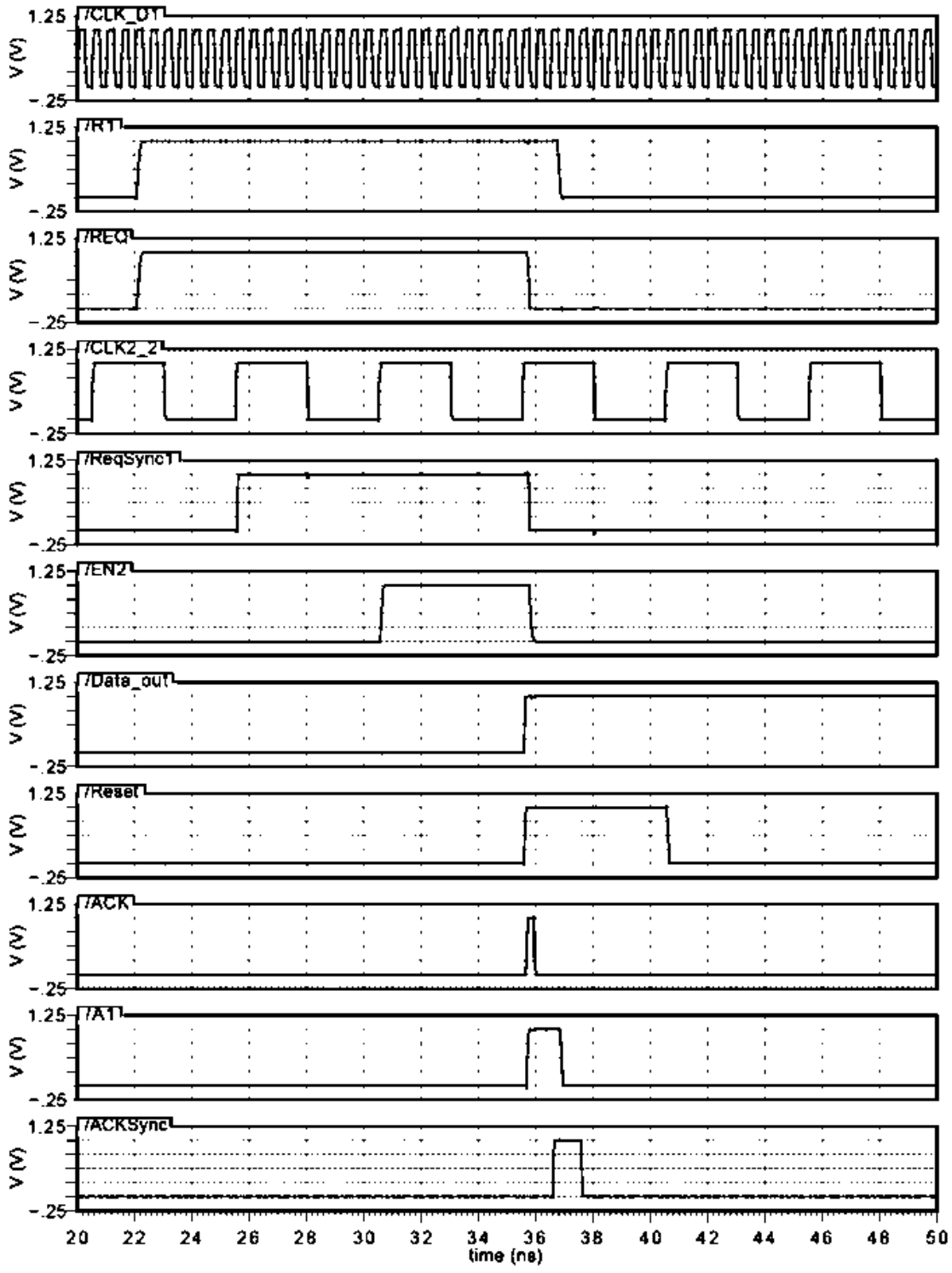


Figure F.5 PLSHS2 fast-to-slow transfer: 500ps to 5ns

Simulations waveforms of LSHS 2:

Fast-to-slow transfer: Tclk1=500ps and Tclk2=10ns

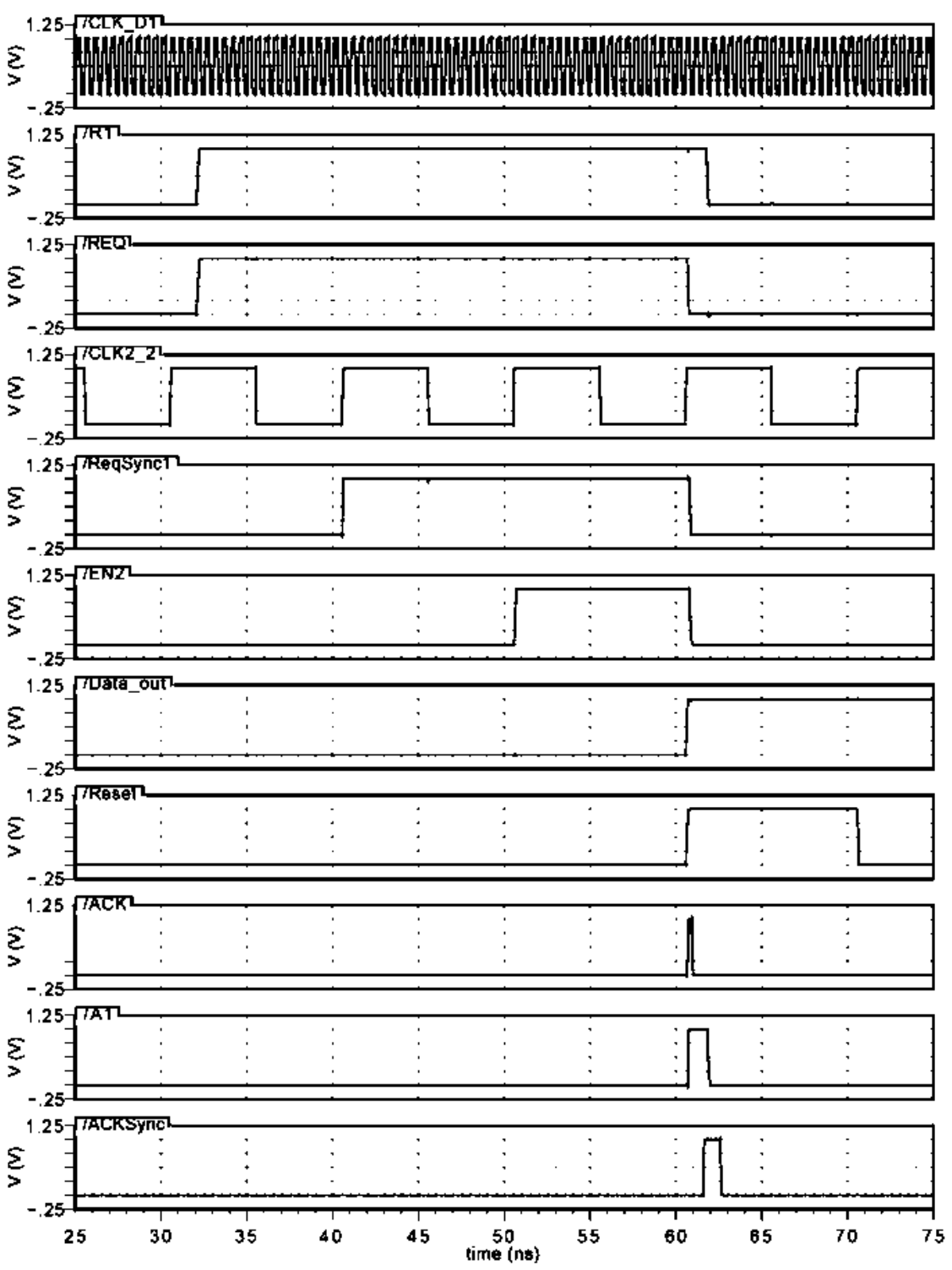


Figure F.6 PLSHS2 fast-to-slow transfer 500ps to 10ns

Simulations waveforms of LSHS 2:

PLSHS2 with domains run at the same clock rate (5ns) and the supply voltage is a varying sine wave at around Vdc=1V with amplitude of 0.4×Vdc and 25MHz on domain1 and 50MHz in domain2.

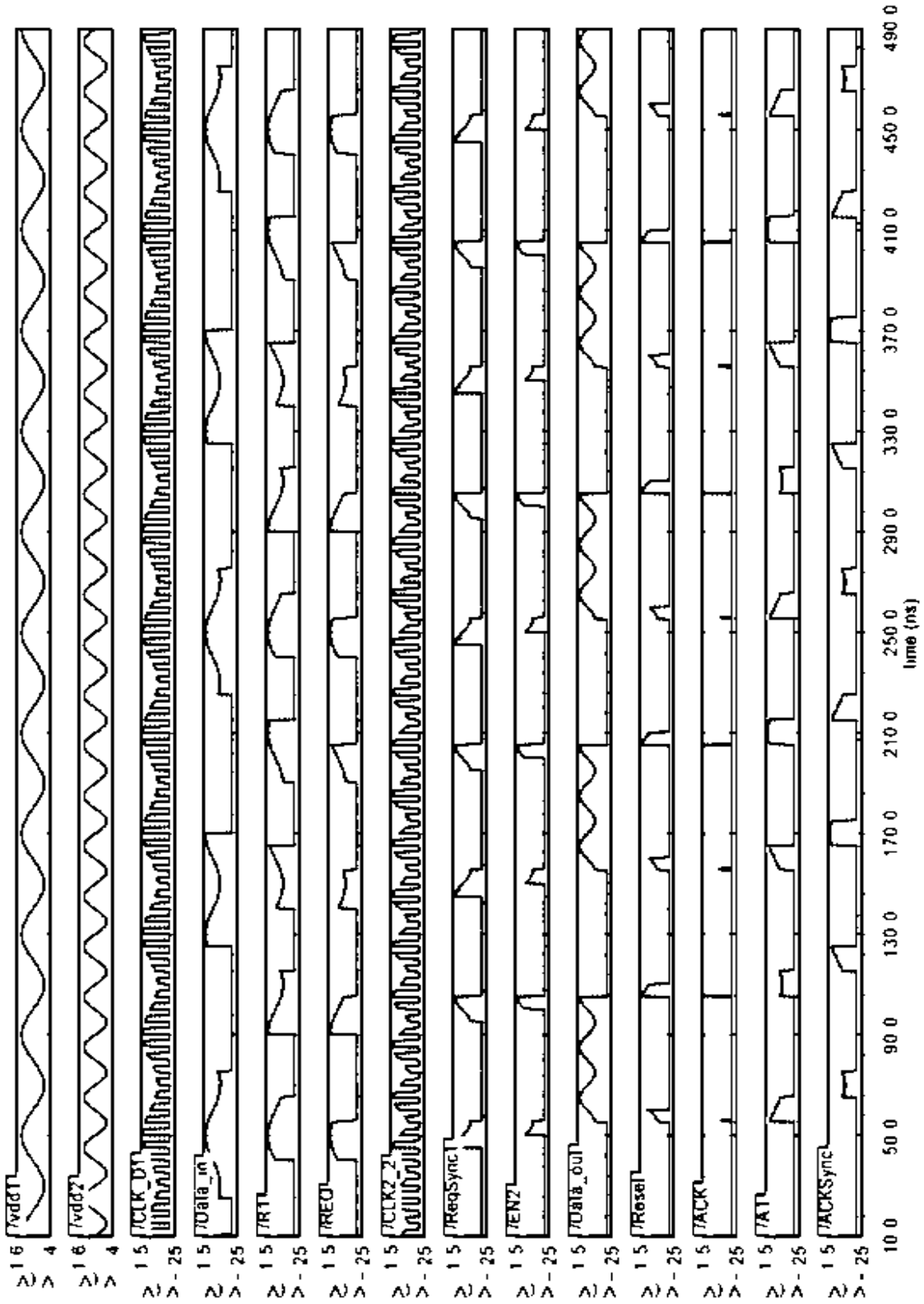


Figure F.7 PLSHS2 with variable V_{DD} sine wave.

Simulations waveforms of LSHS 2:

PLSHS2 under variable supply voltages: slow-to-fast transfer.

Domain 1: V_{DD1} = 0.8 (0.4 sin(2π25Mhz)+1), $Tclk1$= 5ns Clock

Domain 2: V_{DD2} = 1.2 (0.4 sin(2π50Mhz)+1), $Tclk2$= 1ns Clock

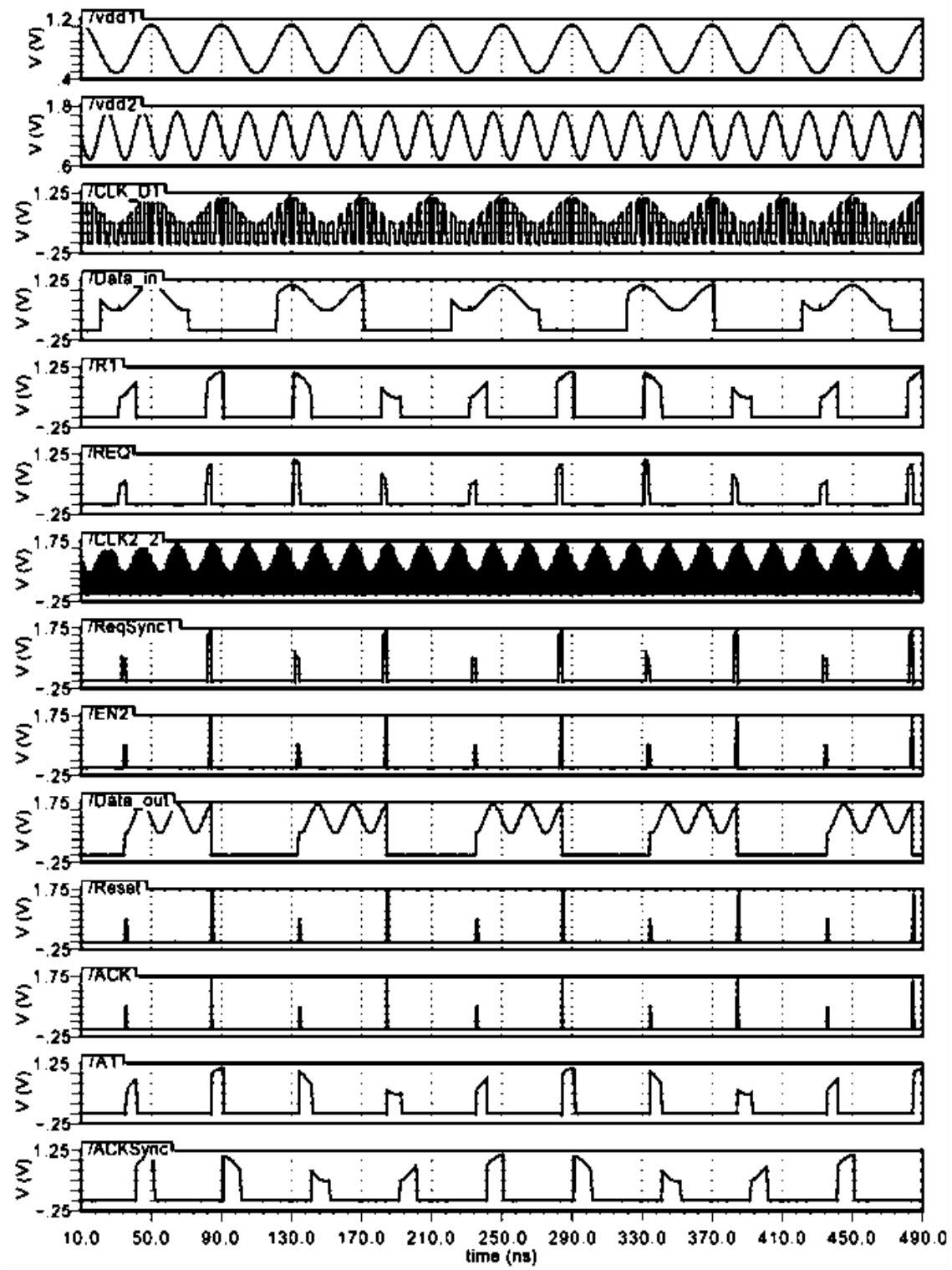


Figure F.8 PLSHS2 under variable supply voltages and slow-to-fast transfer.

Simulations waveforms of LSHS 2:

PLSHS2 under variable supply voltages: fast-to-slow transfer.

Domain 1: V_{DD1} = 1.2 (0.4 sin(2π25Mhz)+1), $Tclk1$= 1ns Clock

Domain 2: V_{DD2} = 0.8 (0.4 sin(2π50Mhz)+1), $Tclk2$= 5ns Clock

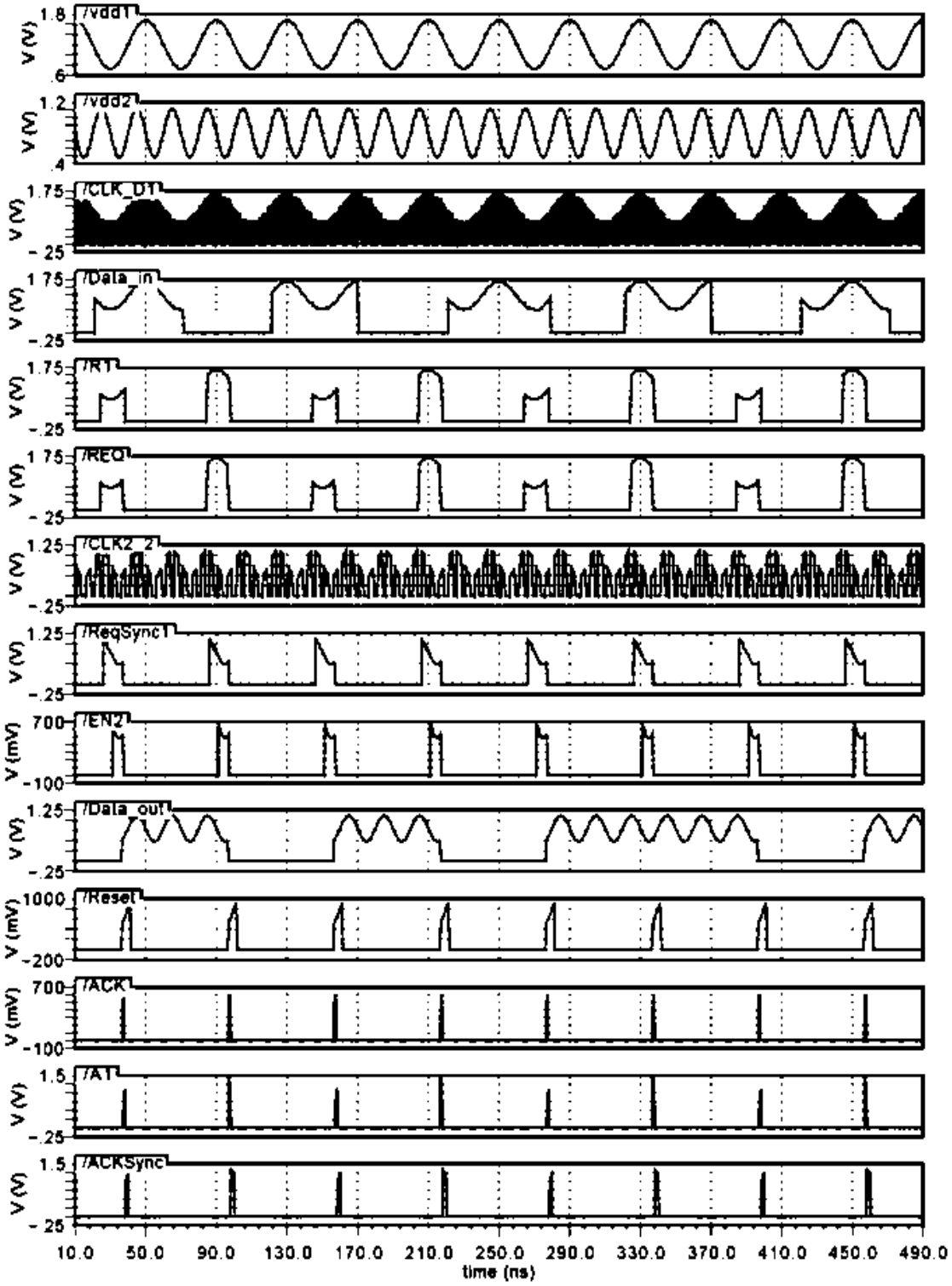


Figure F.9 PLSHS2 under variable supply voltages and fast-to-slow transfer.

Simulations waveforms of LSHS 2:

PLSHS2 emulation under DVFS effect: DC level at 1V

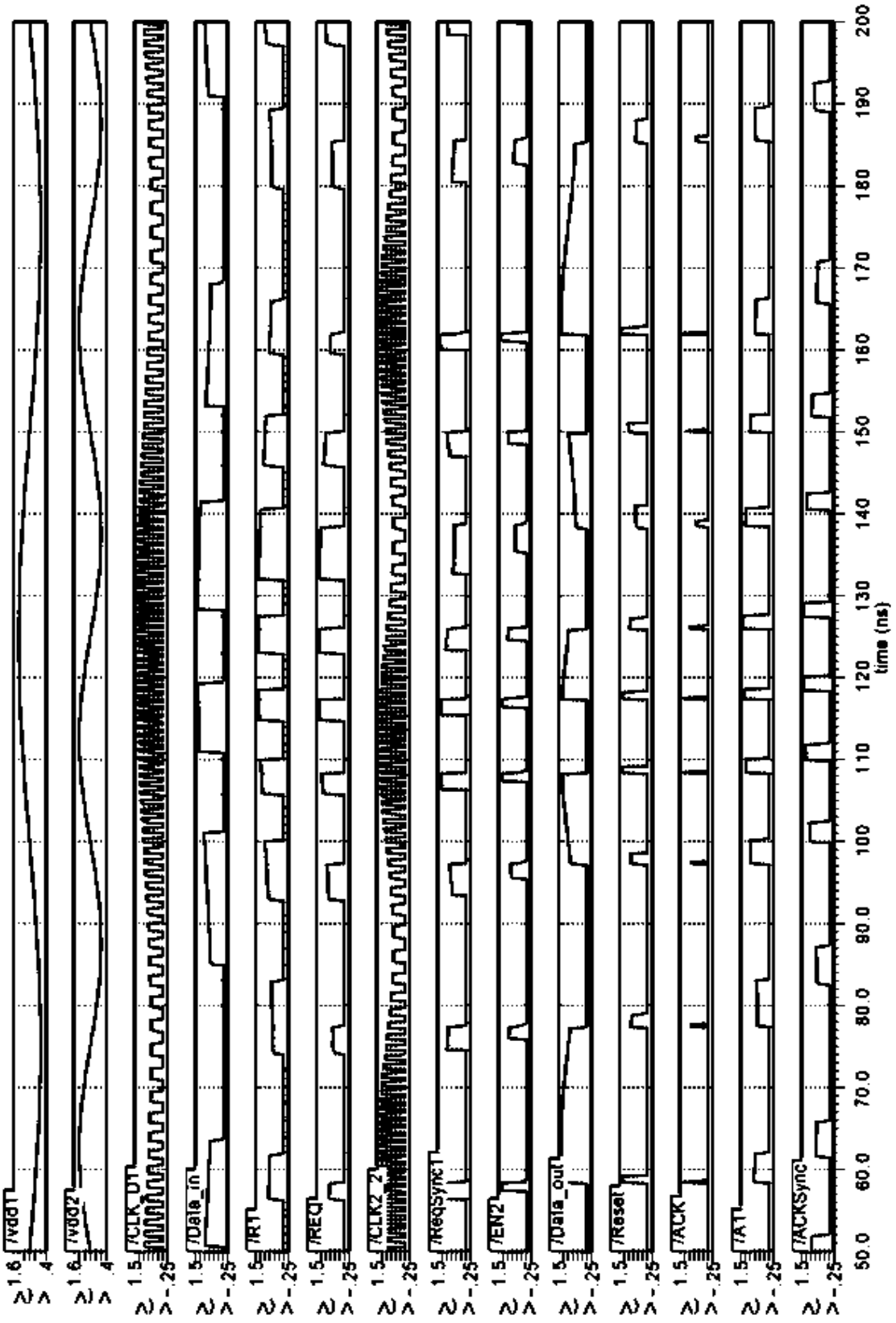


Figure F.10 PLSHS2 emulation under DVFS effect: DC level at 1V

Simulations waveforms of LSHS 2:

PLSHS2 emulation under DVFS effect: DC levels at 0.8V(D1) and 1.2V(D2)

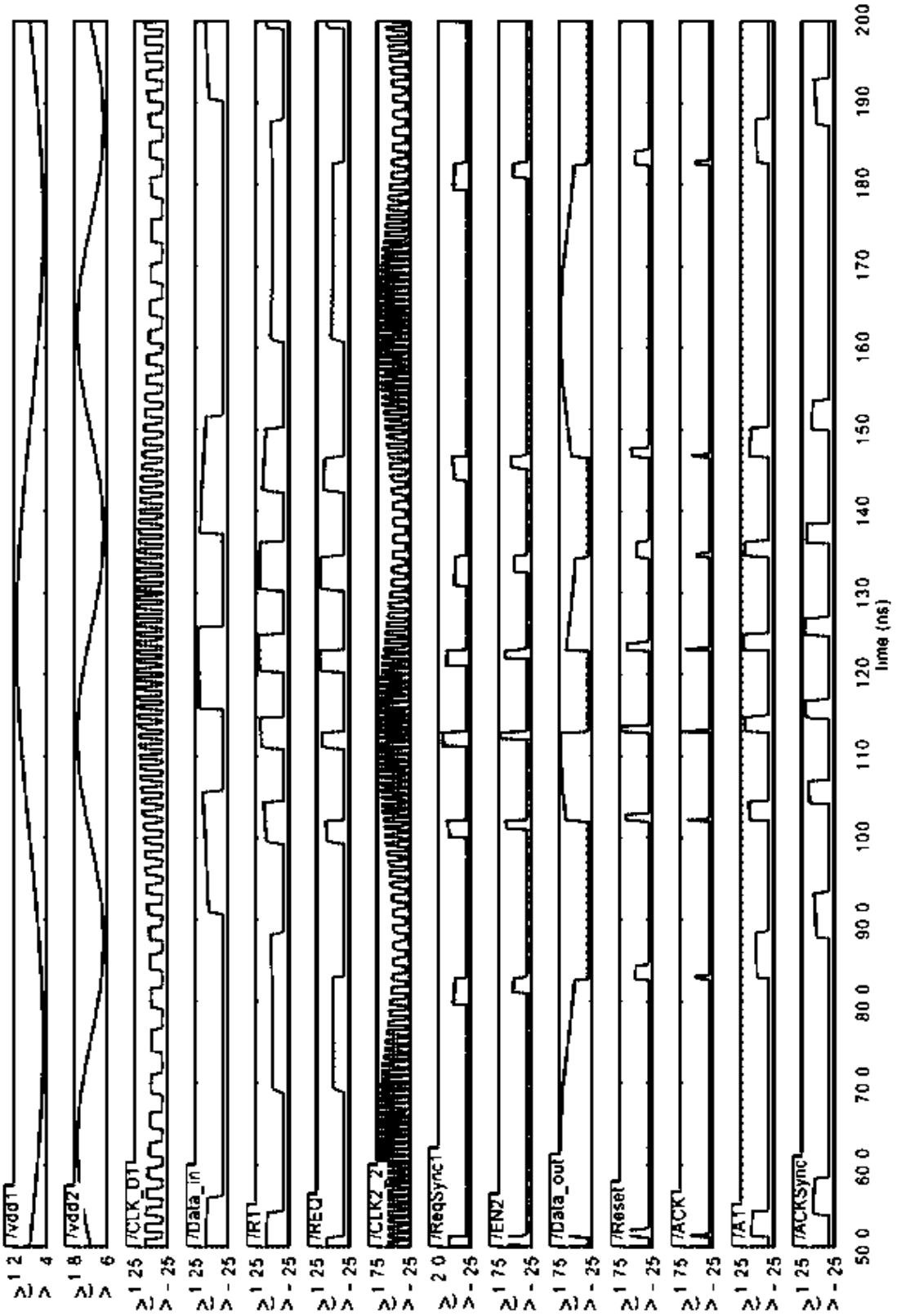


Figure F.11 PLSHS2 emulation under DVFS effect: DC levels at 0.8V(D1) and 1.2V(D2)

Simulations waveforms of LSHS 2:

PLSHS2 emulation under DVFS effect: DC levels at 1.2V(D1) and 0.8V(D2)

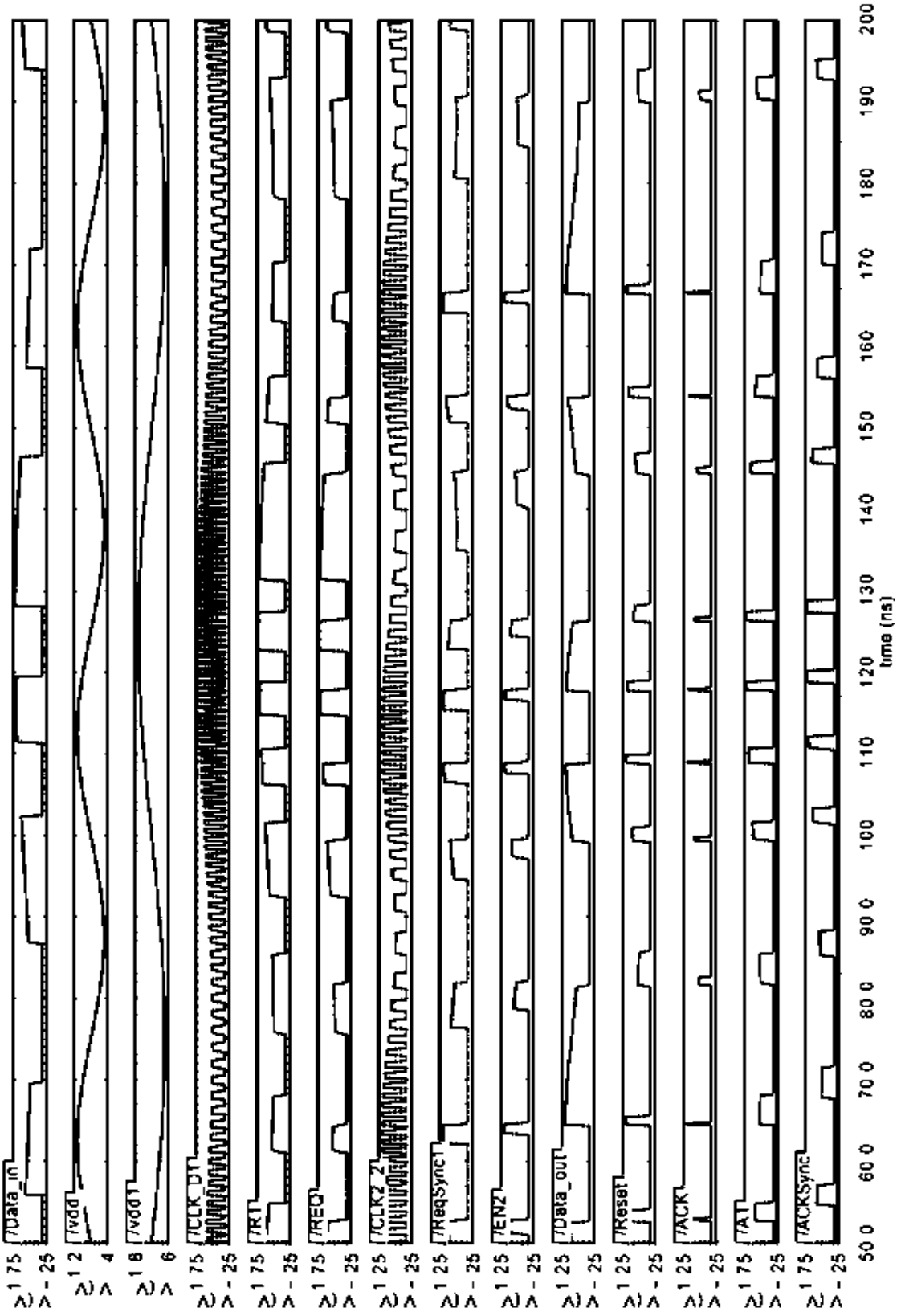


Figure F.12 PLSHS2 emulation under DVFS effect: DC levels at 1.2V(D1) and 0.8V(D2)

References

[1] *The International Technology Roadmap for Semiconductors (ITRS) (2007-2011 ed.)*. Available: http://public.itrs.net

[2] S. Borkar, "Design Challenges of Technology Scaling," *IEEE Micro,* vol. 19, pp. 23-29, 1999.

[3] "Executive summary reports," in *The International Technology Roadmap for Semiconductors (ITRS)*, 2007-2011 ed. http://public.itrs.net

[4] N. H. E. Weste and D. M. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*, 4th ed. Boston: Pearson/Addison-Wesley, 2010.

[5] J. M. Rabaey, A. P. Chandrakasan, and B. Nikolic, *Digital integrated circuits : a design perspective*, 2nd ed. N.J.: Prentice Hall, 2003.

[6] (10/05/2012). *UMC technology overview, EUROPRACTICE IC Service*. Available: http://www.europractice-ic.com/technologies_UMC.php

[7] S. R. Nassif, "Modeling and analysis of manufacturing variations," in *Custom Integrated Circuits, 2001, IEEE Conference on.*, 2001, pp. 223-228.

[8] A. Chandrakasan, W. J. Bowhill, and F. Fox, *Design of high-performance microprocessor circuits*. New York: IEEE Press, 2001.

[9] A. Srivastava, D. Sylvester, and D. Blaauw, *Statistical analysis and optimization for VLSI : timing and power*. New York: Springer, 2005.

[10] A. A. Mutlu and M. Rahman, "Statistical methods for the estimation of process variation effects on circuit operation," *Electronics Packaging Manufacturing, IEEE Transactions on,* vol. 28, pp. 364-375, 2005.

[11] K. Bernstein, D. J. Frank, A. E. Gattiker, W. Haensch, B. L. Ji, S. R. Nassif, *et al.*, "High-performance CMOS variability in the 65-nm regime and beyond," *IBM Journal of Research and Development,* vol. 50, pp. 433-449, 2006.

[12] K. Bernstein, D. J. Frank, A. E. Gattiker, W. Haensch, B. L. Ji, S. R. Nassif, *et al.*, "High-performance CMOS variability in the 65-nm regime and beyond," *IBM J. Res. Dev.,* vol. 50, pp. 433-449, 2006.

[13] K. J. Kuhn, "Reducing Variation in Advanced Logic Technologies: Approaches to Process and Design for Manufacturability of Nanoscale CMOS," in *Electron Devices Meeting, 2007. IEDM 2007. IEEE International*, 2007, pp. 471-474.

[14] S. Borkar, "Designing reliable systems from unreliable components: the challenges of transistor variability and degradation," *Micro, IEEE,* vol. 25, pp. 10-16, 2005.

[15] M. Bohr, "The new era of scaling in an SoC world," in *Solid-State Circuits Conference - Digest of Technical Papers, 2009. ISSCC 2009. IEEE International*, 2009, pp. 23-28.

[16] "Design reports," in *The International Technology Roadmap for Semiconductors (ITRS)*, 2007-2011 ed. http://public.itrs.net

[17] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture," in *Design Automation Conference, 2003. Proceedings*, 2003, pp. 338-342.

[18] K. A. Bowman, S. G. Duvall, and J. D. Meindl, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *Solid-State Circuits, IEEE Journal of,* vol. 37, pp. 183-190, 2002.

[19] B. P. Wong, A. Mittal, Y. Cao, and G. Starr, *Nano-CMOS circuit and physical design*. N.J.: J. Wiley & Sons, Inc., 2005.

[20] K. Bowman, J. Tschanz, C. Wilkerson, L. Shih-Lien, T. Karnik, V. De, *et al.*, "Circuit techniques for dynamic variation tolerance," in *Design Automation Conference, 2009. DAC '09. 46th ACM/IEEE*, 2009, pp. 4-7.

[21] M. Ikeda, K. Ishii, T. Sogabe, and K. Asada, "Datapath Delay Distributions for Data/Instruction against PVT Variations in 90nm CMOS," in *Electronics, Circuits and Systems, 2007. ICECS 2007. 14th IEEE International Conference on*, 2007, pp. 154-157.

[22] K. Bowman, J. Tschanz, C. Wilkerson, S.-L. Lu, T. Karnik, and V. De, "Resilient Circuits for Dynamic Variation Tolerance," presented at the 2009 CMOS Emerging Technologies Workshop, Vancouver, Canada, 2009.

[23] S. Narendra, D. Antoniadis, and V. De, "Impact of using adaptive body bias to compensate die-to-die Vt variation on within-die Vt variation," in *Low Power Electronics and Design, 1999. Proceedings. 1999 International Symposium on*, 1999, pp. 229-232.

[24] J. W. Tschanz, J. T. Kao, S. G. Narendra, R. Nair, D. A. Antoniadis, A. P. Chandrakasan*, et al.*, "Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage," *Solid-State Circuits, IEEE Journal of,* vol. 37, pp. 1396-1402, 2002.

[25] J. W. Tschanz, S. Narendra, R. Nair, and V. De, "Effectiveness of adaptive supply voltage and body bias for reducing impact of parameter variations in low power and high performance microprocessors," *Solid-State Circuits, IEEE Journal of,* vol. 38, pp. 826-829, 2003.

[26] T. Rahal-Arabi, G. Taylor, M. Ma, and C. Webb, "Design and validation of the Pentium III and Pentium 4 processors power delivery," in *VLSI Circuits Digest of Technical Papers, 2002. Symposium on*, 2002, pp. 220-223.

[27] O. S. Unsal, J. W. Tschanz, K. Bowman, V. De, X. Vera, A. Gonzalez*, et al.*, "Impact of Parameter Variations on Circuits and Microarchitecture," *Micro, IEEE,* vol. 26, pp. 30-39, 2006.

[28] J. Tschanz, K. Nam Sung, S. Dighe, J. Howard, G. Ruhl, S. Vanga*, et al.*, "Adaptive Frequency and Biasing Techniques for Tolerance to Dynamic Temperature-Voltage Variations and Aging," in *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, 2007, pp. 292-604.

[29] T. J. Chaney and C. E. Molnar, "Anomalous Behavior of Synchronizer and Arbiter Circuits," *Computers, IEEE Transactions on,* vol. C-22, pp. 421-422, 1973.

[30] S. Mitra, N. Seifert, M. Zhang, Q. Shi, and K. S. Kim, "Robust system design with built-in soft-error resilience," *IEEE, Computer,* vol. 38, pp. 43-52, 2005.

[31] M. Nicolaidis, "Design techniques for soft-error mitigation," in *IC Design and Technology (ICICDT), 2010 IEEE International Conference on*, 2010, pp. 208-214.

[32] D. Ernst, K. Nam Sung, S. Das, S. Pant, R. Rao, P. Toan*, et al.*, "Razor: a low-power pipeline based on circuit-level timing speculation," in *Microarchitecture, 2003. MICRO-36. Proceedings. 36th Annual IEEE/ACM International Symposium on*, 2003, pp. 7-18.

[33] S. Das, D. Roberts, L. Seokwoo, S. Pant, D. Blaauw, T. Austin*, et al.*, "A self-tuning DVS processor using delay-error detection and correction," *Solid-State Circuits, IEEE Journal of,* vol. 41, pp. 792-804, 2006.

[34] K. A. Bowman, J. W. Tschanz, K. Nam Sung, J. C. Lee, C. B. Wilkerson, S. L. L. Lu*, et al.*, "Energy-efficient and metastability-immune timing-error detection and recovery circuits for dynamic variation tolerance," in *Integrated Circuit Design and Technology and Tutorial, 2008. ICICDT 2008. IEEE International Conference on*, 2008, pp. 155-158.

[35] K. A. Bowman, J. W. Tschanz, K. Nam Sung, J. C. Lee, C. B. Wilkerson, S. L. L. Lu*, et al.*, "Energy-Efficient and Metastability-Immune Resilient Circuits for Dynamic Variation Tolerance," *Solid-State Circuits, IEEE Journal of,* vol. 44, pp. 49-63, 2009.

[36] S. Das, C. Tokunaga, S. Pant, W. H. Ma, S. Kalaiselvan, K. Lai*, et al.*, "RazorII: In Situ Error Detection and Correction for PVT and SER Tolerance," *Solid-State Circuits, IEEE Journal of,* vol. 44, pp. 32-48, 2009.

[37] S. V. Devarapalli, P. Zarkesh-Ha, and S. C. Suddarth, "SEU-Hardened Dual Data Rate Flip-Flop Using C-Elements," in *Defect and Fault Tolerance in VLSI Systems (DFT), 2010 IEEE 25th International Symposium on*, 2010, pp. 167-171.

[38] T. Calin, M. Nicolaidis, and R. Velazco, "Upset hardened memory design for submicron CMOS technology," *Nuclear Science, IEEE Transactions on,* vol. 43, pp. 2874-2878, 1996.
[39] S. M. Jahinuzzaman, D. J. Rennie, and M. Sachdev, "A Soft Error Tolerant 10T SRAM Bit-Cell With Differential Read Capability," *Nuclear Science, IEEE Transactions on,* vol. 56, pp. 3768-3773, 2009.
[40] J. Zhou, D. Kinniment, G. Russell, and A. Yakovlev, "Adapting synchronizers to the effects of on chip variability," in *IEEE International Symposium on Asynchronous Circuits and Systems*, Piscataway, NJ 08855-1331, United States, 2008, pp. 39-47.
[41] A. J. Martin and M. Nystrom, "Asynchronous Techniques for System-on-Chip Design," *Proceedings of the IEEE,* vol. 94, pp. 1089-1120, 2006.
[42] C.-K. Tang, C.-Y. Lin, and Y.-C. Lu, "An Asynchronous Circuit Design with Fast Forwarding Technique at Advanced Technology Node," in *Quality Electronic Design, 2008. ISQED 2008. 9th International Symposium on*, 2008, pp. 769-773.
[43] A. J. Martin, "Can Asynchronous Techniques Help the SoC Designer?," in *Very Large Scale Integration, 2006 IFIP International Conference on*, 2006, pp. 7-11.
[44] M. Krstic, E. Grass, F. K. Gurkaynak, and P. Vivet, "Globally Asynchronous, Locally Synchronous Circuits: Overview and Outlook," *Design & Test of Computers, IEEE,* vol. 24, pp. 430-441, 2007.
[45] D. Sylvester, K. Agarwal, and S. Shah, "Variability in nanometer CMOS: Impact, analysis, and minimization," *Integration, the VLSI Journal,* vol. 41, pp. 319-339, 2008.
[46] J. Sparsø and S. Furber, *Principles of Asynchronous Circuit Design: A System Perspective*: Kluwer, 2001.
[47] P. A. Beerel, R. O. Ozdag, and M. Ferretti, *A Designer's Guide to Asynchronous VLSI*: Cambridge University Press, 2010.
[48] D. Marculescu and E. Talpes, "Energy awareness and uncertainty in microarchitecture-level design," *Micro, IEEE,* vol. 25, pp. 64-76, 2005.
[49] D. Ernst, S. Das, S. Lee, D. Blaauw, T. Austin, T. Mudge*, et al.*, "Razor: circuit-level correction of timing errors for low-power operation," *Micro, IEEE,* vol. 24, pp. 10-20, 2004.
[50] D. Kinniment, *Synchronization and arbitration in digital systems*. Hoboken, NJ: J. Wiley & Sons, 2007.
[51] D. J. Kinniment and J. V. Woods, "Synchronisation and arbitration circuits in digital systems," *Electrical Engineers, Proceedings of the Institution of,* vol. 123, pp. 961-966, 1976.
[52] M. Elgebaly and M. Sachdev, "Variation-Aware Adaptive Voltage Scaling System," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 15, pp. 560-571, 2007.
[53] T. Chen and S. Naffziger, "Comparison of adaptive body bias (ABB) and adaptive supply voltage (ASV) for improving delay and leakage under the presence of process variation," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 11, pp. 888-899, 2003.
[54] J. Tschanz, K. Bowman, S. Walstra, M. Agostinelli, T. Karnik, and V. De, "Tunable replica circuits and adaptive voltage-frequency techniques for dynamic voltage, temperature, and aging variation tolerance," in *VLSI Circuits, 2009 Symposium on*, 2009, pp. 112-113.
[55] D. Flynn, R. Aitken, A. Gibbons, and K. Shi, *Low Power Methodology Manual: For System-on-Chip Design*: Springer Publishing Company, Incorporated, 2010.
[56] *Virtuoso® Spectre® Circuit Simulator Reference*: Cadence Design Systems, 2008.
[57] *Virtuoso® Advanced Analysis Tools User Guide*: Cadence Design Systems, 2007.
[58] W. Ning, G. Fen, and W. Fei, "Design of a GALS Wrapper for Network on Chip," in *Computer Science and Information Engineering, 2009 WRI World Congress on*, 2009, pp. 592-595.

[59] A. Sheibanyrad, A. Greiner, and I. Miro-Panades, "Multisynchronous and Fully Asynchronous NoCs for GALS Architectures," *Design & Test of Computers, IEEE,* vol. 25, pp. 572-580, 2008.
[60] D. Ludovici, A. Strano, D. Bertozzi, L. Benini, and G. N. Gaydadjiev, "Comparing tightly and loosely coupled mesochronous synchronizers in a NoC switch architecture," in *Networks-on-Chip, 2009. NoCS 2009. 3rd ACM/IEEE International Symposium on*, 2009, pp. 244-249.
[61] T. Bjerregaard and S. Mahadevan, "A survey of research and practices of Network-on-chip," *ACM Comput. Surv.,* vol. 38, p. 1, 2006.
[62] J. Sparsø, "Asynchronous design of networks-on-chip," in *Norchip, 2007*, 2007, pp. 1-4.
[63] A. Janarthanan, V. Swaminathan, and K. A. Tomko, "MoCReS: an Area-Efficient Multi-Clock On-Chip Network for Reconfigurable Systems," in *VLSI, 2007. ISVLSI '07. IEEE Computer Society Annual Symposium on*, 2007, pp. 455-456.
[64] T. Bjerregaard and J. Sparso, "Implementation of guaranteed services in the MANGO clockless network-on-chip," *Computers and Digital Techniques, IEE Proceedings -,* vol. 153, pp. 217-229, 2006.
[65] T. J. Chaney, "Comments on "A Note on Synchronizer or Interlock Maloperation"," *Computers, IEEE Transactions on,* vol. C-28, pp. 802-804, 1979.
[66] T. Kacprzak, "Analysis of oscillatory metastable operation of an RS flip-flop," *Solid-State Circuits, IEEE Journal of,* vol. 23, pp. 260-266, 1988.
[67] K. Sung-Mo and L. Yusuf, *CMOS Digital Integrated Circuits Analysis and Design*, 3rd ed.: McGraw-Hill, Inc., 2003.
[68] T. Kacprzak and A. Albicki, "Analysis of mesastable operation in RS CMOS flip-flops," *Solid-State Circuits, IEEE Journal of,* vol. 22, pp. 57-64, 1987.
[69] L. S. Kim and R. W. Dutton, "Metastability of CMOS latch/flip-flop," *Solid-State Circuits, IEEE Journal of,* vol. 25, pp. 942-951, 1990.
[70] R. Ginosar, "Metastability and Synchronizers: A Tutorial," *Design & Test of Computers, IEEE,* vol. 28, pp. 23-35, 2011.
[71] S. Beer, R. Ginosar, M. Priel, R. Dobkin, and A. Kolodny, "The Devolution of Synchronizers," in *Asynchronous Circuits and Systems (ASYNC), 2010 IEEE Symposium on*, 2010, pp. 94-103.
[72] L. Kleeman and A. Cantoni, "Metastable Behavior in Digital Systems," *Design & Test of Computers, IEEE,* vol. 4, pp. 4-19, 1987.
[73] P. E. Allen and D. R. Holberg, *CMOS analog circuit design*, 2nd ed. New York,: Oxford University Press, 2002.
[74] C. Dike and E. Burton, "Miller and noise effects in a synchronizing flip-flop," *Solid-State Circuits, IEEE Journal of,* vol. 34, pp. 849-855, 1999.
[75] C. Foley, "Characterizing metastability," in *Advanced Research in Asynchronous Circuits and Systems, 1996. Proceedings., Second International Symposium on*, 1996, pp. 175-184.
[76] D. J. Kinniment, A. Bystrov, and A. V. Yakovlev, "Synchronization circuit performance," *Solid-State Circuits, IEEE Journal of,* vol. 37, pp. 202-209, 2002.
[77] D. Kinniment, K. Heron, and G. Russell, "Measuring deep metastability," in *Asynchronous Circuits and Systems, 2006. 12th IEEE International Symposium on*, Grenoble, France, 2006, pp. 2-11.
[78] D. J. Kinniment, C. E. Dike, K. Heron, G. Russell, and A. V. Yakovlev, "Measuring Deep Metastability and Its Effect on Synchronizer Performance," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 15, pp. 1028-1039, 2007.
[79] J. Zhou, D. J. Kinniment, C. E. Dike, G. Russell, and A. V. Yakovlev, "On-chip measurement of deep metastability in synchronizers," *IEEE Journal of Solid-State Circuits,* vol. 43, pp. 550-557, 2008.
[80] C. L. Portmann and H. Y. Meng, "Metastability in CMOS library elements in reduced supply and technology scaled applications," *Solid-State Circuits, IEEE Journal of,* vol. 30, pp. 39-46, 1995.

[81] M. S. Baghini and M. P. Desai, "Impact of technology scaling on metastability performance of CMOS synchronizing latches," in *Design Automation Conference, 2002. Proceedings of ASP-DAC 2002. 7th Asia and South Pacific and the 15th International Conference on VLSI Design. Proceedings.*, 2002, pp. 317-322.

[82] J. Zhou, D. Kinniment, G. Russell, and A. Yakovlev, "A robust synchronizer," in *Emerging VLSI Technologies and Architectures, 2006. IEEE Computer Society Annual Symposium on*, 2006, pp. 442-443.

[83] J. Zhou, M. Ashouei, D. Kinniment, J. Huisken, G. Russell, and A. Yakovlev, "Sub-threshold synchronizer," *Microelectronics Journal,* vol. 42, pp. 840-850, 2011.

[84] C. L. Portmann and T. H. Y. Meng, "Loading effects on metastable parameters of CMOS latches," in *VLSI Circuits, 1993. Digest of Technical Papers. 1993 Symposium on*, 1993, pp. 21-22.

[85] B. Rebaud, M. Belleville, C. Bernard, M. Robert, P. Maurine, and N. Azemard, "A comparative study of variability impact on static flip-flop timing characteristics," in *Integrated Circuit Design and Technology and Tutorial, 2008. ICICDT 2008. IEEE International Conference on*, Minatec Grenoble, France, 2008, pp. 167-170.

[86] J. Zhou, M. Ashouei, D. Kinniment, J. Huisken, and G. Russell, "Extending Synchronization from Super-Threshold to Sub-threshold Region," in *Asynchronous Circuits and Systems (ASYNC), 2010 IEEE Symposium on*, 2010, pp. 85-93.

[87] T. Polzer and A. Steininger, "C-element metastability mitigation using schmitt-triggers," Technische Universitat Wien, Institut fur Technische Informatik, Research Report 1/2012, 2012.

[88] R. L. Cline, "Method and circuit for improving metastable resolving time in low-power multi-state devices," U.S. Patent No. 5,789,945, 1998.

[89] M. Kayam, R. Ginosar, and C. E. Dike, "Symmetric Boost Synchronizer for Robust Low Voltage, Low Temperature Operation," EE Technical Report, Technion,2007.

[90] R. L. Freyman, "Latch circuit with reduced metastability," U.S. Patent No. 5,081,377, 1992.

[91] M. Priel, S. B. Gingold, and D. Kuzmin, "Device having synchronizing capabilities," U.S. Patent No. 7,977,983, 2011.

[92] V. Stojanovic and V. G. Oklobdzija, "Comparative analysis of master-slave latches and flip-flops for high-performance and low-power systems," *Solid-State Circuits, IEEE Journal of,* vol. 34, pp. 536-548, 1999.

[93] N. Nedovic and V. G. Oklobdzija, "Dual-edge triggered storage elements and clocking strategy for low-power systems," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 13, pp. 577-590, 2005.

[94] J. Tschanz, S. Narendra, C. Zhanping, S. Borkar, M. Sachdev, and D. Vivek, "Comparative delay and energy of single edge-triggered and dual edge-triggered pulsed flip-flops for high-performance microprocessors," in *Low Power Electronics and Design, International Symposium on, 2001.*, 2001, pp. 147-152.

[95] M. B. Josephs and J. T. Yantchev, "CMOS design of the tree arbiter element," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 4, pp. 472-476, 1996.

[96] C. H. van Berkel and C. E. Molnar, "Beware the three-way arbiter," *Solid-State Circuits, IEEE Journal of,* vol. 34, pp. 840-848, 1999.

[97] A. C. Davies, "Dynamic properties of a multiway arbiter," in *Circuits and Systems, 2000. Proceedings. ISCAS 2000 Geneva. The 2000 IEEE International Symposium on*, 2000, pp. 221-224 vol.3.

[98] O. Maevsky, D. J. Kinniment, A. Yakovlev, and A. Bystrov, "Analysis of the oscillation problem in tri-flops," in *Circuits and Systems, 2002. ISCAS 2002. IEEE International Symposium on*, 2002, pp. I-381-I-384 vol.1.

[99] A. Bystrov and A. Yakovlev, "Ordered arbiters," *Electronics Letters,* vol. 35, pp. 877-879, 1999.

[100] A. J. Martin, "Collected papers on asynchronous VLSI design," Technical Report, Caltech-CS-TR-90-09, Dept. of Computer Science, California Institute of technology, 1990.

[101] L. Yu, G. Xuguang, Y. Yang, and Y. Yintang, "An asynchronous low latency ordered arbiter for network on chips," in *Natural Computation (ICNC), 2010 Sixth International Conference on*, 2010, pp. 962-966.
[102] A. Bystrov, D. J. Kinniment, and A. Yakovlev, "Priority arbiters," in *Advanced Research in Asynchronous Circuits and Systems, 2000. (ASYNC 2000) Proceedings. Sixth International Symposium on*, 2000, pp. 128-137.
[103] W. J. Dally and J. W. Poulton, *Digital systems engineering*. Cambridge, U.K.: Cambridge University Press, 1998.
[104] P. Teehan, M. Greenstreet, and G. Lemieux, "A Survey and Taxonomy of GALS Design Styles," *Design & Test of Computers, IEEE,* vol. 24, pp. 418-428, 2007.
[105] E. L. Sueur and G. Heiser, "Dynamic voltage and frequency scaling: the laws of diminishing returns," presented at the Proceedings of the 2010 international conference on Power aware computing and systems, Vancouver, BC, Canada, 2010.
[106] F. Ishihara, F. Sheikh, and B. Nikolic, "Level conversion for dual-supply systems," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 12, pp. 185-195, 2004.
[107] S. H. Kulkarni and D. Sylvester, "High performance level conversion for dual VDD design," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 12, pp. 926-936, 2004.
[108] R. Puri, L. Stok, J. Cohn, D. Kung, D. Pan, D. Sylvester*, et al.*, "Pushing ASIC performance in a power envelope," in *Design Automation Conference, 2003. Proceedings*, 2003, pp. 788-793.
[109] Q. A. Khan, S. K. Wadhwa, and K. Misri, "A single supply level shifter for multi-voltage systems," in *VLSI Design, 2006. Held jointly with 5th International Conference on Embedded Systems and Design., 19th International Conference on*, 2006, pp. 557-560.
[110] R. Garg, G. Mallarapu, and S. P. Khatri, "A Single-supply True Voltage Level Shifter," in *Design, Automation and Test in Europe, 2008. DATE '08*, 2008, pp. 979-984.
[111] R. H. Liu, "Multi-voltage Adaptive Voltage Scaling SoC Reference Design," *National Semiconductor Corporation,* pp. 1-8, 2006.
[112] S. Krishnamoorthy, "Method for determining static flip-flop setup and hold times," U.S. Patent No. 6,311,148, 2001.
[113] S. A. Sadrossadat, M. Mirsaeedi, K. Ponnambalam, and M. Anis, "Framework for statistical design of a flip-flop," in *Electronics, Circuits, and Systems, 2009. ICECS 2009. 16th IEEE International Conference on*, 2009, pp. 679-682.
[114] I. W. Jones, Y. Suwen, and M. Greenstreet, "Synchronizer Behavior and Analysis," in *Asynchronous Circuits and Systems, 2009. ASYNC '09. 15th IEEE Symposium on*, 2009, pp. 117-126.
[115] D. Li, P. Chuang, and M. Sachdev, "Comparative analysis and study of metastability on high-performance flip-flops," in *Quality Electronic Design (ISQED), 2010 11th International Symposium on*, 2010, pp. 853-860.
[116] A. Wang, B. H. Calhoun, and A. P. Chandrakasan, *Sub-threshold design for ultra low-power systems*: Springer, 2006.
[117] I. E. Sutherland, "Micropipelines," *Commun. ACM,* vol. 32, pp. 720-738, 1989.
[118] G. Fuchs, M. Fugger, and A. Steininger, "On the Threat of Metastability in an Asynchronous Fault-Tolerant Clock Generation Scheme," in *Asynchronous Circuits and Systems, 2009. ASYNC '09. 15th IEEE Symposium on*, 2009, pp. 127-136.
[119] M. Alshaikh, D. Kinniment, and A. Yakovlev, "A synchronizer design based on wagging," in *Microelectronics (ICM), 2010 International Conference on*, 2010, pp. 415-418.
[120] J. Jex and C. Dike, "A fast resolving BiNMOS synchronizer for parallel processor interconnect," *Solid-State Circuits, IEEE Journal of,* vol. 30, pp. 133-139, 1995.
[121] J. Cortadella, M. Kishinevsky, A. Kondratyev, L. Lavagno, and A. Yakovlev, *Logic Synthesis of Asynchronous Controllers and Interfaces* vol. 8: Springer, 2002.

[122] H. Mahmoodi-Meimand and K. Roy, "Dual-edge triggered level converting flip-flops," in *Circuits and Systems, 2004. ISCAS '04. Proceedings of the 2004 International Symposium on*, 2004, pp. II-661-4 Vol.2.
[123] R. Bai and D. Sylvester, "Analysis and design of level-converting flip-flops for dual-Vdd/Vt integrated circuits," in *System-on-Chip, 2003. Proceedings. International Symposium on*, 2003, pp. 151-154.
[124] A. Chattopadhyay and Z. Zilic, "GALDS: a complete framework for designing multiclock ASICs and SoCs," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on,* vol. 13, pp. 641-654, 2005.