



Citation	Hans Reyserhove, Wim Dehaene, (2017) Design Margin Elimination Through Robust Timing Error Detection at Ultra-Low Voltage Proceedings of the SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S), San Francisco, US, 2017
Archived version	Author manuscript: the content is identical to the content of the published paper, but without the final typesetting by the publisher
Published version	https://doi.org/10.1109/S3S.2017.8308743
Journal homepage	http://s3sconference.org
Author contact	Hans.Reyserhove@esat.kuleuven.be + 32 (0)16 321169

(article begins on next page)



Design Margin Elimination Through Robust Timing Error Detection at Ultra-Low Voltage

Hans Reyserhove and Wim Dehaene

KU Leuven, ESAT-MICAS, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium

Email: Hans.Reyserhove@esat.kuleuven.be, Wim.Dehaene@esat.kuleuven.be

Abstract—This paper discusses a timing error masking-aware ARM Cortex M0 microcontroller system. Through in-path timing error detection, operation at the point-of-first-failure is possible without corrupting the pipeline state, effectively eliminating traditional timing margins. Error events are flagged and gathered to allow dynamic voltage scaling. The error-aware microcontroller was implemented in a 40nm CMOS process and realizes ultra-low voltage operation down to 0.29V at 5MHz consuming 12.90pJ/cycle, or a MEP of 11.11pJ/cycle at 7.5MHz. Measurements show the *in situ* approach is ideal to overcome traditional SS corner design margins (75% energy reduction). Additionally it overcomes the limitations introduced by replica path based techniques typically plagued by intradie variations (8% reduction).

Index Terms—CMOS digital integrated circuits, nearthreshold logic, better-than-worst-case design, transmission gate logic, variation resilience, timing margin elimination, soft edge flip-flop, time borrowing, transition detector, error detection, error masking, point-of-first-failure, *in situ*, ultra-low voltage.

I. INTRODUCTION

Ultra-low voltage operation is a widely used strategy to decrease dynamic energy consumption in digital systems. While decreasing the supply voltage seems straightforward, device sensitivity to variations increases significantly in the weak inversion regime. A typical way of handling these variations is taking margins at design time. Although these margins are effective, they are suboptimal both in energy and speed. Alternatively, on-die process monitoring in the form of (critical path) replicas can reduce these margins. However, mismatch between the replica and the actual system again requires margins, especially at the lowest supply voltages. Operating at the point-of-first-failure (PoFF) through *in situ* timing error detection can eliminate these margins [1], [2]. The overhead introduced by such a detection system can be overcome due to a reduced operating voltage at identical speed, hence decreased total energy. In [3] a system that implements a timing error masking aware strategy is presented. The focus of this work is 3-fold: 1) the relevant trade-offs are discussed; 2) the integration of the strategy in the digital design flow with commercial tools is shown; and 3) it compares measured speed and energy performance of the error detection system with identical designs using no margin, SS corner design margin or margin based on on-die ring oscillator monitoring.

II. TIMING ERROR DETECTION SYSTEM

The timing error detection strategy is shown in Fig. 1. It consists of a soft-edge flip-flop [6], a transition detector and an error latch. They are being controlled by a timing/control block. The applied strategy consists of detecting data transitions in a predefined window after the triggering clock edge. In the soft-edge flip-flop, the master latch is kept transparent during this window. Hence, data arriving in this window can still propagate, preventing pipeline corruption (error masking). Concurrently, a transition detector is enabled

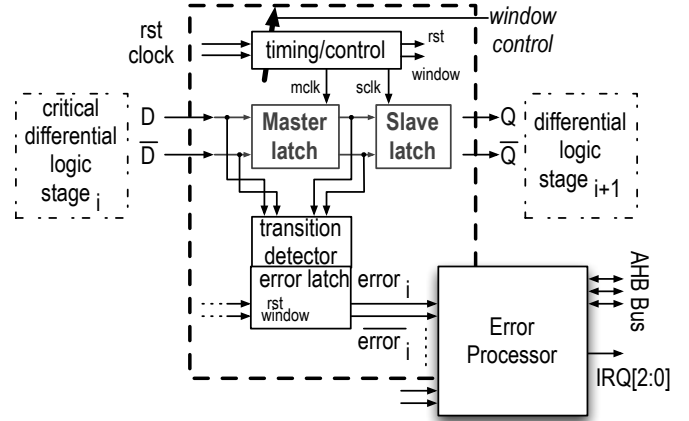


Fig. 1. Overview of the timing error-aware soft edge flip-flop, consisting of a timing/control block, a flip-flop, a transition detector and an error latch.

to flag data arriving late. The transition detector in its turn enables a set/reset latch (error latch), storing the flag for for the duration of the current clock cycle. This allows error data from the full system to propagate through the entire pipeline. The timing/control block generates clocks for the flip-flop and enables the transition detector and error latch. While the soft-edge flip-flops still propagates data arriving late, this data will also arrive late in the subsequent pipeline stage and thus have an effect at system level, much like time borrowing in a latch based pipeline.

A. Soft-Edge Flip-flop

While the soft-edge flip-flop has a separate master and slave clock, it operates as a normal flip-flop due to the integration with the timing/control block: a delayed master clock, a slave clock and a timing window signal are derived from the original clock. The delayed master clock results in a significant hold time increase, resulting in the trade-off explained in section II-D.

B. In-Latch Transition Detection

The transition detector (TD) translates any rising edge transitions on either input of the flip-flop to a pulse by comparing the input and output of the master latch. The transition detector is sized to make it fast enough to be triggered during the 1-1 overlap of these signals and create a pulse on the edge signal. The complementary structure of the TD enables ultra-low voltage operation. By using the master latch as a delay element, no additional transition detection logic needs to be inserted in the data path. Detecting such a transition after the clock edge is only possible due to the soft-edge flip-flop equipped in this system.

C. Error Signaling

The pulse signal created by the transition detector triggers a set/reset latch when it occurs during a predefined timing

TABLE I
SUMMARY OF APPLIED ERROR DETECTION STRATEGY AND COMPARISON WITH LITERATURE.

	This work	TIMBER [2]	Razor I [4]	Razor II [4]	Kim <i>et al.</i> [5]
Mechanism	Error masking FF	Error masking FF	Error detection FF	Error detection FF	Error detection latch
Correction	Masking	Masking	Pipeline stall	Instruction replay	V _{dd} boost
Evaluation event	After clock edge	Neg. clock edge	After clock edge	After clock edge	Neg. clock edge
Window generation	Local	Local/pipeline stage	None	Local	None
Hold time buffering	Increase	Increase	Increase	Increase	High increase
Detection mechanism	In-latch TD	Double sampling	Double sampling	TD	Double sampling
Timing analysis	Yes	/	/	/	/
Timing margin recovery	Full	Full	Full	Full	Full
ULV enabled	Yes	No	No	No	Yes

window after the rising edge of the clock. This event qualifies the transition as a timing error and sets the error latch. Because this occurs after the original clock edge, the event can be considered as if it had been an actual timing error. The error latch stores the error signal and propagates it through an error processor before the next rising edge of the clock, when it is reset to be able to flag a new error. The error processor consists of a timing slack prioritized OR-tree, a running mean error register, an adder and several interrupts to signal programmable timing error events to the main processor.

D. Timing Detection Window

The master latch clock is created locally inside each soft-edge flip-flop. The amount of time the master latch clock is delayed compared to the slave latch clock directly determines the timing detection window and is crucial to the functionality of this system for a number of reasons. First, increasing the window size requires a larger delay to create the window, hence more energy overhead. Second, the window size directly impacts how much time can be borrowed from the subsequent pipeline stage and thus how critical it can become. Third, reducing the window means the chance of a non-monitored path becoming slower (and thus failing) before a monitored path increases. Fourth, increasing the window size directly impacts the amount of hold time buffers necessary to account for the increased hold time in the error detection flip-flop. Fifth, the window should be large enough for the transition detector to set the error latch, limiting the lower bound of the window. Last, effective timing error monitoring should be possible: in a DVS system during voltage scaling, the window should be large enough to move from no errors to detected errors without compromising correct functionality. Since the resolution of the DC/DC converter generating the supply voltage is limited, the window should be large enough to account for this.

E. Summary

Table I shows a summary of the error detection strategy and compares with literature. The error masking soft-edge flip-flop overcomes the need for true error correction, which eliminates a significant part of the overhead present in similar systems. Evaluating the data after the clock edge allows full timing margin recovery. The unique in-latch transition detection introduced in this work allows error detection without double sampling or other overhead. The full integration of the strategy in the digital design flow with commercial tools is key (see section III). Finally, the entire circuit implementation and design flow is focused on ultra-low voltage application, hence the ultra-low operating voltage (see section IV).

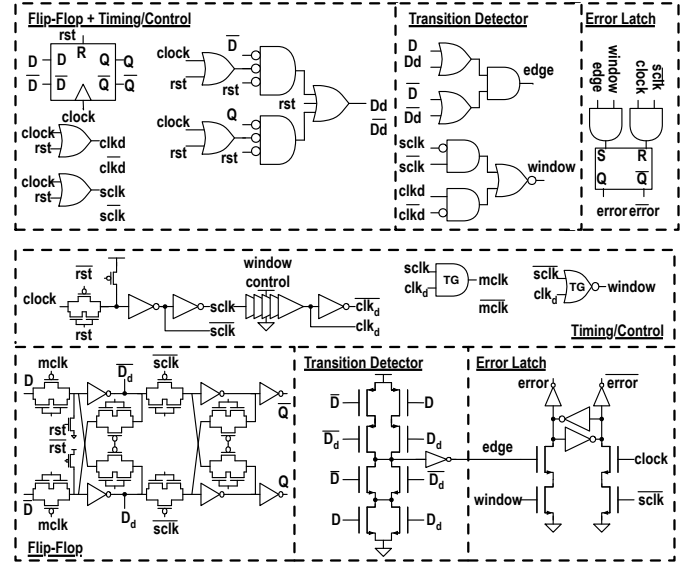


Fig. 2. Top: Functional division of the error detection flip-flop for the purpose of characterization. Bottom: Circuit implementation of the error detection flip-flop.

III. MODELING & DESIGN FLOW

To use the proposed timing error detection system (section II) in an automated standard cell design flow, its functionality should be formalized and characterized. As shown in Fig. 2 the complex timing error detection flip-flop is split in 3 cells, making it easier to handle its different functions. Correct characterization in subblocks is the only way to enable full digital implementation of such a timing error detection system. A functional division is made so that timing races are split across different blocks. This facilitates straightforward characterization for different slews, loads, corners and supplies. Different detection windows can be analyzed by characterizing multiple libraries and enabling them in multi-mode optimization.

The procedure to implement the error detection system in a standard cell design flow is shown in Fig. 3. After the initial design flow without error detection, critical paths are extracted. A subset of flip-flops based on critical endpoint analysis is replaced by error detection flip-flops. Their respective error signals are wired to the error processor through a variable size OR-tree. A second back-end iteration completes the design and enables full timing analysis with error detection flip-flops in the commercial tool flow.

IV. IMPLEMENTATION & MEASUREMENT RESULTS

The design flow presented in section III is used to implement a 32-bit ARM Cortex M0 based microcontroller system

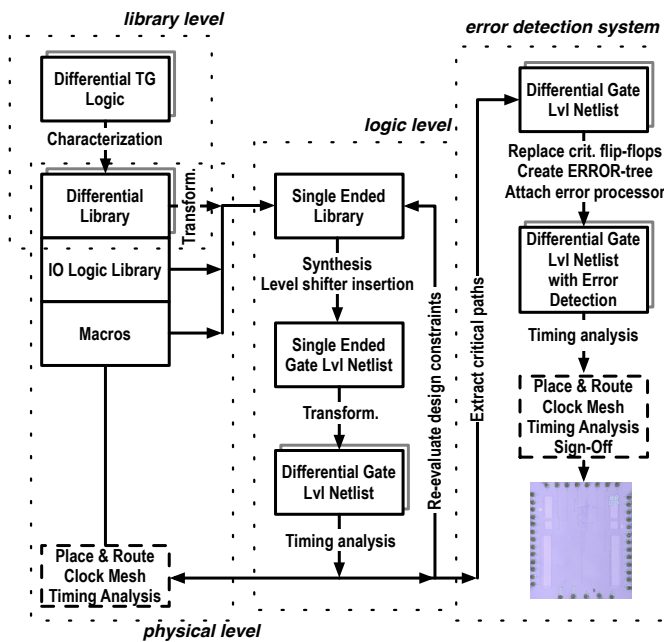


Fig. 3. Digital design flow including error detection system realization applied to the differential transmission gate design flow as proposed in [7].

as shown in Fig. 4. More implementation details can be found in [3]. The timing error detection enables automatic dynamic voltage scaling and finds the PoFF for a predefined range of target frequencies. Fig. 5 shows the measured energy consumption, while comparing it to an identical system in 3 configurations: without design margins [7]; with SS corner design margins; and with on-die process monitoring through a ring oscillator based replica circuit. The microcontroller with error detection system achieves minimum energy operation at 7.5MHz, corresponding to a 306mV supply voltage at 11.11pJ/cycle. Despite the overhead due to error detection, the full system outperforms a system with traditional design margins by far: a 75% energy reduction is achieved compared to the measured energy of the baseline design with SS corner margins. For the baseline design, ring oscillator frequencies are measured at the corresponding PoFF voltage and mapped to the critical path frequency taking into account inter- and intradie variations. As such, the system speed is predicted with some margin. At this speed and voltage, energy is measured on a baseline architecture without error detection. Despite the effort to match the ring oscillator speed to the system speed, the replica circuit based system suffers from intradie variations at the lowest supply voltages. Hence, timing error detection is superior here. At the MEP, an 8% energy reduction is achieved compared to the replica based design. At higher voltages, both the system and the replica are less sensitive to intradie variations, requiring less margin to operate correctly.

V. CONCLUSION

This work discusses a timing-error aware microcontroller system and its implementation in 40nm CMOS. Soft-edge flip-flops are equipped to monitor timing errors. Error information is collected in an error processor and controls the closed loop DVS system. This overcomes the need for traditional design margins and enables the system to operate at its PoFF. Total energy consumption is reduced, even when taking into account overhead due to error detection. The system outperforms

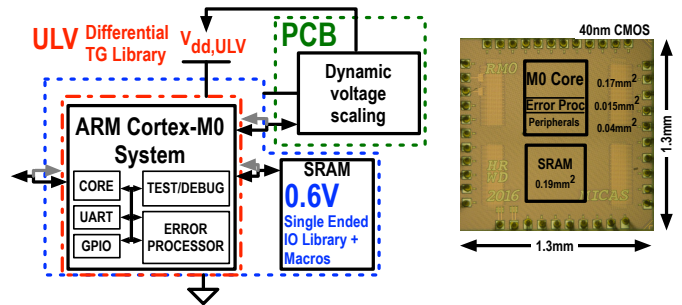


Fig. 4. Overview of the microcontroller system and chip implementation in 40nm CMOS.

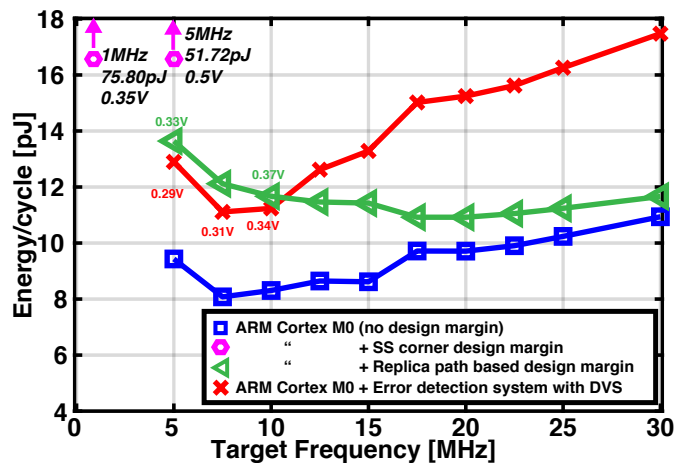


Fig. 5. Measurements of the error detection system compared to different cases with or without margin.

different cases of design margins, showcasing the need for *in situ* timing error detection, especially in conditions where intradie variations are predominant. The error detection system provides the microcontroller with a degree of robustness, far superior to margined designs. It enables standalone operation at the PoFF, which guarantees optimal energy consumption.

REFERENCES

- [1] S. Das, *et al.*, "A self-tuning DVS processor using delay-error detection and correction," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 4, pp. 792–804, April 2006.
- [2] M. Choudhury, V. Chandra, K. Mohanram, and R. Aitken, "TIMBER: Time borrowing and error relaying for online timing error resilience," in *Proceedings of the IEEE 2010 Design Automation Test in Europe Conference Exhibition (DATE)*, March 2010, pp. 1554–1559.
- [3] H. Reyserhove and W. Dehaene, "Design Margin Elimination in a Near-Threshold Timing Error Masking-Aware 32-bit ARM Cortex M0 in 40nm CMOS," *Proceedings of the IEEE 2017 European Solid-State Circuits Conference (ESSCIRC)*, September 2017.
- [4] S. Das, D. M. Bull, and P. N. Whatmough, "Error-Resilient Design Techniques for Reliable and Dependable Computing," *IEEE Transactions on Device and Materials Reliability*, vol. 15, no. 1, pp. 24–34, March 2015.
- [5] S. Kim and M. Seok, "Variation-Tolerant, Ultra-Low-Voltage Microprocessor With a Low-Overhead, Within-a-Cycle In-Situ Timing-Error Detection and Correction Technique," *IEEE Journal of Solid-State Circuits*, vol. 50, no. 6, pp. 1478–1490, June 2015.
- [6] M. Wiecekowski, *et al.*, "Timing yield enhancement through soft edge flip-flop based design," in *Proceedings of the IEEE 2008 Custom Integrated Circuits Conference (CICC)*, Sept 2008, pp. 543–546.
- [7] H. Reyserhove and W. Dehaene, "A Differential Transmission Gate Design Flow for Minimum Energy Sub-10-pJ/Cycle ARM Cortex-M0 MCUs," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 7, pp. 1904–1914, July 2017.