Journal of Electrical and Computer Engineering Innovations
(JECEI)
Journal homepage: http://www.jecei.sru.ac.ir

Research paper

Adaptive Energy-Efficient Variation-Aware Dynamic Frequency Management

H. Dorosti*

Department of Computer Systems Architecture, Faculty of Computer Engineering, Shahid Rajaee Teacher Training University, Tehran, Iran.

Article Info

Article History:

Received 18 September 2021
Reviewed 27 October 2021
Revised 12 January 2022
Accepted 13 January 2022

Keywords:

Internet of things
Timing slack monitoring
Negative slack measurement
Clock stretching
Frequency scaling
Ultra-Low-Energy

*Corresponding Author's Email Address: *hdorosti@sru.ac.ir*

Abstract

Background and Objectives: Considering the fast growing low-power internet of things, the power/energy and performance constraints have become more challenging in design and operation time. Static and dynamic variations make the situation worse in terms of reliability, performance, and energy consumption. In this work, a novel slack measurement circuit is proposed to have precise frequency management based on timing violation measurement.

Methods: the Proposed slack measurement circuit is based on measuring the delay difference between the edge clock pulse and possible transition on path end-points (primary outputs of design). The output of the proposed slack monitoring circuits is a digital code related to the current state of target critical path delay. In order to convert this digital code to equivalent delay difference, the delay of a reference gate is mandatory which is the basic unit in the proposed monitor. This monitor enables the design to have more precise and efficient frequency management, while maintaining the correct functionality regarding low-power mode.

Results: Applying this method on a MIPS processor reduces the amount of performance penalty and recovery energy overhead up to 30% with only 2% additional hardware. Results for benchmark applications in low-power mode, show 7-30% power improvement in normal execution mode. If the application is resilient against occurred errors duo to timing violations, the proposed method achieves 20-60% power reduction considering approximate computation as long as application is showing resilience. The performance of the proposed method depends on the degree of application resilience against the timing errors. In order to keep generality of the proposed monitor for different applications, the resilience threshold is user programmable to configure according to the requirements of each application.

Conclusion: The results show that precise frequency scheduling is more energy/power efficient in static and dynamic variation management. Utilizing a proper monitor capable of measuring the amount of violation will help to have finer frequency management. On the other hand, this method will help to use the resilience of application according to estimation about the possible error value based on measured violation amount.

Introduction

Internet of Things (IoT) is a fast emerging technology which enables continuous sensing data flow and actuation controls through everything and involves different applications from health, industry, automation, military and etc. The demands for these applications are different in terms of performance, power/energy, reliability, and lifetime.

Energy efficiency is the common objective for applications ranging from energy-constrained with low performance and high lifetime requirements to high-performance maintainable needs.

Ultra-low-energy processors providing performance

demands for IoT applications must be kept at a reasonable lifetime for network operation.

Feature size scaling makes the design of those processors more cost effective, but in contrast, due to leakage current and process variation, energy efficiency is dropped down and reliability issues are imposed. On the other hand, wear-out mechanisms will shorten the lifetime of the network and impose energy overhead. Clearly, any variation related to design timing, due to Process-Voltage- Temperature variations (PVT) & Aging, will result in reliability and performance issues. Timing-error if occurs, will impose energy overhead during the operation of the processor. Therefore, timing errors in design time as well as frequency management during the operation of the design should be considered.

The rest of this paper is organized as follows: In the next section, related works are explored and their achievements are presented. After literature review, we discuss about the proposed method and design details as another section. The experimental results and analysis of the proposed method will come afterward. Finally we conclude the paper in the last section.

Related Works

In order to address reliability issues due to PVT & Aging variations, timing error detection is a need in recent technologies. To date studies have mostly focused on timing error detection or slack measurement to prevent the design functionality failure instead of inserting longer guard-bands (20% margins in [1]) between the path delays and clock period.

There are a class of timing error detection methods namely called In-situ timing error detection consist of RAZOR [2], [3], RAZOR Lite [4]. There exists also another method called Replica circuits [5] which are configured with the latency of the design paths to forecast the error probability due to temporal and spatial dependency of circuit elements. Measurements of these methods are used to capture the statistical state of the timing variations inside the design and to manage the voltage and frequency or even to utilize the error correction [7] and recovery [2], [3], [7] methods in order to compensate the variation effects and to retrieve the processor against failure state.

Another class of timing error forecasting is utilizing positive slack monitoring circuits in conjunction with timing guard-band. The authors of [8] have proposed slack monitoring circuit at end-points of the design to measure the available guard-band between the path delays and the period of the clock cycle. The measurement results are available through the scan chain and the slack monitors are inserted using ATPG toolset. In case the design timing is altered, the signal cannot cross the whole elements of the delay line and the embedded register will capture the state of the delay margin at rising edge of the clock cycle. When timings margin is reached to critical state, the processor is configured to increase the margin and to ensure the correct functionality.

The authors of [9] used activation probability and correlated detection window to maximize the efficiency of measurements of positive slack monitors. Using this method, slack monitors are placed in proper nodes while more precise measurements are achieved.

A. Benhassain et al. have used a voltage regulation feedback based on positive slack monitor to perform a better voltage selection according to variation in path delays or circuit operation modes. Using this method, the proper voltage and frequency selection is simpler and sign-off difficulties for different operating conditions are reduced [10].

SlackProbe in [11] is a method to increase the efficiency of positive slack monitor insertion as well as to increase the observability and as a result, to reduce the number of required slack monitors. In this work, the monitors are inserted at intermediate nodes instead of the end-points for early detection and smaller area overhead. This method could be used jointly with different compensation techniques.

The authors of [12] have proposed a novel slack monitor circuit in transistor level to improve its accuracy, power, area, and aging resistance. The designed monitor circuit is placed in master-slave flip-flop to add the monitor capability into the registers. These registers are used with a high-resolution TDC circuit to measure the remaining positive slack of the selected paths. TDC [13] circuit is similar to ring oscillator [14] design based on basic gate elements which is configured to measure the path delay with higher timing resolution.

Timing speculation [15] is another proposed method to prevent the effects of timing violation within the same clock cycle. In this work and similar publications [16] the monitors are placed in mid-point of candidate path and detect the failure event before the end of clock period. Global clock stretching is a short time method to reduce the power and performance penalty and could not provide the lifetime efficiency. Authors of [17] used machine learning methods to estimate current aging state and remaining lifetime by 97% average precision. This work achieves an acceptable estimation, but not sufficient for precise real-time frequency scaling.

There are another class of works such as [18] which try to improve design lifetime by utilizing aggressive voltage and frequency scaling without any precise measurements. These methods rely on planning voltage and frequency of the design to reduce the stress and improve the lifetime by 40%.

A major group of recent works have been exploited positive slack monitoring for timing margin

measurement to scale the working frequency in order to prevent timing error occurrence. These methods act precautionary to ensure correct operation by placement of in-situ timing circuits in mid or end points of candidate paths. Another group of studies utilize the error detection, state retrieval, and re-compute with new configuration. These methods are known as recovery methods. They provide however limited power/energy and lifetime improvements and have lower flexibility considering different application requirements due to lack of the exact timing behavior consideration. In order to have energy efficient variation management, both of likelihood and severity of timing violation should be considered, while recent studies only consider threshold-based monitoring of delay growth (as severity of possible violation). Precise monitoring of delay growth by tracking the occurrence of timing violation over a period of time will result in more energy efficient voltage and frequency scaling method.

In this work, we propose another intermediate method to monitor the amount of negative slack at the end-points. This method reduces the performance and energy overhead significantly and provides more precise frequency management capability. The key contributions to this work include:

1) A novel negative slack monitoring circuit capable of measuring the amount of timing violations up to ½ clock period.

2) A novel monitor insertion method based on timing analysis of the design.

3) A novel frequency management scheme with the introduction of the forecasting pipe stage based on the proposed slack monitor. This method provides more efficient voltage and frequency management in terms of performance and energy consumption while preventing functional failure of the design.

Negative Slack Monitoring

In this section, we present a new architecture for negative slack monitor and explain its functionality and design in details. Careful timing considerations are required which will be discussed in this section. The proposed slack monitor insertion method is used based on the statistical static timing analysis of the design. The architecture level utilization of the monitors together with the added benefits based on the timing analysis of the design will be presented as follows.

Negative Slack Monitor Architecture

In order to measure the positive slack, designers [8]-[12] used a circuit which generates a pulse according to difference between the input and output of flip-flops which are placed at the circuit end-points. The length of this pulse is measured using another circuit known as time-to-digital converter (TDC) and the measurement result is captured in rising edge of the clock signal. The story behind the negative slack monitor is quite different with the one with positive slack, in which the negative slack measurement starts at the rising edge of the clock signal and continues until a transition detector detects the difference between the input and output of the end-point flip-flop. Fig. 1 shows the architecture of the negative slack monitor. As shown in this figure, the clock signal is fed to the TDC circuit and the generated signal from the transition detector is used to capture the measurement results. While in positive slack monitor, the output of the design path crosses the TDC circuit and the clock signal captures the measurement result as available timing margin.

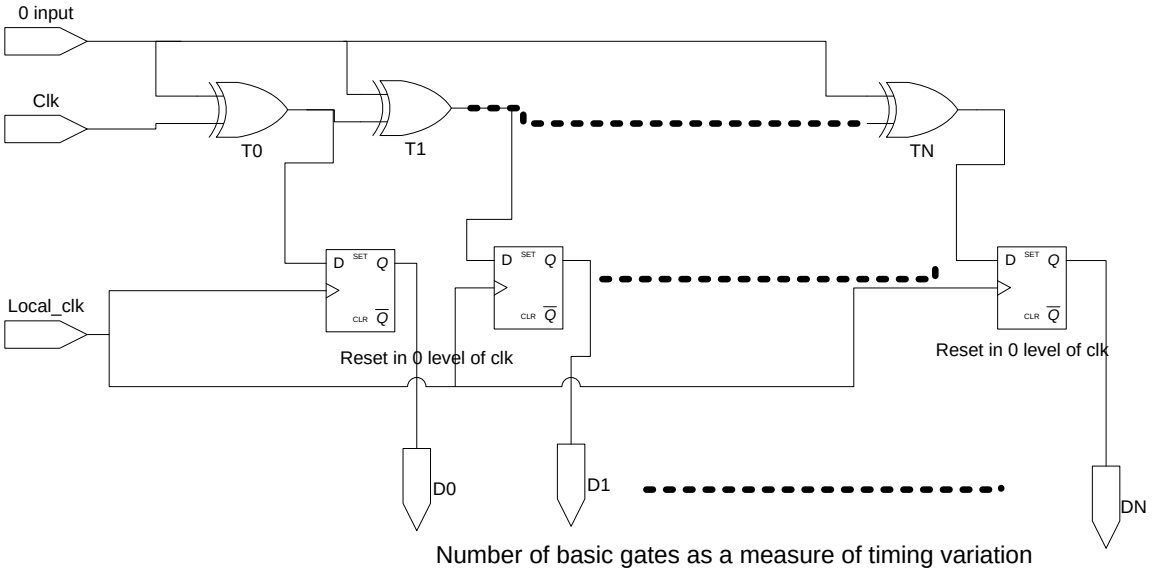

Fig. 1: The architecture of negative slack monitoring sensor.

Fig. 2 shows the structure of transition detector which is used jointly with the circuit above. The signal naming in both figures are the same to follow the dependency between these figures.

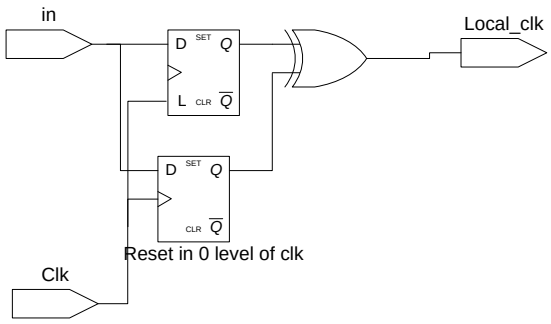

Fig. 2: The architecture of transition detector which is used with slack monitor circuit.

In order to use this monitor in the design, careful timing considerations are required. Minimum delay of the paths ending to the transition detector should be more than half of the clock period. The mentioned constraint means that the negative slack should be less than half of the clock period and the TDC should have capability to measure the latency with higher precision. If the delay of activated path ending to the slack monitor has no timing violation, there is no transition and meanwhile the register captures zero value meaning no negative slack occurred. If the signal violates the safety margin or even crosses the rising edge of the clock cycle, the transition detector will activate the error signal (Local_clk) as a result. At the rising edge of the error signal, the state of the clock signal crossing the delay line is captured in the existing flip-flops (Fig. 1). The captured digital data is a code that represents the amount of negative slack for the expected end-point, and will count the amount of '1's in this code to return the negative slack in unit delay. In this structure the unit delay means the delay of a basic XOR gate which shapes a buffer with a specific delay. In this scheme, the transition detector has a delay that is equal to two-unit delay measurement and should be considered in the final result.

It is important to know that the existence of a latch in transition detector is mandatory and the propagation delay of that latch should be greater than or equal to that of the flip-flop. Removing the latch from the design creates instability in measurements due to detector delay discrepancy. There is a hidden exception in the operation of the proposed monitor circuit, when the amount of negative slack is smaller than the propagation delay of the main flip-flop. In this case, the flip-flop delay will be counted on the measured negative slack, while it is not counted for greater negative slacks. Here, consider a condition in which the value of the end-point remains the same for a set of two clock cycles. After the rising edge of the second pulse, the transition detector activates the error signal due to the late signal arrival. In this situation when the logic state of the arriving signal remains the same in comparison with the previous clock edge, the measured value for the same negative slack will vary time to time and the delay of the main flip-flop in transition detector will not be counted on delay measurement accordingly. This event can therefore affect the accuracy of the sensor and should be compensated in order to resolve the problem; we need to reset the transition detector flip-flop during zero level of the clock signal to remove the consecutive '1's. Fig. 3 presents the proposed flip-flop structure using two latches to enable the flip-flop reset at zero level of the clock signal. The two consecutive zero values problem is resolved using additional flip-flop that will capture the comparison between the two consecutive clocked signals. When the output of this flip-flop is activated and the measured delay is less than or equal to the three unit delays, then the delay of the flip-flop should be taken into account for measurement correction. In order to reduce the compensation area and power overhead, a latch is placed between the arriving signal and comparator to make the problem to be uniformed for all instability conditions.

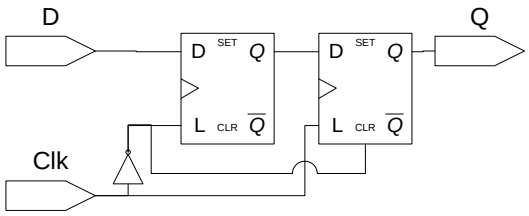

Fig. 3: The flip-flop structure enabling reset on clock zero level for negative slack monitor.

Our proposed slack monitoring circuit is capable of detecting the delayed transition from the beginning of safety margin to the next half clock period.

A. Slack Monitor Insertion

To have an efficient monitor insertion, statistical timing analysis is required. Timing violation and safety margin impose that the most vulnerable endpoints are the best candidates to insert the proposed monitor. Since any timing variation, due to a path entering the safety margin, may cause that path to violate the timing, the ending point to that path can be a candidate to place the monitor. The insertion method for a standard synthesis flow, according to Fig. 4, could be extended to every netlist generated from physical design and layout by applying the timing analysis to the netlist. Statistical static timing analysis removes unnecessary timing margins and reduces the amount of critical nodes which

results in smaller area and energy overhead due to slack monitors. The delay distribution of the paths is extracted from the timing analysis. Critical end points and critical paths will be identified with regards to timing margin requirement and different variation sources (such as 10% of clock period). These nodes are identified as critical nodes in which the slack monitors are inserted to measure the amount of timing violation.

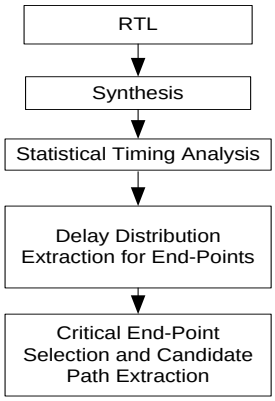

Fig. 4: The monitor insertion flow diagram.

First of all, we need to analyze the path delay distribution for the design to evaluate the nature of the delay distribution with respect to the clock period. Considering Tacc as an acceptable path delay,

Tacc = Clock Period – Safety Margin (1)

It is obvious that major parts of the design paths are shorter than the Tacc. These paths do not cause the safety margin to enter the critical state except some minor parts (known as critical path) that could be the main source of timing violation but are rarely activated. Figs. 5, 6, and 7 are shown the delay distribution of the paths in combination with the activity probability for three ISCAS'85 benchmarks which are achieved applying random test vectors and confirm the above sentence. The delay distribution combination of the design paths with activity probability is used to determine the candidate end-points.

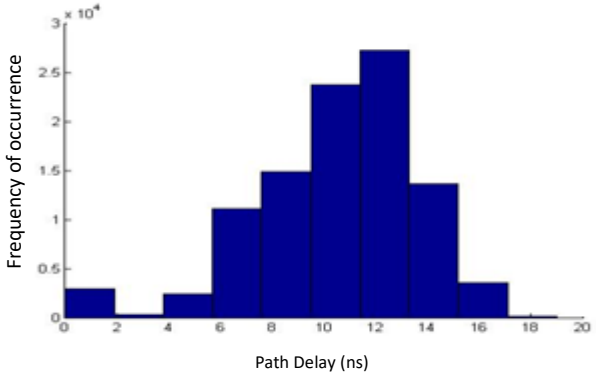

Fig. 5: The delay distribution (histogram) of paths for C3540 ISCAS benchmark.

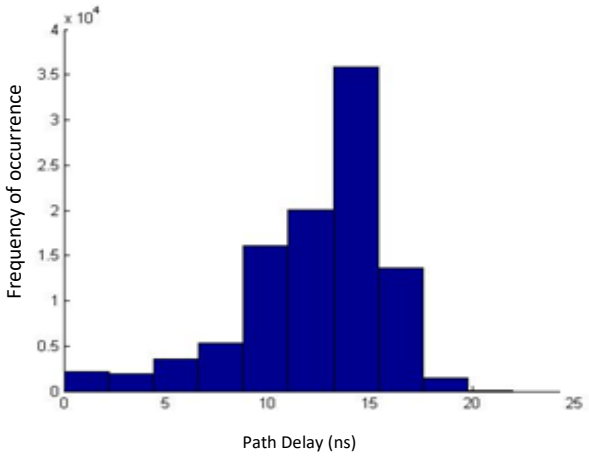

Fig. 6: The delay distribution (histogram) of paths for C6288 ISCAS benchmark.

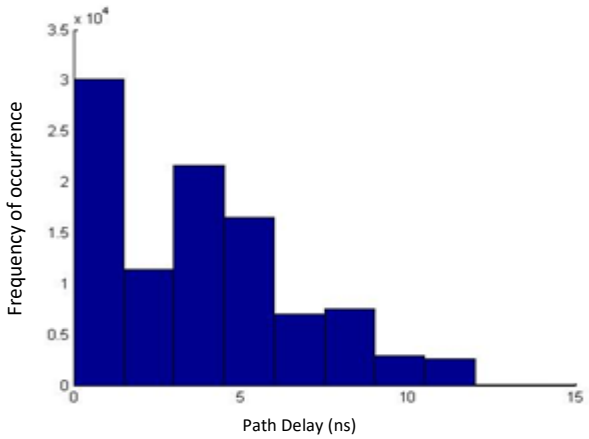

Fig. 7: The delay distribution (histogram) of paths for C7552 ISCAS benchmark.

These candidates have the capability to grow and violate the timing constraints which result in timing errors and possible functional failures. We have used Tacc in our previously published work [21] as a measure of criticality to prevent the failure due to process variations.

The usage of this parameter for every static and dynamic variation which modifies the timing specifications is proposed accordingly.

The candidate paths are then extracted for the design, according to delay distributions of the paths terminating each end-point, and the maximum variability due to different variation sources and performance requirements (clock frequency).

Any point, at the end of those paths violating the safety margin of the clock cycle, remains candidate for timing error and the others are considered to be error-prone end-points.

Fig. 8 presents the candidate end-points for the benchmark circuits with 10% safety margin assigned (Red Line).

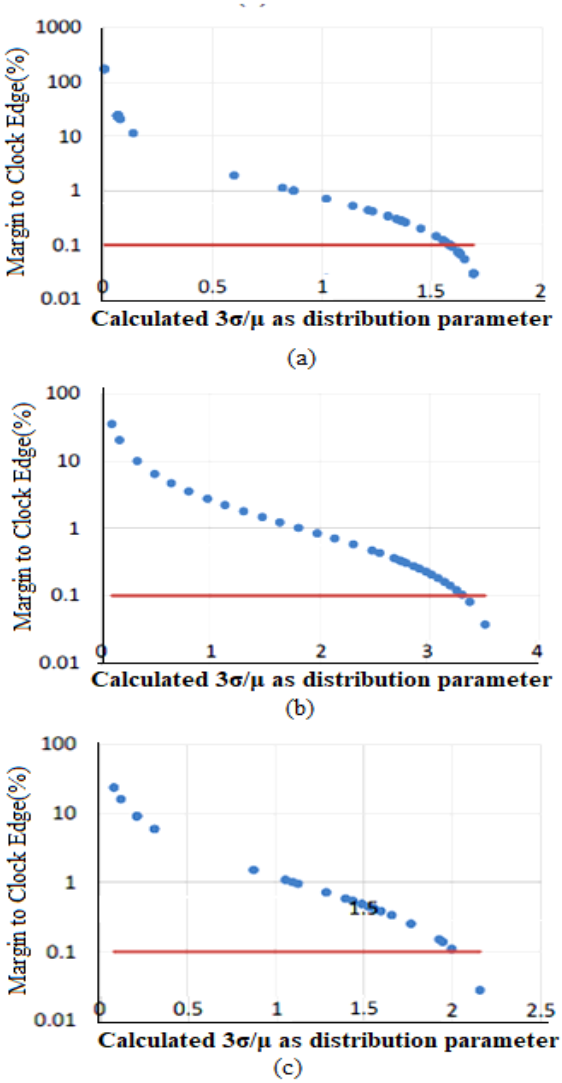

Fig. 8: Number of candidate end-points for a) c3540, b) c6288, and c) c7552 ISCAS benchmarks. Horizontal axis is calculated using the average and standard deviation of path delays for each benchmark.

Utilization Method

After the end-point selection, it is time to measure slacks to be used in dynamic management method and to compensate the variation effects.

As mentioned in the second section, recent studies use the slack monitor to measure the positive slack and to keep the safety guard-band in order to ensure the correct functionality. When the safety guard-band could not be established, the use of dynamic frequency scaling is inevitable as a result. Considering the lower probability of timing error at the end-points and the severity of delay growth with respect to different variation sources, there is no need to scale the frequency (and probably the voltage) of the design. Frequency scaling burdens more performance and energy overhead while clock stretching will resolve the problem with smaller overhead. In order to clarify the situation, one could consider a design working with 2GHz frequency and 10% activation probability for timing errors.

According to recent studies, anytime the path delays enter the safety margin, the clock frequency will be scaled to 1.5GHz. Reconfiguration reduces the performance of the design by 25% which makes the design to consume almost 25% more energy in order to finish the same task. The more the frequency steps are further away, the more energy overhead will be imposed to the circuit.

While, clock stretching to prevent timing error, will imposes only 10% performance and energy overhead. In order to reduce the supply voltage to achieve higher energy efficient design, this method makes the design to perform more reliable functionality and to achieve a reasonable performance.

Fig. 9 shows the timing diagram of the negative slack monitor and the clock stretch mechanism.

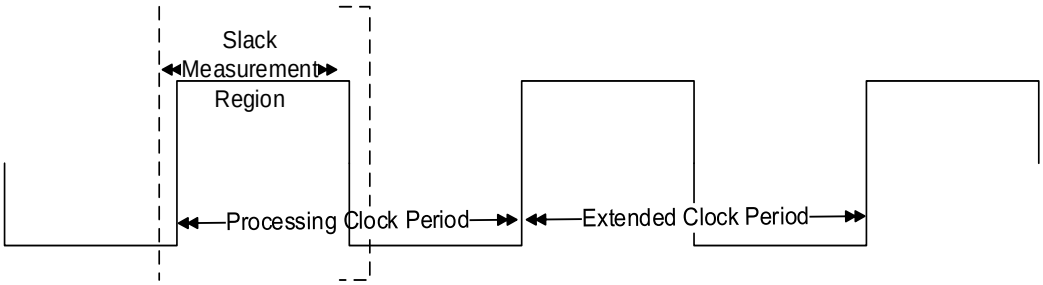

Fig. 9: The timing diagram for negative slack monitoring and clock stretching.

Since the continuous adjustment of the clock frequency is not possible in reality and the designers use discrete frequency steps to manage working frequency, there is a cross point between the model of performance overhead for the proposed stretching method and the methods presented in literature review (2) shows the mathematical description for this cross point in which the left hand side is for the proposed method and the right hand side is for the frequency scaling.

$$(N_{clk} + p \times N_{clk}) \times T_{clk} = N_{clk} \times (1+k) \times T_{clk} \quad (2)$$

where N_{clk} is the number of cycles required to execute the application, p is the activation probability of timing error, k is the steps of frequency scaling and T_{clk} is period of baseline clock signal.

According to (2), the proposed method is demonstrated to have higher performance and lower energy (neglecting the energy consumption for additional hardware for complex design) consumption compared to the others for k > p and vice versa. The imposed overhead is the same for k = p. Therefore, the proposed method will act as a finer frequency management method in comparison with the recent studies when k > p.

It is important to know that, when the paths exceed beyond the two consecutive cycles, this method cannot prevent timing error and frequency scaling is therefore mandatory. Then, we have to measure the amount of the negative slack rather than relying only to the timing error detection. Using the negative slack monitor, any latent transition is detected and measured to be considered in stretching and scaling of the clock cycle. In order to have efficient frequency management method, we need to introduce replica pipe stage as the forecasting stage. Candidate end-points and critical paths ending to these points are copied to a prior pipeline stage which is known as replica stage, and the same input vector is applied to both stages. When the monitor asserts the timing error prediction signal, due to similarity in both of replica and original end-points and identical input vector, it is obvious to occur in original stage. Then, the next clock is stretched or scaled to extend the design lifetime. Fig. 10 presents the position of both replica (gray blocks) and original stages (white blocks).

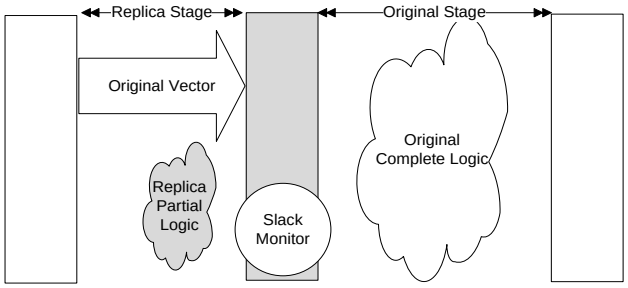

Fig. 10: The schematic of replica pipe stage which is used to measure negative slack monitoring in dynamic frequency management.

This method gives another chance to use approximate computing without any additional overhead in terms of area, performance and energy consumption. The amount of computation error has a direct relation to the severity of timing variation (or the amount of measured negative slack). According to the application and its resiliency against the computation errors, we have measured the tolerable error threshold and mapped the threshold to the slack measurements. When the negative slack is shorter than the tolerable value, the timing error is ignored and masked by the application and there is no need to further compensation in architecture level. This is an outstanding feature of the proposed method for multimedia applications which have a degree of resilience against the errors.

Results and Discussion

The negative slack monitor using the mentioned insertion and utilization method is applied to ISCAS benchmarks and MIPS processor. The benchmark circuits are synthesized in 45nm technology using standard synthesis tools, and the energy consumption and performance of the design are achieved using power compiler and timing analyzer tools. The variation effects on path delays are modeled using proper models for different sources in gate level simulation utilizing PLI interface. The delay distribution of the basic gates is used to model the effect of process variation. The effects of voltage and temperature variations are modeled as random fluctuation in candidate basic elements. Wear-out effects are modeled as progressive timing growth with respect to the effective parameters such as voltage, frequency, activity factor, operating time, and etc.

Table 1 presents the specifications of the proposed slack monitor. According to the results, the monitor has negligible area and energy overhead and it measures the negative slack with higher accuracy.

Table 1: Specifications of the proposed slack monitor in 45nm

Area (nm^2)	Power (uW)
19	25

Appling the monitor to the candidate ISCAS benchmarks shows that the proposed method provides higher performance and lower energy consumption compare to pure guard-banding and frequency scaling methods. The proposed method is compared to a baseline end-point monitor insertion method [8] as baseline method. Table 2 presents the performance and energy consumption of both methods on candidate benchmark circuits. The last two columns of this table present the performance overhead in comparison with the state with no timing variation.

Table 2: Comparison of ISCAS benchmarks using the baseline and the proposed slack monitors in terms of energy and performance in 45nm technology

Benchmark	Power (mW)		Performance Reduction	
	Proposed	[8]	Proposed	[8]
C3540	0.65	0.8	5%	30%
C6288	1.4	1.7	6%	30%
C7552	1.15	1.3	10%	30%

Considering the execution unit of the MIPS processor

to be vulnerable to the timing errors, the candidate end-points for this stage is extracted and the replica stage is added to the architecture. Table 3 presents the area and energy overhead in comparison to the baseline method. In most cases, the proposed method results in better performance and lower energy consumption. In some cases, both methods have the same efficiency due to severity of the delay growth in design paths.

Table 3: Efficiency of the baseline and the proposed slack monitor and imposed overhead on MIPS processor in 45nm

Error Rate	Power (mW)		Performance Reduction	
	Proposed	[8]	Proposed	[8]
10%	3.32	3.91	10%	30%
30%	3.95	3.94	30%	30%
50%	3.95	3.96	30%	30%
80%	3.95	3.96	30%	30%

Using the capability of the proposed method to reduce the voltage deliberately to achieve lower power/energy consumption while keeping the application functionality, 7-30% power improvement is achieved (Table 4). In this case, we have used lower supply voltages which results in timing violation to improve power/energy consumption, and the proposed method tolerates the effect of supply voltage reduction by stretching the clock period to keep the correct functionality.

Table 4: Power improvement of the processor when the studied benchmarks have been run under different operating voltage level without performance overhead

Benchmark	V_{DD} (V)	Power (mW)	Improvement
adpcm	0.9	2.8	25%
	1.1	3.7	
blowfish	0.9	3.2	30%
	1.1	4.6	
Dfmul	0.9	2.2	19%
	1.1	2.8	
Gsm	0.9	2.6	25%
	1.1	3.5	
Gaussian blur	0.9	1.6	8%
	1.1	1.8	
JPEG	0.9	2.4	12%
	1.1	2.7	
Sobel	0.9	1.7	7%
	1.1	1.8	

Also, considering inherent error resiliency of the benchmark applications, remarkable results are achieved. Table 5 shows the power/voltage improvements for both the normal and low-power mode. Fig. 11 shows an execution result for JPEG benchmark application in both operating modes. The acceptable condition for error resilience is considered as MSSIM [22] to be greater than 0.9 for all benchmark applications and the negative slack threshold is then determined according to this parameter. Enabling low power mode in comparison to the sense of error resilience capability in the proposed method results in 20-60% power improvement without significant performance overhead.

Table 5: Power improvement of the processor when the studied benchmarks have been run under different operating voltage level achieving MSSIM [20] greater than 0.9 without performance overhead

Benchmark	V_{DD} (V)	Power (mW)	Improvement
adpcm	0.9	1.7	54%
	1.1	3.7	
blowfish	0.9	2	57%
	1.1	4.6	
dfmul	0.9	1.7	39%
	1.1	2.8	
gsm	0.9	1.6	54%
	1.1	3.5	
Gaussian blur	0.9	1.4	22%
	1.1	1.8	
JPEG	0.9	2	26%
	1.1	2.7	
Sobel	0.9	1.4	22%
	1.1	1.8	

(a) (b)

Fig. 11: JPEG benchmark output, (a) original image, (b) degraded image of voltage 0.9.

In order to have better comparison between performance of the proposed work and the state of the art literature, we have provided comparison data with two recent works in terms of area and power overhead. Table 6 summarizes the comparison data. The reported

area and power overhead is for all required extra hardware and power consumption after applying the method compared to initial design. Power overhead is a total value extracted from individual evaluation process of each work and reported as an average of experiments. In this table, it is clear that the proposed work operates with only 2% extra hardware (smallest area overhead) and consumes 3% extra power.

Table 6: Efficiency of the proposed monitor compared to literature

Overhead	Proposed	[8]	[11]	[15]
Area overhead (%)	2	5	9	3
Power overhead (%)	3	10	16	3

Conclusion

Considering the growth rate of low-power internet of things, the power/energy and performance constraints for these applications are so tight. Static and timing variations make the situation worse in terms of reliability, performance and power/energy consumption. In this work, a new slack monitoring circuit and monitor insertion method is proposed. A novel frequency management scheme is introduced based on the proposed slack monitor at the candidate end-points. We have used negative slack monitor instead of positive one to enable more precise frequency management by measuring the actual delay growth of design paths. In order to have in situ error correction by clock stretching, the measurement on the selected end-points was performed one cycle before. A *replica pipe stage* is required to measure the probability and severity of timing error to decide the clock cycle stretch or to scale the frequency. This method provided another low-power mode which enabled the design to execute the applications in lower supply voltages with correct functionality with 7-30% power improvement. Also, in this method there is another option to sense the resilience of application against the timing error which reduces the amount of performance and energy overhead (20-60% power improvement) to high extent.

Author Contributions

H. Dorosti designed the experiments and collected the data through proper simulations, and data analysis is carried out by him. Finally H. Dorosti interpreted the results and wrote the manuscript.

Acknowledgment

The author would like to acknowledge the scientific support of Professor Sied Mehdi Fakhraie whose experience in the field and insightful guidelines was very helpful for the project. The authors would like also to thank Dr. Fakhraie for providing the computing equipments at the Silicon Intelligence laboratory of the University of Tehran.

Conflict of Interest

Author declare that there is no conflict of interests regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancy have been completely observed by the authors.

Abbreviations

IoT	Internet of Things
PVT	Process, Voltage and Temperature
TDC	Time to Digital Converter
ATPG	Automatic Test Pattern Generation

References

[1] W. Abadeer, W. Ellis, "Behavior of NBTI under AC dynamic circuit conditions," presented at the 41'st International Reliability Physics Symposium, Dallas, TX, USA, 2003.

[2] D. Ernst, N. S. Kim, S. Das, S. Pant, R. Rao, T. Pham, C. Ziesler, D. Blaauw, T. Austin, K. Flautner, T. Mudge, "Razor: A low-power pipeline based on circuit-level timing speculation," presented at the IEEE/ACM Int. Symposium Microarchitecture, 2003.

[3] S. Das, C. Tokunaga, S. Pant, W.H. Ma, S. Kalaiselvan, K. Lai, D.M. Bull, D.T. Blaauw, "RazorII: In situ error detection and correction for PVT and SER tolerance," IEEE J. Solid-State Circuits, 44(1): 32–48, 2009.

[4] S. Kim, I. Kwon, D. Fick, M. Kim, Y.P. Chen, D. Sylvester, "Razor-lite: A side-channel error-detection register for timing-margin recovery in 45nm SOI CMOS," in Proc. 2013 IEEE ISSCC: 264–265.

[5] F. Firouzi, F. Ye, K. Chakrabarty, M.B. Tahoori, "Representative critical-path selection for aging-induced delay monitoring," in Proc. 2013 IEEE International Test Conference (ITC): 1-10, 2013.

[6] M. Fojtik, D. Fick, Y. Kim, N. Pinckney, D.M. Harris, D. Blaauw, D. Sylverster, "Bubble razor: Eliminating timing margins in an ARM cortex-M3 processor in 45 nm CMOS using architecturally independent error detection and correction," IEEE J. Solid-State Circuits, 48(1): 66–81, 2013.

[7] K. Bowman, J. Tschanz, C. Wilkerson, S. L. Lu, T. Karnik, V. De, S. Borkar, "Circuit techniques for dynamic variation tolerance," in proc. Design Automation Conference (DAC), 2009.

[8] M. Sadi, L. Winemberg, M. Tehranipoor, "A robust digital sensor IP and sensor flow for in-situ path timing slack monitoring in SoCs," in Proc. IEEE 33rd VLSI Test Symposium (VTS), 2015.

[9] S. Sarrazin, S. Evain, I. Miro-Panades, L.A.D. B. Naviner, V. Gherman, "Flip-flop selection for in-situ slack-time monitoring based on activation probability of timing-critical paths," in Proc. IEEE 20th Int. On-Line Testing Symposium (IOLTS), 2014.

[10] A. Benhassain, F. Cacho, V. Huard, M. Saliva, L. Anghel, C. Parthasarathy, A. Jain, F. Giner, "Timing in-situ monitors: implementation strategy and applications results," in Proc. IEEE Custom Integrated Circuits Conference (CICC), 2015.

[11] L. Lai, V. Chandra, R. C. Aitken, P. Gupta, "SlackProbe: A flexible and efficient in situ timing slack monitoring methodology," IEEE Trans. Comput. Aided Des. Integr. Circuits Syst., 33(8): 1168-1179, 2014.

[12] N. Pour Aryan, G. Georgakos, D. Schmitt-Landsiedel, "Reliability Monitoring of Digital Circuits by in situ Timing Measurement," in

Proc. 23rd International Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS), 2013.

[13] D. Fick, N. Liu, Z. Foo, M. Fojtik, J. S. Seo, D. Sylvester, D. Blaauw, "In situ delay-slack monitor for high-performance processors using an all-digital self-calibrating 5ps resolution time-to-digital converter," presented at the IEEE ISSCC, San Francisco, CA, USA, 2010.

[14] X. Wang, M. Tehranipoor, R. Datta, "Path-RO: A novel on-chip critical path delay measurement under process variations," presented at the IEEE/ACM ICCAD, San Jose, CA, USA, 2008.

[15] H.A. Balef, H. Fatemi, K. Goossens, J.P.D. Gyvez, "Timing speculatoin with optimal In-Situ monitoring placement whithing-cycle error prevention," IEEE Trans. Very Large Scale Integr. VLSI Syst., 27(5): 1206-1217, 2019.

[16] H.A. Balef, K. Goossens, J.P.D. Gyvez, "Chip health tracking using dynamic In-Situ delay monitoring," presented at the 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE), Florence, Italy, May, 2019.

[17] Y.G. Chen, I.C. Lin, Y.C. Wei, "A novel NBTI-Aware chip remaining lifetime prediction framework using machine learning," presented at ISQED, Santa Clara, CA, USA, April, 2021.

[18] F. Nakhaee, M. Kamal, A. Afzali-Kusha, M. Pedram, S.M. Fakhraie, H. Dorosti, "Lifetime improvement by exploiting aggressive voltage scaling during runtime of error-resilient applications," Integr. VLSI J., 61: 29-38, 2018.

[19] H. Reyserhove, W. Dehaene, "Design margin elimination through robust timing error detection at ultra-low voltage," in Proc. IEEE SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S), 2017.

[20] Y. Sazeides, A. Bramnik, R. Gabor, C. Nicopoulos, R. Canal, D. Konstantinou, G. Dimitrakopoulos, "2D error correction for F/F based arrays using In-Situ Real-Time Error Detection (RTD)," in Proc. IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2020.

[21] M. Faryabi, H. Dorosti, M. Modarresi, S.M. Fakhraie, "Process variation-aware approximation for efficient timing management of digital circuits," in Proc. IEEE East-West Design and Test Symposium (EWDTS), 2015.

[22] Z. Wang, A.C. Bovik, H.R. Sheikh, E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Trans. Image Process., 13: 600-612, 2004.

[23] J. Rabaey, "Low Power Design Essentials," Springer Science & Business Media, USA, 2009.

[24] E. Karl, D. Blaauw, D. Sylvester, T. Mudge, "Reliability modeling and management in dynamic microprocessor-based systems," in Proc. the 43rd ACM/IEEE in Design Automation Conference: 1057–1060, 2006.

[25] F. Oboril, M.B. Tahoori, "Reducing wearout in embedded processors using proactive fine-grain dynamic runtime adaptation," in Proc. the 17th IEEE European Test Symposium (ETS): 1–6, 2012.

[26] B. Soltani, H. Dorosti, M. E. Salehi, S. M. Fakhraie, "Ultra-low-energy DSP processor design for many-core parallel applications," JECEI, 8 (1): 71-84, 2019.

[27] H. Dorosti, et al., "Ultralow-energy variation-aware design: adder architecture study," IEEE Trans. Very Large Scale Integr. VLSI Syst., 24(3): 1165-1168, 2016.

[28] A. Teymouri, H. Dorosti, M.E. Salehi, S.M. Fakhraie, "Energy-efficient variation-resilient high-throughput processor design," JECEI, doi: 10.22061/JECEI.2021.8253.499, 2022.

[29] M.B. Taylor, "A landscape of the new dark silicon design regime," Micro, IEEE Micro, 33(5): 8-19, 2013.

[30] H. Esmaeilzadeh, et al., "Dark silicon and the end of multicore scaling," in Proc. 2011 38th Annual International Symposium on Computer Architecture (ISCA): 365-376, 2011.

Biographies

Hamed Dorosti was born in Khoy, in 1986 and received the B.S. and M.S. degree in computer engineering from University of Tehran, Tehran, Iran, in 2009 and 2011, respectively. He received his Ph.D. in computer engineering (computer architecture) from University of Tehran in 2017. Since 2009, he was member of Silicon Intelligence and VLSI Signal Processing Lab., University of Tehran and co-operated in low-power ASIP project from 2010 to 2012. His research interest includes VLSI design, digital signal processing, adaptive timing error detection and correction and low-power high-throughput/performance processor architecture design considering static and dynamic variations. He is now an assistant professor of Shahid Rajaee Teacher Training University.

- Email: hdorosti@sru.ac.ir
- ORCID: 0000-0001-6554-1607
- Web of Science Researcher ID: L-5928-2019
- Scopus Author ID: 36662029700
- Homepage: https://www.sru.ac.ir/dorosti

How to cite this paper:
H. Dorosti, "Adaptive energy-efficient variation-aware dynamic frequency management," J. Electr. Comput. Eng. Innovations, 10(2): 477-486, 2022.

DOI: 10.22061/JECEI.2022.8331.507

URL: https://jecei.sru.ac.ir/article_1701.html