J. Low Power Electron. Appl. **2012**, *2*, 180-196; doi:10.3390/jlpea2020180

OPEN ACCESS

Journal of Low Power Electronics and Applications

ISSN 2079-9268

www.mdpi.com/journal/jlpea/

Article

Timing-Error Detection Design Considerations in Subthreshold: An 8-bit Microprocessor in 65 nm CMOS

Jani Mäkipää [1,*], Matthew J. Turnquist [2], Erkka Laulainen [2] and Lauri Koskinen [2]

[1] VTT Technical Research Centre of Finland, FI-02044 VTT, Finland

[2] Department of Micro- and Nanosciences, Aalto University, FI-00076 Aalto, Finland; E-Mails: matthew.turnquist@aalto.fi (M.J.T.); elaulain@ecdl.tkk.fi (E.L.); lkoskine@ecdl.tkk.fi (L.K.)

* Author to whom correspondence should be addressed; E-Mail: jani.makipaa@vtt.fi; Tel.: +358-(0)20-722-111; Fax: +358-(0)20-722-7001.

Received: 2 March 2012; in revised form: 29 May 2012 / Accepted: 30 May 2012 / Published: 6 June 2012

Abstract: This paper presents the first known timing-error detection (TED) microprocessor able to operate in subthreshold. Since the minimum energy point (MEP) of static CMOS logic is in subthreshold, there is a strong motivation to design ultra-low-power systems that can operate in this region. However, exponential dependencies in subthreshold, require systems with either excessively large safety margins or that utilize adaptive techniques. Typically, these techniques include replica paths, sensors, or TED. Each of these methods adds system complexity, area, and energy overhead. As a run-time technique, TED is the only method that accounts for both local and global variations. The microprocessor presented in this paper utilizes adaptable error-detection sequential (EDS) circuits that can adjust to process and environmental variations. The results demonstrate the feasibility of the microprocessor, as well as energy savings up to 28%, when using the TED method in subthreshold. The microprocessor is an 8-bit core, which is compatible with a commercial microcontroller. The microprocessor is fabricated in 65 nm CMOS, uses as low as 4.35 pJ/instruction, occupies an area of 50,000 μm^2, and operates down to 300 mV.

Keywords: subthreshold; ultra-low-power; timing-error detection; subthreshold source-coupled logic; SCL; weak inversion; dynamic supply voltage; dynamic voltage scaling

---

1. Introduction

Exploiting the full potential of ubiquitous ambient intelligence, smart sensor networks, and energy-harvesting, requires extremely low power processing. One of the saving graces is that, in many cases, power can be traded for performance, and thus the main target in these systems should be low energy per operation. Targeting low energy per operation, while simultaneously taking advantage of the relaxed performance requirements, can mainly be achieved by using a lower operating voltage. Low energy (and low power) operation extends the operating time of the systems, which reduces maintenance costs, device size, and unit cost. Systems with a small form-factor and low energy operation can also utilize alternate energy sources (e.g., they can harvest energy from body heat). These systems might be deployed in smart sensor network applications where it is cost prohibitive or not feasible to replace batteries [1–3].

In addition to sensor networks, a large number of applications exist that benefit from extremely low energy processing. One application that benefits from low energy processing is a fully autonomous robot capable of learning and adapting. The intelligence behind such robots is likely to be enabled by neuromorphic algorithms [4]. Such algorithms are inherently parallelizable, run efficiently on architectures resembling graphics processing units (GPU), and, if parallelized sufficiently, do not require high performance in a single processing element. Therefore, the brain behind a future small autonomous robot could very likely be a massively parallel computing unit running at a low energy point for a single processing node.

For CMOS static logic technologies down to 45 nm, the minimum energy per operation point (MEP) is achieved in the subthreshold operation region [3,5], thereby making subthreshold operation a target for the above-mentioned applications. However, design for the subthreshold region is more complicated than for strong inversion. The effects of process, supply voltage, temperature, and aging (PVTA) variance are amplified in the subthreshold region due to the exponential dependency of the subthreshold current on parameters that are susceptible to PVTA variance. Without intelligent design solutions, countering the increased variance effects requires large design margins or individual post-fabrication measurements of the components. In terms of these options, the former negates the minimum energy operation, while the latter increases production costs considerably. Further, in a massively parallel system these measurements would have to be performed separately for each processing node. Otherwise, the system would operate at the speed of the slowest node. In strong inversion, a popular solution for overcoming margining has been to use canary (replica) circuits [6]. However, canary circuits cannot compensate for local variations and, therefore, they are not suitable for subthreshold operation.

To compensate for global and local variations, timing-error detection (TED) can be used [6]. By allowing for the detection and correction of timing errors, TED systems are able to reduce the safety margins required to ensure the correct timing under PVTA variations [6–8]. Furthermore, TED can be used to mitigate thermal and power supply variations across the chip in massively parallel systems and take into account the effects of ageing without extra effort.

This paper presents a subthreshold TED microprocessor, which could represent a computation node for a future, massively parallel system. To our knowledge, this is the first known subthreshold TED system. The paper is organized as follows. Section 2 explains the characteristics and benefits of

subthreshold design, describes the motivation behind using TED techniques, and discusses previous works on subthreshold and TED. Section 3 explains the architecture and operation of the subthreshold TED microprocessor that we designed. Section 4 provides design and measurement results. Finally, we present our conclusions in Section 5.

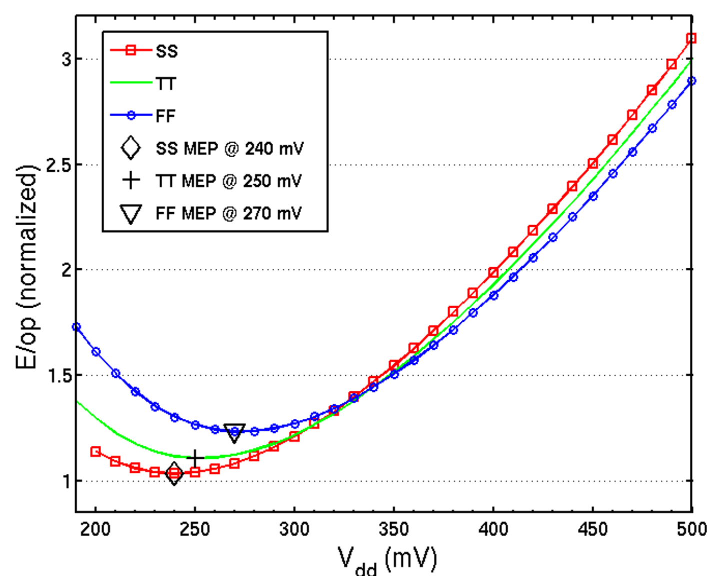
2. Background

2.1. Minimum Energy Point and Subthreshold Operation

The minimum energy point (MEP) denotes the operating point where the energy per operation is minimized. The energy per operation is composed of switching and leakage energy. Theoretically, the MEP for static CMOS logic depends on the technology [5]. For a given technology, the absolute MEP is tied to a certain threshold voltage. For newer technologies, there is typically a choice of devices with different threshold voltages (e.g., high threshold voltage, HVT, or low threshold voltage, LVT). These devices have their own respective MEPs which may differ from the absolute MEP. When the threshold voltage is fixed, the MEP is mainly dependent upon the technology and activity factor. For example, a 90 nm CMOS process has a MEP that ranges from 250 mV to 400 mV depending on the architecture and activity factor [3,5].

The MEP is situated in the subthreshold region for technologies down to 45 nm [5]. Figure 1 shows the MEP for a 65 nm process. A ring oscillator, with an activity factor (α) of 0.1, was used to generate the MEP curves. Different process corners change the leakage energy and, thus, change the MEP.

Figure 1. Simulation of a 65 nm ring oscillator with an activity factor (α) of 0.1. The energy per operation (E/op) is normalized to the SS corner.



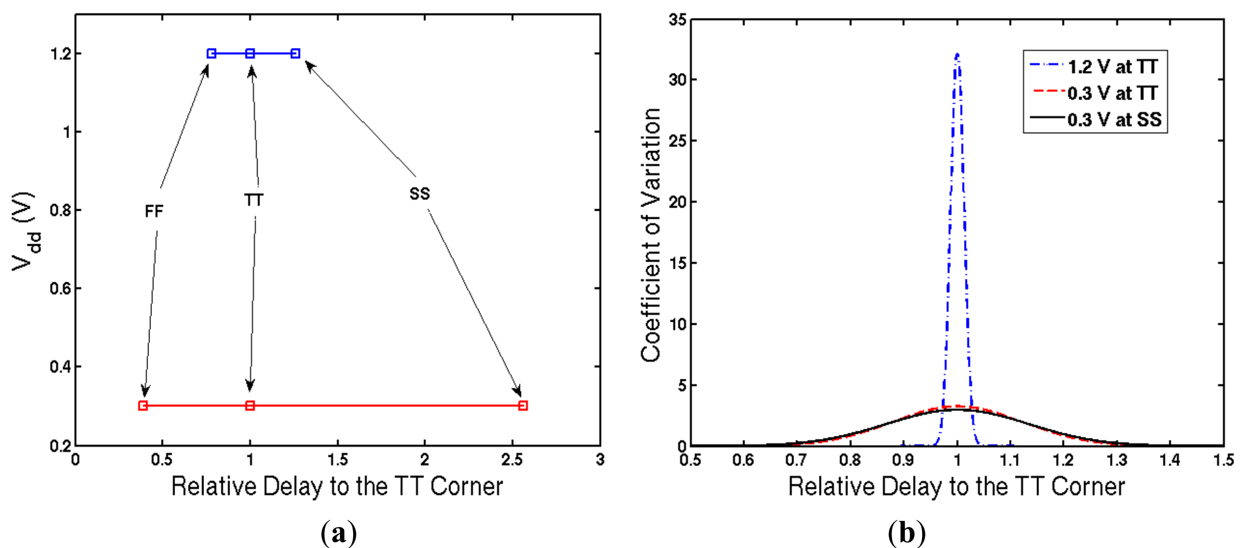
As shown in Figure 1, the MEP of 65 nm CMOS lies in the subthreshold region. The functional boolean design of static CMOS gates for the subthreshold region is comparable to a design for the strong inversion region with a few exceptions, which are mainly due to logic level deterioration due to leakage. However, in the subthreshold region, I_{ds} has exponential dependencies [3]:

$$I_{ds,sub} = I_O \exp\left(\frac{V_{gs} - V_t}{n\,V_{th}}\right)\left(1 - \exp\left(\frac{-V_{ds}}{V_{th}}\right)\right) \quad (1)$$

where I_O is the drain current when $V_{gs} = V_t$, V_t is the threshold voltage, n is the subthreshold slope factor, and V_{th} is the thermal voltage. As can be seen from Equation (1), PVTA variations cause exponential changes in the subthreshold current (e.g., a change in V_t due to process variations).

To show the impact of the exponential effects of Equation (1), different process corners and temperatures were simulated on an inverter chain. As shown in Figure 2(a), at 1.2 V, the SS and FF corners are, respectively, 1.26 and 0.78 times the delay at the TT corner. At 0.3 V, the SS and FF corners are, respectively, 2.56 and 0.39 times the delay at the TT corner. Low temperatures further exacerbate the variation impact. For example, the delay is 60 times larger at a voltage of 0.3 V, a temperature of −40 °C, and the SS corner, than at the TT corner. Figure 2(b) shows the coefficient of variation (σ/μ) for the local variance at 0.3 V and 1.2 V. The σ/μ at 0.3 V is 10 times larger than at 1.2 V. A 1000-point Monte-Carlo is used to generate both Figure 2(a) and (b).

Figure 2. (**a**) Relative delay compared to the TT corner at different corner and voltage combinations. The delays at each corner are the mean value of distribution generated from the Monte-Carlo runs; (**b**) Coefficient of variation for the TT and SS corners.



(**a**) (**b**)

Several subthreshold processors have been presented previously. In a recent study by Kwong *et al.* a 16-bit processor that is based on the MSP430 microcontroller and built in 65 nm is presented [9]. The processor achieves a frequency of 434 kHz and consumes 27.2 pJ/cycle at a V_{dd} of 0.5 V. In another study, Zhai *et al.* present an 8-bit custom ISA processor fabricated in 130 nm [10]. The processor achieves a frequency of 833 kHz and consumes 2.6 pJ/instruction at a V_{dd} of 360 mV.

Typically, the functionality of processors during variations in temperature is not analyzed in modern low voltage processors. Recently, Bol *et al.* addressed the issue of global PVT variation by utilizing a compensation system [11]. However, both the study by Kwong *et al.* [9] and the one by Zhai *et al.* [10] show frequency measurements over temperature but do not comment on the functionality of the circuit during variations in temperature. Prior studies also rarely focus on active variance robustness methods are also slightly commented. In a study by Hanson *et al.* [12], body bias is used to achieve variance

robustness in 130 nm technology. However, the effect of body bias decreases with smaller process nodes [2].

2.2. Timing-Error Detection

Timing-error detection (TED) has been shown to remove PVTA variation-incurred safety margins [6,7,13], which would conventionally guarantee operation across all corners with a sufficient yield. The lower safety margins can then either be turned into power savings (*i.e.*, lower V_{dd} [13]) or a higher yield [6]. The TED methodology is based on having the system operate at a voltage and frequency point in which the timing of critical paths fails intermittently. The failed timing occurrences are detected and corrected, for example, with an instruction replay system. If the error rate is low enough (e.g., 0.04% in a study by Blaauw *et al*. [13]), then an energy consumption benefit is achieved as a result of operating at a lower V_{dd}. If the error rate is too high, the instruction replay portion of the TED system begins to consume too much energy.

The key component of a TED system is an error-detection sequential (EDS) circuit. EDS circuits generate error signals when the path setup timing fails. This is also known as late signal detection, and it is a well-known synchronization concept. With a TED system, the EDS circuits are placed at critical logic paths where timing errors can occur. When using an EDS circuit, a timing error is flagged when a transition of D occurs in the TED window, as shown in Figure 3(a). The TED window for the EDS circuits can be tied to the clock signal [13], or it can be independently generated [6].

There are two main types of EDS architectures: a dynamic node [13–15] and a delayed shadow latch [7,8]. Of these architectures, the dynamic node can achieve a lower power and lower clock node capacitance. The dynamic node implementation typically uses an inverter delay chain and a logic gate (e.g., XOR) to produce a signal pulse. The signal pulse, or PULSE, as shown in Figure 3(a), is used to change the state of a dynamic node and generate a timing error signal. The inverters and logic gates used to produce the PULSE signal require a high level of precision across all PVTA variations, especially at low voltage levels. In addition to being robust, the size of the PULSE should be minimized since it limits the speed of the entire TED system as is further explained in Section 3.3.

Figure 3. (**a**) Basic timing-error detection (TED) operation with a dynamic node-style error-detection sequential (EDS) circuit. The generated PULSE signal is used to flip the state of a dynamic node and generate a timing error, or ERRf [15]; (**b**) Block diagram of a TED system.

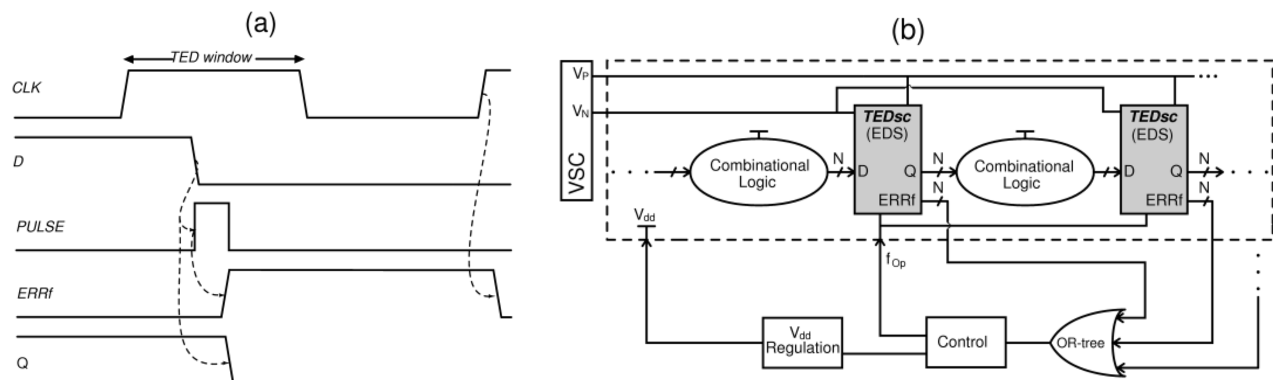


Figure 3(b) shows a high-level block diagram of a TED system using EDS circuits rather than normal FFs. The EDS circuits, called *TEDsc*, are explained in more detail in Section 3.3. Since the TED window is reserved for detecting timing errors in the previous clock cycle, no signals from the current clock cycle can arrive within the TED window. A signal that propagates too quickly through the combinational logic leads to a false error being flagged. Thus, the minimum delay for the combinational logic is the TED window. To prevent these false errors from being generated because of fast transitions, additional buffers are required as is described in more detail in Section 3.4.

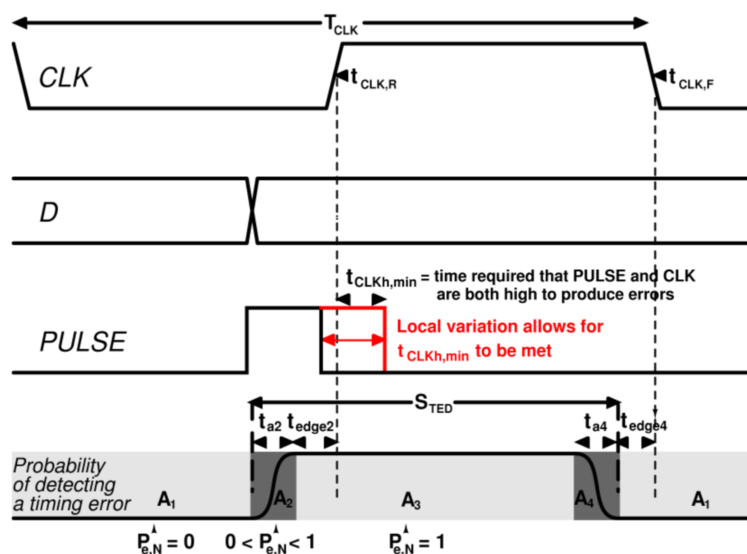
In practice, the minimum and maximum delay are both limited by the design uncertainties rather than by the logical operation. More specifically, the design of the EDS circuit defines two uncertainty regions, during which an error is captured with a finite probability (Figure 4). Local variation in an EDS circuit results in an uncertainty region at the microprocessor's clock signal (CLK) positive and negative edges, as shown in Figure 4. Near the CLK edges, the probability that N EDS circuits in a system would generate a timing error may not be 100% at some positions of the CLK (*i.e.*, uncertainty regions A_2 an A_4). In Figure 4, t_{edge2} and t_{edge4} are defined as the position before the positive CLK edge at which the probability of a timing error is 100% and the position before the falling CLK edge at which the probability of detecting a timing error is 0%, respectively.

Thus, the uncertainty region can be defined as the location within a CLK cycle (T_{CLK}) in which the probability of a timing error for N EDS circuits (EDS_0 to EDS_N) is between 0 and 100%:

$$P_{e,N} = 1 - \prod_{i=0}^{N-1} (1 - P_{e,EDSi}) \quad (2)$$

A study by Bull *et al*. refers to a similar concept at the positive CLK edge as setup pessimism [6]. For new processes or applications with weak to moderate inversion voltage levels, it is essential to understand the size and location of the uncertainty region. Since the uncertainty region is largely determined by the EDS circuit, it needs to be considered at the same time as the EDS design.

Figure 4. Local variation (e.g., in PULSE) causes the uncertainty region t_{a2} and t_{a4}. At the positive microprocessor's clock signal (CLK) edge, local variation in the PULSE signal makes it possible for $t_{CLKh,min}$ to be met earlier for some EDS circuits than for others.



3. Subthreshold TED Microprocessor

We studied timing error detection in a microprocessor that is capable of subthreshold operation. The central processor unit (CPU) that we implemented had an 8-bit core, which is compatible with a commercial microcontroller. The design was done in VHDL and the entire code was developed in-house for TED design testing purposes. By using an existing instruction set, we were also able to use of a readily available assembler and other software development tools.

3.1. Architecture

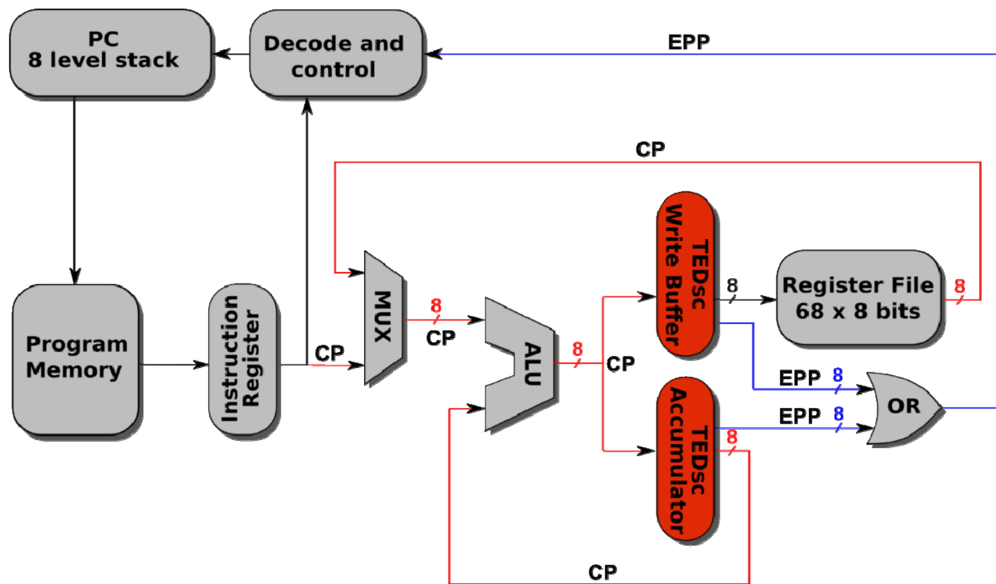
The architecture of the general purpose processor is an accumulator-based style in which the second operand is always the accumulator register. The processor core is pipelined into three stages: "Fetch", "Execution", and "Write". The instruction memory, which has a size of 256 bytes, resides in a separate block; the size of the block is 256 bytes. Due to design-time resource constraints, we do not consider here the memory design associated with the processor. The memory is designed for functionality and is not optimized in any way.

As explained in Section 2, the EDS-cells are inserted on the critical paths. The three-stage pipeline is configured so that the first stage, "Fetch," and the last stage, "Write," are shorter than the "Execute". Thus, the "Fetch" or the "Write" stages never fail before the "Execute" stage, and only the paths on the "Execute" stage had to be considered as potential candidates for critical paths. This design choice limits both the length of the clock cycle and the number of EDS circuits, and it facilitates the placement of the EDS latches by limiting the critical paths to one pipeline stage of the core. Since the error signals from the EDS circuits are combined using a logical OR tree, this design choice keeps the OR tree shallow. This simplifies the error control, keeps the control delay short, and reduces the control overhead. The study by Bull *et al.* solves this control delay by adding two stages to the pipeline [6]; in this study the clock cycle remains unchanged, but the clock cycles per instruction may increase. In the solution presented in this paper, the length of the clock cycle may be limited depending on how balanced the logic is between the pipeline stages of the core.

Figure 5 shows the block diagram of the core. The paths that can generate timing errors are highlighted in red. The core contains a total of 20 EDS circuits; 8 of them are in the accumulator register, 8 of them in the register file write buffer, and 4 of them are used for the arithmetic and logic unit (ALU) flags. The error signal paths are highlighted in blue.

The design requires more circuit modifications than a conventional design. For example, we inserted buffers on the fastest paths during the place and route stage to ensure that the hold time requirement for the TED error detection window was met. During the "Decode" stage, there are significant modifications to allow for error recovery.

Figure 5. *TEDsc*-enabled subthreshold microprocessor architecture. The timing error signal propagation paths (EPP) are highlighted in blue and the critical paths (CP) in red.



3.2. Timing-Error Detection and Recovery

Both the architecture of the core and the timing constraints set during the synthesis ensure that timing errors can only occur during the "Execution" stage. A timing error occurs when a data signal on a critical path arrives too late to the subsequent EDS data storage element (*i.e.*, the latch). At this point, incorrect values can be written to the accumulator and register file. In addition, the Program Counter (PC) and the stack might be incorrectly updated due to incorrect ALU flags.

After a timing error, the core needs to be able to restore the previous state using the following methods. First, when a timing error is detected, the system operation is halted by disabling the clocking. Next, the data stored during the previous cycle is restored (*i.e.*, the previous values of the PC, the accumulator, and the last stack push/pop are stored in the data FFs). Thus, the system stage becomes the previous stage. Finally, the failed instruction is re-executed using two clock cycles instead of one to guarantee an error-free operation. After the two clock cycle execution, the normal operation frequency is restored.

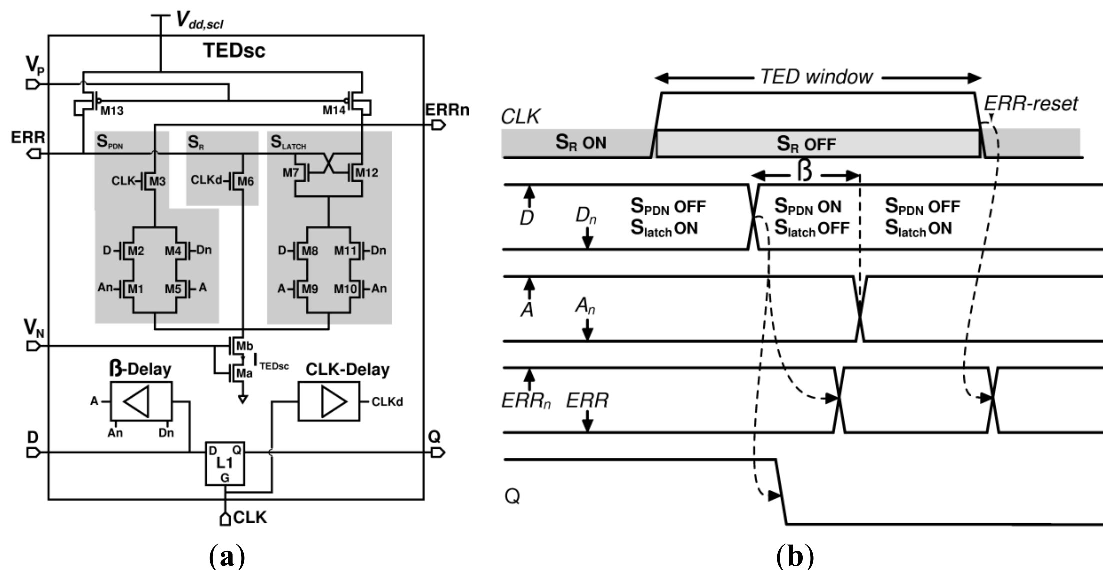
The error signals are not distinguished from each other, but are, instead, combined with one another. Thus, the system does not know which path generated an error. This arrangement is simple and it enables fast operation. With regards to functionality, it is not necessary to know on which path an error occurred.

Correct TED operation requires that signals do not arrive too early or late with respect to a TED window ($TED_{win,N}$), since these signals are not accounted for in real time at the system level or within the EDS. A signal that arrives too early has an insufficient delay time and, thus, it incorrectly arrives in the previous TED detection window ($TED_{win,N-1}$). In other words, a timing error is incorrectly generated (false positive). False positives are avoided by constructing correctly sized delay buffers. When a signal arrives too late (*i.e.*, at $TED_{win,N+1}$), it means that the delay is too large and that an error has not been correctly detected. To avoid these false negatives, timing constraints within the design are implemented to ensure that a signal cannot be delayed too greatly.

3.3. TEDsc

TEDsc is an EDS circuit [Figure 6 (a)] that uses subthreshold source-coupled logic (STSCL) to detect timing errors [15]. Depending on the logic depth, the leakage current, the activity factor, and the operation frequency of a system, STSCL can have several advantages over static CMOS (e.g., tunability, reduced power consumption, and a decreased sensitivity to supply noise [16,17]). STSCL has been shown to be advantageous for ultra-low-power (ULP) systems.

Figure 6. (**a**) *TEDsc* circuit [15]; (**b**) *TEDsc* timing.

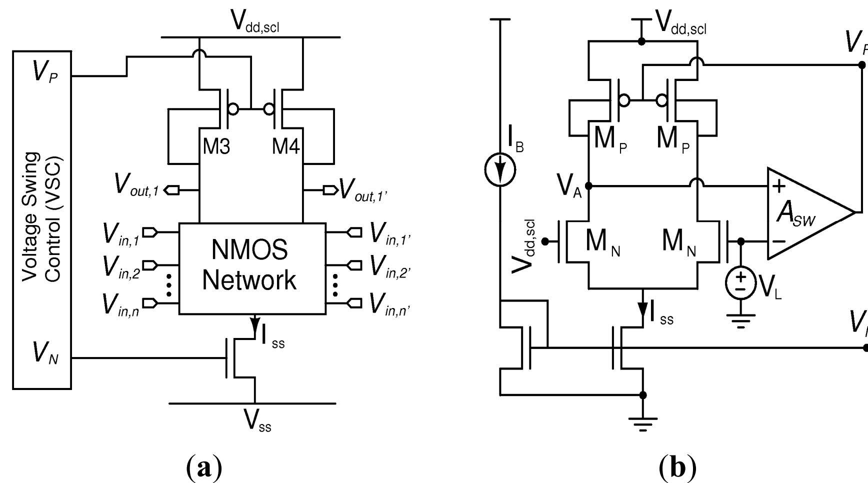


(**a**) (**b**)

An STSCL gate is composed of a network of differential NMOS pairs, an adjustable PMOS load (M_3,M_4) with output resistance R_P, and an adjustable tail current I_{SS} [Figure 7(a)]. The NMOS pairs are used to construct logic gates. The voltage swing is defined as $V_{SW} = R_P \cdot I_{SS}$, and it is maintained by dynamically adjusting the size of R_P and the magnitude of I_{SS}. Since I_{SS} can be reduced to the pA range, R_P needs to be in the GΩ range to achieve a proper V_{SW} (*i.e.*, $V_{SW} > 150$ mV). By connecting the bulk of the PMOS load devices to the drain, a large R_P is achieved without excessively large transistor lengths [16,17].

The size of R_P and the magnitude of I_{SS} are both adjusted by the voltage swing control (VSC) block as shown in Figure 7(b). The VSC decreases the dependence on global variations (e.g., supply noise, temperature fluctuations, and ageing). The VSC ensures a voltage swing greater than 150 mV across all global variations. The VSC for *TEDsc* uses a two-stage, miller-compensated opamp for ASW. The opamp is able to maintain an open loop gain of 40 dB for all the global process corners. The bias voltage (V_P) from one VSC can be used for a large number of *TEDsc* gates [16].

Figure 7. (**a**) Subthreshold source-coupled logic (STSCL) circuit; (**b**) Voltage swing control (VSC).



(**a**) (**b**)

Since *TEDsc* uses STSCL, it has the unique ability to adjust its D-to-timing error delay (D-ERRf delay); this results in an adjustable TED window. This ability to adjust the D-ERRf delay can be explained by first understanding that during a D transition, *TEDsc* requires a minimum amount of charge (Q_{emin}) to move from the dynamic output node in order to induce a differential timing error [15]. Reaching Q_{emin} is dependent on I_{TEDsc} and the β-delay that is extended under the CLK high (*i.e.*, $t_{\beta CLK}$). For example, when I_{TEDsc} is increased, the TED window is widened at both of the CLK's edges since the required $t_{\beta CLK}$ is decreased to meet Q_{emin}.

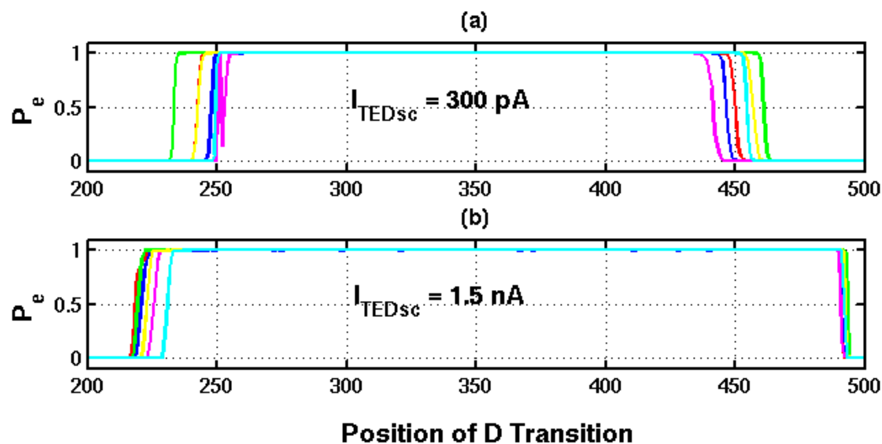
The starting point of the TED window ($t_{a2} + t_{edge2}$ from Figure 4) has two important implications. First, at the positive CLK edge, an excessively early starting point of the TED window (*i.e.*, $(t_{a2} + t_{edge2})/\mathrm{T_{CLK}}$ is too large) does not allow for the maximum clock frequency to be reached and, thus, the energy consumption is increased. Second, for a flip-flop based pipeline, an overly delayed TED window starting point (*i.e.*, due to a low I_{TEDsc}) does not correctly report all setup time failures as timing errors, which results in a non-functional design. In the presence of large global variation susceptibility, as found in subthreshold, the tunable TED window enables fine tuning on the system level.

Fine tuning of the TED window is achieved by adjusting I_{TEDsc} within *TEDsc*. To understand how the I_{TEDsc} affects the TED window, three *TEDsc* circuits were measured on the same die. *TEDsc* and VSC used the following settings: $V_{dd,\mathrm{scl}}$ = 400 mV, V_L = 200 mV, and V_{dd} = 300 mV. A total of 500 positions of D were applied as input to *TEDsc*. There were 16,384 transitions of D at each of the 500 positions. The duty cycle of the CLK was at 50%. The TED window for *TEDsc* in Figure 8 is located between (Position of D Transition) 250 and 500. Figure 8 shows the error probability of the three *TEDsc* circuits as a function of the D transition. For this measurement, the frequency of the CLK was 10.37 kHz.

As shown in Figure 8, by adjusting I_{TEDsc}, *TEDsc* can adjust its D-ERRf delay. This subsequently makes fine tuning of the TED window (and the uncertainty region) possible. For example, to reduce the D-ERRf delay, I_{TEDsc} was increased from 300 pA to 1.56 nA (Figure 8). In previous designs [14,15], the uncertainty region and TED window have been fully defined at design time, which is not favorable

for weak inversion TED design. Simulations showed an uncertainty region (*i.e.*, A_2, A_4) of approximately the same size as found in measurement [15].

Figure 8. (**a**) I_{TEDsc} at 300 pA; (**b**) When I_{TEDsc} is increased to 1.5 nA, the size of the TED window increases.



As the microprocessor's performance is altered by local and global variations, it is essential that the EDS circuit operate correctly and accurately. Through simulations, *TEDsc* was shown to be robust to both local and global variations. Local variations were accounted for by applying Monte-Carlo simulations at each process corner (*i.e.*, TT, FF, SS, SF, and FS). This simulation also showed a robustness to global process corners due to the VSC. Additionally, *TEDsc* showed a correct functionality from −40 °C to 90 °C as a result of the VSC. Using STSCL also reduces the sensitivity of *TEDsc* to changes in the supply voltage [17]. In addition, the probability of a fast change in the supply voltage at the exact same time that D transitions is low. To verify this, we applied a sawtooth-wave ripple voltage from 0 to 40 mV and a frequency from 10 MHz to 100 MHz to *TEDsc*; the correct functionality was shown under these ripple conditions.

The effects of local variations on *TEDsc* are minimized by proper sizing techniques developed by Wang, Calhoun and Chandrakasan [3] and Alioto and Leblebici [18]. The effects of global variations on *TEDsc* are minimized due the STSCL design choice. As mentioned in Section 3.3, STSCL uses the VSC to maintain proper operation during the application of both static and dynamic global variations [17]. As mentioned in Section 2.2, larger local variations increase the size of t_{a2} and t_{edge2}. This fundamentally limits the speed of the entire TED system since if $(t_{a2} + t_{edge2})/T_{CLK}$ is too large, there is not ample time to detect errors.

3.4. Implementation of Core 1 and 2

To compare the benefits of TED, we designed a TED-enabled core (Core 1) and a non-TED core (Core 2). The designs of both cores were fabricated in 65 nm CMOS. The supply voltage range of both designs is from 300 mV to 500 mV, which is at the edge or below the strong inversion region for the process and all the digital cells. However, we optimized *TEDsc* to work deep into subthreshold; the analysis below will only include 300 mV and 400 mV operation points.

To simplify the design process of Core 1, two power domains were used in the design. The instruction memory and the error propagation path are located within one power domain, while the rest of the design is in a second power domain. The size of the instruction memory is 256 instructions and the size of the register file is 68 bytes. The area of the TED core (without instruction memory) is approximately 50,000 μm^2. The length of the CLK period is approximately 160 times the FO4 delay. The clock period is limited by the "Execute" stage and EDS design.

The foundries did not provide digital EDA tool library information for subthreshold operation. To acquire the library's timing and power information for the EDA tools, we re-characterized the standard cells for subthreshold operation by using the Synopsys library characterization workflow. During the re-characterization process, we used the standard libraries as templates, considered all the timing arcs, and acquired the new timing and power information via analog simulation. The re-characterization process was repeated for the typical, best, and worst corners. The acquired library information was used by the EDA tools in the automated design flow. Due to their sensitivity variation in subthreshold, the smallest gates were removed from the libraries.

It was not possible to characterize the EDS element and include it to the digital library due to the asynchronous nature of the element's error signal. Furthermore, the VSC block that generates bias voltages for the *TEDsc* blocks is inherently analog. Therefore, a digital simulation of the full system was not possible. An analog simulation of the system would have been excessively long. Thus, we performed a mixed-mode simulation on the system. The VCS and *TEDsc* blocks were simulated using Spice transistor level models. All of the digital blocks were simulated using the post-layout netlist (including parasitics). Mentor Graphics Questa ADMS was used to perform the mix-mode simulation.

The die microphotograph of Core 1 (TED) and Core 2 without instruction memory is shown in Figure 9. Both Cores include all the logic, delays, and buffers. The VSC block and the EDS circuits are also shown in Core 1 (TED).

Figure 9. (**a**) The microcontroller core with and without TED are shown as Core 1 (TED) and 2, respectively; (**b**) Core 1 (TED) including the VSC and *TEDsc* circuits.

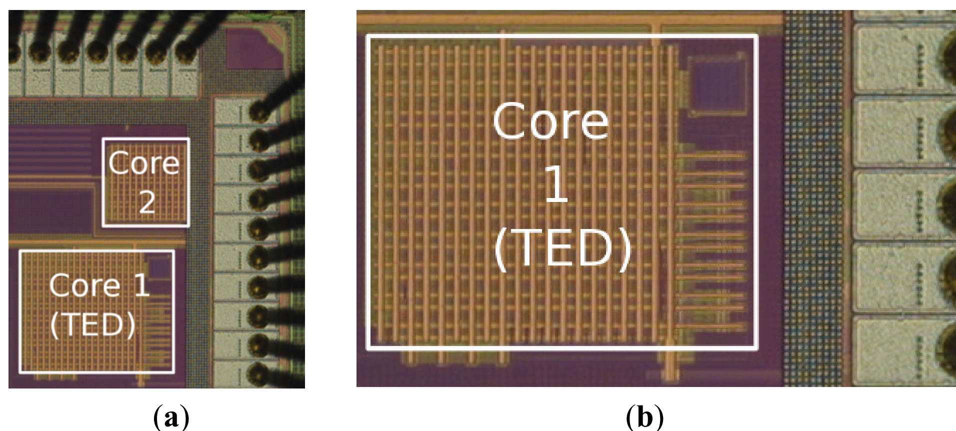


Table 1 shows a comparison of Core 1 and Core 2. The area of Core 2 is approximately 18,000 μm^2, which is approximately 64% smaller than that of the TED version. For the comparison, the chip I/O compatibility level-shifters present in the subthreshold version are excluded, which gives the total area for the TED version as approximately 50,000 μm^2. The VSC block occupies an area of approximately

1750 μm^2 in the subthreshold design. It should be noted that in a larger design, the VSC area gets proportionally smaller. The areas of the different blocks were measured so that only the active area occupied by the blocks was taken into account.

The data in Table 1 shows that both the clock delay cells and the buffer cells occupy a substantially larger area than in the nominal voltage design. The number and the area of the logic ports are comparable. The area of the data storage elements is approximately two times larger in the subthreshold design, which can be explained by the fact that the EDS cells in general are larger in area than their conventional style counterparts. This applies especially to the EDS circuits designed for subthreshold operation due to their variation immunity requirements as explained in Section 3.3. The area in the table that is unaccounted for is occupied by the decap and antenna protection elements, and in Core 1 by the VSC block. It should be noted that the Core 1 (TED) design has not been optimized area-wise. Also, the I/O port functionality has been excluded from the Core 1 design. This makes the comparison somewhat less favorable for the Core 1 design in terms of the area. Also, the error recovery mechanism modification adds to the logic size slightly. The last columns of the table show the percentage of the area of the Core 1 design compared to the area of the nominal voltage design (Core 2).

Table 1. An area comparison of Core 1 (TED) and Core 2.

	Core 2 (Total Area ≈ 18,000 μm^2)		Core 1 (TED) (Total Area ≈ 50,000 μm^2)		Area of Cells in Core 1 ÷ Area of Cells in Core 2 (*i.e.*, % larger area that Core 1 uses than Core 2)
	Number of Cells	% of the Total Area	Number of Cells	% of the Total Area	
Buffer Cells	139	2.5%	934	5%	538%
Clock Buffer Cells	223	3.5%	66	<1%	45%
Clock Delay Cells	37	1%	1580	21%	4644%
Data Storage Cells	777	35%	897	27%	205%
Logic Port Cells	1942	45%	2191	17%	108%
Filler Cells	563	10%	4756	19%	539%

4. Silicon Measurement Results

Measurements were done using an automated measurement setup that used Labview to manage the measurement instruments. Due to the nature of the effects of PVT variance in subthreshold, the correct start-up values for the supply voltage and operation frequency were known in advance. Thus, the aforementioned parameters were swept to adjust the core to the safe area of operation. This was accomplished by inputting test vectors to the core and monitoring the register dump and the timing error signals.

The test programs were coded using an assembler and uploaded to the instruction memory using a pattern generator. The error rate was recorded during run-time. After the program execution, the register dump was loaded from the chip, and the dumped register values were compared against known results to verify the correct functionality.

Tables 2 and 3 show shmoo plots for Core 1 (TED), running at 300 mV and 400 mV, respectively. The x-axis indicates the CLK operation period (T_{CLK}) and the y-axis indicates the duty cycle (D_{cycle}).

The green squares display the duty cycle and frequency pairs in which the circuit is able to operate correctly. As the duty cycle is increased, the size of the TED window is also increased since the amount of minimum delay is directly proportional to the size of the TED window size. Additionally, as the frequency is decreased, the minimum delay requirement increases.

Table 2 shows that the maximum usable duty cycle at V_{dd} = 300 mV is 20%–25%. Those values will still give approximately a 50% tuning range of the frequency. As Table 2 shows, the circuit does not function more quickly than 2.95 kHz. This limitation is set by the HVT logic speed at the supply voltage of 300 mV.

Table 3 shows that the maximum usable duty cycle at V_{dd} = 400 mV is 15%–20%. As Table 3 shows, the circuit does not function more quickly than 37.2 kHz. This limitation is set by the HVT logic speed at a supply voltage of 400 mV.

Table 2. Shmoo plot of Core 1 (TED) at V_{dd} = 300 mV. The maximum clock frequency, f_{max1}, is 2.95 kHz. The green squares (and checkmarks) display the duty cycle and frequency pairs in which the circuit is able to operate correctly.

	T_{CLK}													
D_{cycle}	**2000 Hz**	**2088 Hz**	**2180 Hz**	**2276 Hz**	**2237 Hz**	**2482 Hz**	**2591 Hz**	**2706 Hz**	**2825 Hz**	**2950 Hz**	**3080 Hz**	**3216 Hz**	**3358 Hz**	**3506 Hz**
2%														
5%	✓	✓	✓	✓	✓	✓	✓							
10%	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
15%	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
20%	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
25%						✓	✓	✓	✓	✓				
30%										✓				
35%														

Table 3. Shmoo plot of Core 1 (TED) at V_{dd} = 400 mV. For this Core, f_{max1} is 37.2 kHz.

	T_{CLK}													
D_{cycle}	**20.0 kHz**	**21.7 kHz**	**23.4 kHz**	**25.2 kHz**	**26.9 kHz**	**28.6 kHz**	**30.3 kHz**	**32.1 kHz**	**33.8 kHz**	**35.5 kHz**	**37.2 kHz**	**39.0 kHz**	**40.7 kHz**	**42.4 kHz**
2%														
5%	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
10%	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
15%	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
20%				✓	✓	✓	✓	✓	✓	✓	✓			
25%									✓	✓	✓			
30%														
35%														

To compare the energy per operation of Core 1 (TED) and Core 2, Core 1 (TED) was first set to a nominal V_{dd} (e.g., 300 mV). At this V_{dd}, or $V_{ddCore1}$, the maximum clock frequency (f_{max1}) was determined as explained in Table 2. In order to guarantee the operation of Core 2 under worst case conditions and at the same frequency as Core 1 (*i.e.*, $f_{max2} = f_{max1}$), a safety margin was found for Core 2. Similarly to other TED implementations [19], the safety margin was found from the worst case delay due to a SS process corner, a temperature of −40 °C, and a voltage droop of 10%. This worst case delay required that $V_{ddCore2}$ be increased from 300 mV to ensure that $f_{max2} = f_{max1}$. This increase in voltage increased the energy per operation of Core 2 relative to Core 1 when operating at f_{max1}.

Figure 10 shows the energy per operation for both cores. At 300 mV, Core 1 (TED) uses 28% less energy per operation than Core 2. At 400 mV, Core 1 and Core 2 consume approximately the same amount of energy per operation. However, Core 1 still has an advantage considering its ability to compensate for all local and global variations. Table 4 shows a summary for Core 1. It is important to note that at 400 mV, the operation speed is more than 10 times faster than at 300 mV while keeping energy consumption per operation essentially the same.

Figure 10. Energy per operation of both Core 1 (TED) and Core 2.

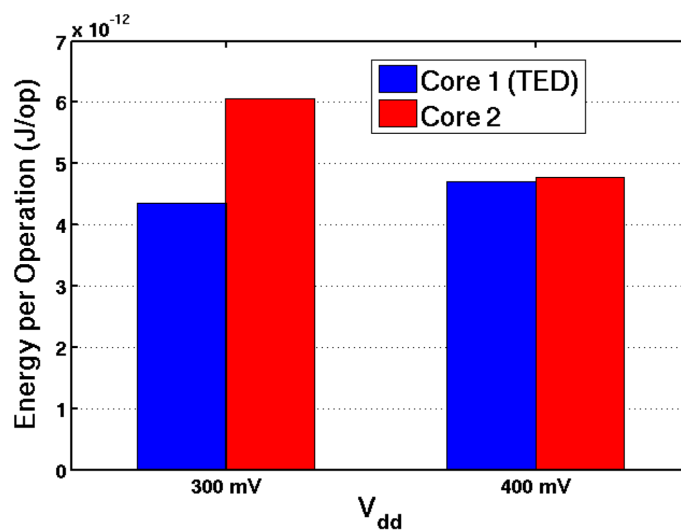


Table 4. Summary of the subthreshold-capable microprocessor (Core 1) performance in 65 nm CMOS.

Core 1 (TED) Summary		
Process Technology	65 nm CMOS	
Number of *TEDsc*'s	20	
Clock cycle length (T_{CLK})	160 FO4	
V_{dd}	**Clock Frequency**	**Energy/Operation (pJ/op)**
300 mV	2.95 kHz	4.35
400 mV	37.2 kHz	4.71

5. Conclusions

The presented microprocessor proves that timing-error detection (TED) is feasible in subthreshold and that TED reduces the energy per operation. When combined with a start-up algorithm, TED would guarantee a correct operation at time zero even under a wide range of global and local variations. However, the adaptive system presented here was not optimized, especially in terms of area. This was mostly due to the fact that conventional synthesis and place and route design was not optimized for subthreshold and the optimization for subthreshold here was restricted to characterizing the library. Thus, there is still great potential for optimization in the subthreshold TED system in order to further reduce energy consumption and area usage.

Acknowledgments

This work is funded by the Academy of Finland (Projects #124029, #140340, and #13139458), and the Finnish Graduate School of Electronics, Telecommunications, and Automation (GETA).

References

1. Shashank, P.; Inman, D. *Energy Harvesting Technologies*, 1st ed.; Springer: New York, NY, USA, 2009.
2. Rabaey, J. Optimizing Power at Standby: Circuits and Systems. In *Low Power Design Essentials*, 1st ed.; Springer: New York, NY, USA, 2009; Chapter 8.
3. Wang, A.; Calhoun, B.; Chandrakasan, A.P. *Sub-Threshold Design for Ultra Low-Power Systems*, 1st ed.; Springer: New York, NY, USA, 2006.
4. Versace, M.; Chandler, B. The brain of a new machine. *IEEE Spectr.* **2010**, *12*, 30–37.
5. Bol, D.; Ambroise, R.; Flandre, D.; Legat, J. Interests and Limitations of Technology Scaling for Subthreshold Logic. *IEEE Trans. Very Large Scale Integr. Syst.* **2009**, *17*, 1508–1519.
6. Bull, D.; Das, S.; Shivashankar, K.; Dasika, G.; Flautner, K.; Blaauw, D. A power-efficient 32 bit ARM processor using timing-error detection and correction for transient-error tolerance and adaptation to PVT variation. *IEEE J. Solid State Circ.* **2011**, *46*, 18–31.
7. Bowman, K.; Tschanz, J.; Lu, L.; Aseron, P.; Khellah, M.; Raychowdhury, A.; Geuskens, B.; Tokunaga, C.; Wilkerson, C.; Karnik, T.; De, V. A 45 nm resilient microprocessor core for dynamic variation tolerance. *IEEE J. Solid State Circuit* **2011**, *46*, 194–208.
8. Crop, J.; Krimer, E.; Moezzi-Madani, N.; Pawlowski, R.; Ruggeri, T.; Chiang, P.; Erez, M. Error detection and recovery techniques for variation-aware cmos computing: A comprehensive review. *J. Low Power Electron. Appl.* **2011**, *1*, 334–356.
9. Kwong, J.; Ramadass, Y.K.; Verma, N.; Chandrakasan, A.P. A 65 nm sub-V_t microcontroller with integrated SRAM and switched capacitor DC-DC converter. *IEEE J. Solid State Circ.* **2009**, *44*, 115–126.
10. Zhai, B.; Pant, S.; Nazhandali, L.; Hanson, S.; Olson, J.; Reeves, A.; Minuth, M.; Helfand, R.; Austin, T.; Sylvester, D.; Blaauw, D. Energy-efficient subthreshold processor design. *IEEE Trans. VLSI Syst.* **2009**, *17*, 1127–1137.

11. Bol, D.; De Vos, J.; Hocquet, C.; Botman F.; Durvaux, F.; Boyd, S.; Flandre, D.; Legat, J.-D. A 25 MHz 7 μW/MHz ultra-low-voltage microcontroller SoC in 65 nm LP/GP CMOS for low-carbon wireless sensor nodes. In *Proceedings of the 2012 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, San Francisco, CA, USA, 2012; pp. 490–492.
12. Zhai, B.; Seok, M.; Cline, B.; Zhou, K.; Singhal, M.; Minuth, M.; Olson, J.; Nazhandali, L.; Austin, T.; Sylvester, D.; Blaauw, D. Exploring variability and performance in a sub-200-mV processor. *IEEE J. Solid State Circuit* **2008**, *43*, 881–891.
13. Blaauw, D.; Kalaiselvan, S.; Lai, K.; Ma, W.H.; Pant, S.; Tokunaga, S.; Das, S.; Bull, D. Razor II: *In situ* error detection and correction for PVT and SER tolerance. In *Proceedings of the 2008 IEEE International Solid-State Circuits Conference*, San Francisco, CA, USA, 3–7 February 2008; p. 400.
14. Turnquist, M.J.; Laulainen, E.; Makipaa, J.; Pulkkinen, M.; Koskinen, L. Measurement of a timing error detection latch capable of sub-threshold operation. In *Proceedings of the 2009 IEEE NORCHIP Circuit Conference*, Trondheim, Norway, 16–17 November 2009; pp. 1–4.
15. Turnquist, M.J.; Laulainen, E.; Mäkipää, J.; Koskinen, L. Measurement of a system-adaptive error-detection sequential circuit with subthreshold SCL. In *Proceedings of the 2011 IEEE NORCHIP Circuit Conference*, Lund, Sweden, 14–15 November 2011; pp. 1–4.
16. Tajalli, A.; Leblebici, Y. Leakage current reduction using subthreshold source-coupled logic. *IEEE Trans. Circuit Syst. II* **2009**, 56, 374–378.
17. Tajalli, A.; Leblebici, Y. *Low-Power Mixed Signal IC Design*, 1st ed.; Springer: New York, NY, USA, 2010.
18. Alioto, M.; Leblebici, Y. Analysis and design of ultra-low power subthreshold MCML gates. In *Proceedings of the IEEE International Symposium on Circuit and Systems*, Taipei, Taiwan, 24–27 May 2009; pp. 2557–2560.
19. Bowman, K.; Tschanz, J.; Kim, N.; Lee, J.; Wilkerso, C.; Lu, S.; Karnik, T.; De, V. Energy-efficient and metastability-immune timing-error detection and instruction-replay-based recovery circuits for dyanmic-variation tolerance. In *Proceedings of the IEEE International Solid-State Circuits Conference*, San Francisco, CA, USA, 3–7 February 2008; pp. 402–403.

© 2012 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution license (http://creativecommons.org/licenses/by/3.0/).