Near-Threshold Processor Design Techniques for Power-Constrained Computing Devices

Jun Zhou, Tony Tae-Hyoung Kim[*], Yong Lian[**]
University of Electronic Science and Technology of China (jun.zhou.sg@ieee.org)
[*] *Nanyang Technological University, Singapore (thkim@ntu.edu.sg)*
[**] *York University, Canada (peterlian@eecs.yorku.ca)*

Abstract— **Ultra-low power digital processors are needed in power-constrained computing devices for IoT, intelligent sensing and wearable computing applications. While near-threshold design is promising to achieve ultra-low power consumption, there are several challenges for this technology to be applied to the design of digital processors, including the functionality issue, performance degradation and variability issue. This paper reviews the existing near-threshold processor design techniques including circuit and architecture level solutions and discuss how the aforementioned issues can be addressed.**

***Index Terms*—ultra-low power, near-threshold processor, functionality issue, performance degradation, variability.**

I. Introduction

Power constrained computing devices (e.g. intelligent sensing, wearable computing and IoT devices) require ultra-low power digital processors. Firstly, these devices are powered by battery or energy harvester. Ultra-low power consumption allows for long battery life or perpetual operation via energy harvester. Secondly, these devices require small device size. As battery and energy harvester usually dominates the device size, ultra-low power consumption allows for small battery or energy harvester, and therefore facilitates miniaturization of the device. Sub/near-threshold design techniques have been shown to be promising in achieving ultra-low power consumption which is not possible by conventional low power digital design techniques such as clock gating and power gating. Compared to other design-specific low power techniques, they are more general and applicable to different designs. The major benefit of sub/near-threshold design is shown in Fig. 1. When the supply voltage is scaled down, the dynamic energy consumption per operation of digital circuits decreases quadratically and the leakage energy decreases moderately, which makes the total energy per operation decreases dramatically with voltage. As the supply voltage approaches the transistor threshold voltage, the leakage energy per operation starts increasing due to largely increasing delay. Below the threshold voltage, the delay increases exponentially with the supply voltage, making the leakage energy increase dramatically and finally balance the decreasing dynamic energy. This forms a minimum energy point in the sub-threshold region. Although the sub-threshold operation is able to achieve the minimum operating energy, it causes significant performance degradation and variation. Therefore near-threshold operation which achieves similar energy saving but with much less performance degradation and variation is more attractive to designers.

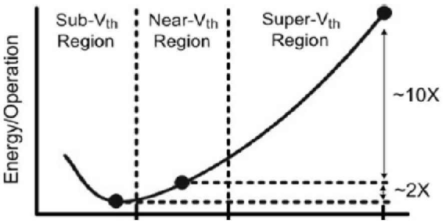

Fig. 1 Sub/Near-threshold operation [1]

However, it is challenging to enable near-threshold operation. When the supply voltage is scaled to the near-threshold region, various issues will arise including functionality, performance and variability issues. To design near-threshold processors, solutions across circuit and architecture levels need to be explored. This paper reviews the existing near-threshold processor design techniques for power-constrained computing devices by addressing the above mentioned issues.

II. Design Techniques for Addressing Functionality Issue

In the past, many work have shown that most of building cells/blocks (e.g. logic gates, ALU blocks) for digital circuits are able to function in the near-threshold region except level shifters and SRAMs. To ensure functionality for near-threshold designs, these cells/blocks need to be re-designed.

A. Level Shifter Design

Level shifters are required to bridge the power domains with different supply voltages (e.g. core to core, core to I/O) in processors. A multi-core processor may contain hundreds to thousands of level shifters as the number of core and data width increases. The conventional level shifters are not able to convert near-threshold voltages directly to super-threshold voltages as the voltage gap is too large. The pull-down transistors controlled by near-threshold voltage cannot overcome the strength of the pull-up transistors controlled by super-threshold voltage, so the output fails to flip when input changes. To address this issue, several techniques have been proposed. In [2][3][4], the pull-up transistors are weakened by using reduced swing inverter or diode to ease the wide-range voltage conversion. While the functionality is restored, the performance is not scalable with supply voltage as the weakening is constant. For [4], the multi-stage design causes

978-1-5090-6625-4/17/$31.00 ©2017 IEEE

larger delay than single-stage design. In [5][6][7], current mirror is employed to reduce the contention between pull-up and pull-down network, and output feedback is used to adaptively control the static current of current mirror, which gives small delay and high energy efficiency with voltage scalability.

B. SRAM Design

Another building block with functionality issue in the near-threshold region is SRAM. It is a crucial component for processor design. While SRAM has similar cell structure as a flip-flop, it is more challenging to ensure SRAM's functionality than flip-flop's in the near-threshold region due to two reasons. One is that in SRAM many cells share the same read/write interface. Also, SRAM pursues high density so small transistors are preferred in SRAM cell. These requirements make it difficult for conventional SRAMs to function in the near-threshold region. For example, small transistors in SRAM cell lead to small strength and large variation at low voltages, which may cause writability or read disturbance issue. The sharing of read/write interface among many cells may cause low I_{on}/I_{off} and thus sensibility issue at low voltages. In [8][9][10], 8T/10T SRAM with decoupled write and read interface is proposed to address the read disturbance and writability issue, and multi-Vt transistors are used to improve the energy efficiency. In [11][12], 9T SRAM and static bitline are proposed to address the I_{on}/I_{off} issue and improve the sense margin. In [8], circuit techniques such as boosted wordline and floating supply are used to improve writability.

III. Design Techniques for Addressing Performance Issue

In addition to functionality issue, another issue of near-threshold design is the performance degradation at ultra-low voltage. This poses a challenge to near-threshold design and limit its applications. To address this issue, co-optimization on both circuit and architecture level is needed.

A. Near-threshold Device Sizing

On the circuit level, it has been found that in the sub/near-threshold region several parasitic effects affect transistor performance more than in the super-threshold region. For example, reverse short channel effect (RSCE) becomes relatively strong in the near-threshold region as the short channel effects such as DIBL decreases with voltage decreasing. RSCE causes the transistor threshold voltage to decrease as transistor length increases. This effect has more impact on transistor performance in sub/near-threshold region than in super-threshold region, as the current is exponentially dependent on voltage in the sub/near-threshold region. It has been utilized to explore optimal transistor length to achieve minimal delay [13]. Another effect is inverse narrow width effect (INWE) which causes the threshold voltage to decrease as the transistor width decreases. This has been utilized in transistor width sizing by adopting minimum finger to reduce the delay and improve energy efficiency [14]. This device sizing method has also been combined with minimum-sizing method and used selectively on critical and non-critical path to further improve the energy-efficiency while meeting the performance requirement [14].

B. Parallel Processing

On architecture level, parallel processing has been adopted to address performance degradation of near-threshold design. This includes homogeneous parallelization and heterogeneous parallelization. In [15], 4 homogeneous parallel processing engines have been designed to accelerate near-threshold JPEG encoding, achieving 40MB/s (or VGA, 30 fps) and 1.33pJ/cycle at 0.6V in a 65nm process. In [16], a near-threshold Centip3De processor with 64 ARM Cotex-M3 cores and 3D stacking (as shown in Fig. 2) has been proposed, achieving 800 DMIPS and 3930 DMIPS/Watt at 0.65V in a130nm process. In [17], a 16x16 mesh NoC with 256 voltage/clock domains is designed. Each of the domains is able to operate from 340mV to 0.9V. At 430mV (near-threshold), the NoC achieves 3.4Tb/s throughput with a peak energy efficiency of 18.3Tb/s/W.

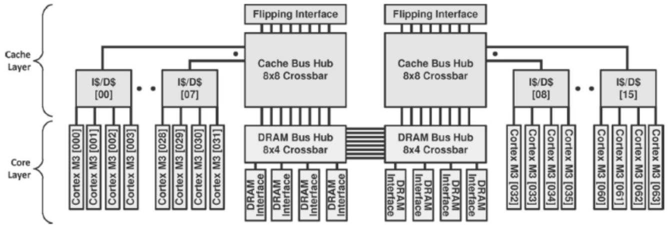

Fig. 2 Centip3De near-threshold processor [16]

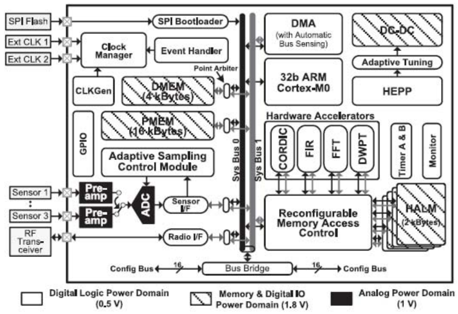

Fig. 3 Near-threshold sensor node processor [19]

In addition to homogeneous parallel processing, heterogeneous parallel processing has also been explored for real-time applications with diverse processing tasks. In [18], multiple heterogeneous processing cores such as CORDIC, FFT are designed to accelerate the processing tasks for a near-threshold biomedical processor, which improves the energy efficiency by more than 10× compared to single programmable processing core. In [19], dual-bus architecture with diverse hardware accelerators such as DWPT and FIR are developed to accelerate the processing tasks for a near-threshold smart sensor node processor (as shown in Fig. 3), achieving 29 pJ/cycle for a vehicle speed estimation application.

C. Ultra-Wide Range DVS

In addition to device sizing and parallel processing, another way to address performance loss at ultra-low voltage is ultra-wide range DVS. For most applications mentioned previously (e.g. IoT, wearable computing), the required performance varies significantly over time. The design should be able to achieve high performance when needed, and save energy when the workload is low. Therefore, ultra-wide range DVS from near/sub-threshold to nominal voltage is needed. In [20], an energy-efficient SIMD vector permutation engine is proposed. This design is able to operate from 280mV to 1.1V and achieve dynamic performance from 16.8MHz to 2.5GHz. To enable fast voltage switching across a ultra-wide voltage range and separate voltage tuning for individual blocks, special consideration is needed for the on-chip voltage generation. Conventional voltage generation schemes using DC-DC converter has limited switching speed (tens to hundreds of μs). Although some latest designs is able to achieve tens of ns, the increased complexity/area make it impractical to dedicate a DC-DC converter for each individual block. In [21], a processor with panoptic (i.e. full range) DVS from sub-threshold to high performance is proposed (Fig. 4). This design uses multiple PMOS header switches at the component level to provide a local supply voltage from a discrete set of global supply voltages. This allow component-level voltage tuning and fast voltage switching.

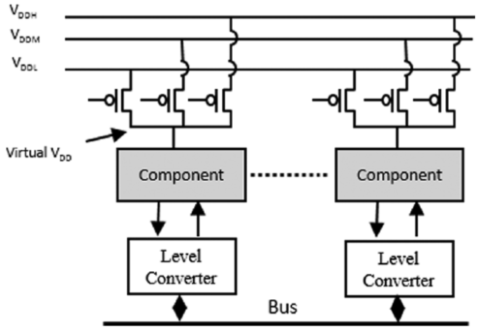

Fig. 4 Panoptic DVS [21]

IV. Design Techniques for Addressing Variability Issue

Another challenge of near-threshold processor design is the large performance variation at ultra-low voltage. In the near-threshold region, the current is exponentially dependent on the voltage, therefore small PVT variation will result in significant variation in delay or performance. Unlike the variation in the super-threshold region, the variation in near-threshold region can reach up to 10×. This leads to extremely large design margin in order to ensure the design works in the worst case or even within 3σ. Circuit and architecture level solutions are needed to address this issue.

A. Library pruning

Many designers perform library pruning to existing standard cell libraries to reduce the performance variation for near-threshold design. The library cells are characterized at ultra-low voltage and the ones with large variation and poor performance are taken out. Usually the logic gates with more than four-stacked transistors are removed due to their large variation at ultra-low voltage [19][22]. For other cells, the designer can directly use them or modify for better performance. For example, previously mentioned device sizing methods can be used to reduce the delay and improve energy efficiency.

B. Setup/Hold Timing Fixing

Digital processors have to meet setup and hold timing in order to function correctly. For near-threshold processors, however, in order to guarantee the timing, significant design margin is needed as mentioned previously. To address this issue, timing monitoring has been used to capture the variation on individual chip and adaptive voltage/clock tuning is used to fix the timing. This reduces the design margin while guaranteeing the timing. For example, Razor uses a shadow latch to capture the late arrival events in the critical paths and correct the errors via processor architectural reply or pipeline stalling. Canary flip-flop uses artificially delayed data path to predict potential setup timing error with voltage scaling. The drawback of Razor is the large overhead while the issue of canary flip-flop is that it cannot detect sudden errors. HEPP technique combines the detection and prediction technique to address the issue.

In addition to setup timing, hold timing fixing for near-threshold processor has also been investigated. Unlike setup timing error which can be detected using clock edge, it is difficult to find timing reference to detect hold timing error. Instead of detecting hold timing error, in [23], the hold timing is fixed by raising the voltage of clock tree in a near-threshold processor. As shown in Fig. 5, the number of hold buffer stages required decreases by 5× when the clock tree voltages is increased by 0.1V at 0.5V.

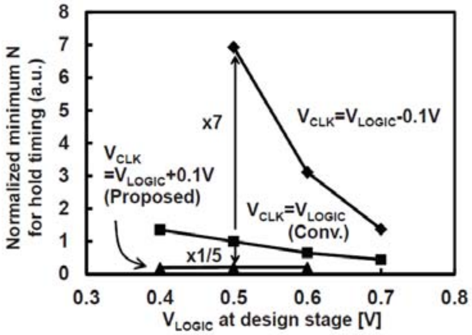

Fig. 5 Required number of hold buffer stages at different clock tree voltage [23]

C. Processor Pipeline Optimization

For near-threshold processor, another technique to reduce processor variation is to optimize the processor pipeline. It is found that large logic depth is able to reduce the random process variations through averaging effect [24], therefore shallow pipeline with large logic depth is adopted. As shown in Fig. 6, a two-stage pipeline with 75 logic gates per pipeline stage reduces the standard deviation by 19%, compared with a design with 10 logic gates per pipeline stage. In [25], decoupled SIMD parallel pipelines are used to allow timing error detection and correction for each pipeline independently, so that all the parallel pipelines do not have to stall or flush together. This help improve the average throughput. Also, pipeline weaving technique is used to mitigate the impact of static variation on the yield and performance of SIMD processor by duplicating the share components between parallel pipelines and connecting them with subsequent stages in a 'weave' fashion. Components that failed testing can be disabled and reconfigured during configuration time.

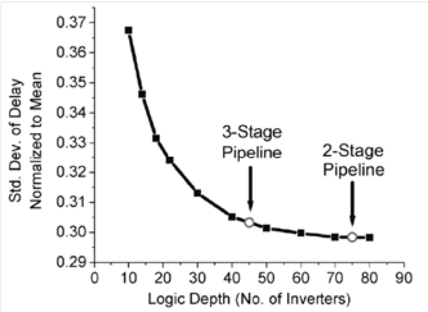

Fig. 6 Random variation vs number of pipeline stages [25]

V. Conclusions

This paper reviews the existing near-threshold processor design techniques for power-constrained computing devices, targeting at IoT, intelligent sensing and wearable computing applications. Both circuit and architecture level design techniques for addressing the major challenges of near-threshold processor design (i.e. functionality issue, performance degradation and variability issue) have been discussed in the paper.

References

[1] R. G. Dreslinski, et al., "Near-Threshold Computing: Reclaiming Moore's Law Through Energy Efficient Integrated Circuits," in Proceedings of the IEEE, vol. 98, no. 2, pp. 253-266, Feb. 2010.

[2] Y. S. Lin, et al., "Single stage static level shifter designfor subthreshold to I/O voltage conversion," in Proc. ACM/IEEE Int. Symp. Low Power Electron. Design (ISLPED), Aug. 11 – 13, 2008, pp.197 – 200.

[3] I. J. Chang, et al., "Robust level converterfor subthreshold/super-threshold operation: 100 mV to 2.5 V," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 19, no. 8, pp.1429 – 1437, Aug. 2011.

[4] S. N. Wooters, et al., "An energy-efficientsubthreshold level converter in 130-nm CMOS," IEEE Trans. CircuitsSyst. II, Exp. Briefs, vol. 57, no. 4, pp. 290 – 294, Apr. 2010.

[5] S. Lutkemeier, et al., "A subthreshold to above-threshold level shifter comprising a wilson current mirror," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 57, no. 9, pp. 721 – 724, Sep. 2010.

[6] Y. Osaki, et al., "A low-power level shifter with logic error correction for extremely low-voltage digital CMOS LSIs," IEEE J. Solid-State Circuits, vol. 47, no. 7, pp.1776 – 1783, Jul. 2012.

[7] J. Zhou, et al., "An Ultra-Low Voltage Level Shifter Using Revised Wilson Current Mirror for Fast and Energy-Efficient Wide-Range Voltage Conversion from Sub-Threshold to I/O Voltage," Circuits and Systems I: Regular Papers, IEEE Transactions on , vol.62, no.3, pp.697,706, March 2015

[8] B.H. Calhoun, et al., "A 256-kb 65-nm Sub-threshold SRAM Design for Ultra-Low-Voltage Operation," Solid-State Circuits, IEEE Journal of , vol.42, no.3, pp.680,688, March 2007

[9] T. Kim et al., "A voltage scalable 0.26 V, 64 kb 8T SRAM with lowering techniques and deep sleep mode," IEEE J. Solid-State Circuits, vol. 44, no. 6, pp. 1785 – 1795, Jun. 2009.

[10] B. Wang, et al., "Maximization of SRAM Energy Efficiency Utilizing MTCMOS Technology," Asia Symposium on Quality Electronic Design (ASQED), pp. 35-40, July 2012

[11] Q. Li,et al., "A 5.61 pJ, 16 kb 9T SRAM with Single-ended Equalized Bitlines and Fast Local Write-back for Cell Stability Improvement," IEEE European Solid-State Device Research Conference (ESSDERC), pp. 201-204, Sept. 2012

[12] A. Do, et al., "A 32kb 9T SRAM with PVT-tracking Read Margin Enhancement for Ultra-low Voltage Operation," IEEE International Symposium on Circuits and Systems (ISCAS), May pp. 2553-2556, 2015

[13] T. Kim, et al., "Utilizing Reverse Short Channel Effect for Optimal Subthreshold Circuit Design," Low Power Electronics and Design, 2006. ISLPED'06. Proceedings of the 2006 International Symposium on , vol., no., pp.127,130, 4-6 Oct. 2006

[14] J. Zhou, et al., "A 40 nm Dual-Width Standard Cell Library for Near/Sub-Threshold Operation," IEEE Transactions on Circuits and Systems I: Regular Papers, vol.59, no.11, pp.2569,2577, Nov. 2012.

[15] Y. Pu, J. Pineda de Gyvez, H. Corporaal and Y. Ha, "An Ultra-Low-Energy Multi-Standard JPEG Co-Processor in 65 nm CMOS With Sub/Near Threshold Supply Voltage," IEEE Journal of Solid-State Circuits, vol. 45, no. 3, pp. 668-680, March 2010

[16] D. Fick et al., "Centip3De: A Cluster-Based NTC Architecture With 64 ARM Cortex-M3 Cores in 3D Stacked 130 nm CMOS," in IEEE Journal of Solid-State Circuits, vol. 48, no. 1, pp. 104-117, Jan. 2013.

[17] G. Chen et al., "16.1 A 340mV-to-0.9V 20.2Tb/s source-synchronous hybrid packet/circuit-switched 16×16 network-on-chip in 22nm tri-gate CMOS," 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), San Francisco, CA, 2014, pp. 276-277.

[18] J. Kwong and A. P. Chandrakasan, "An Energy-Efficient Biomedical Signal Processing Platform," in IEEE Journal of Solid-State Circuits, vol. 46, no. 7, pp. 1742-1753, July 2011.

[19] X. Liu et al., "An Ultralow-Voltage Sensor Node Processor With Diverse Hardware Acceleration and Cognitive Sampling for Intelligent Sensing," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 62, no. 12, pp. 1149-1153, Dec. 2015.

[20] S. Hsu et al., "A 280mV-to-1.1V 256b reconfigurable SIMD vector permutation engine with 2-dimensional shuffle in 22nm CMOS," 2012 IEEE International Solid-State Circuits Conference, San Francisco, CA, 2012, pp. 178-180.

[21] K. Craig et al., "A 32 b 90 nm Processor Implementing Panoptic DVS Achieving Energy Efficient Operation From Sub-Threshold to High Performance," in IEEE Journal of Solid-State Circuits, vol. 49, no. 2, pp. 545-552, Feb. 2014.

[22] Y. Kim et al., "A 0.5 V 54 μW Ultra-Low-Power Object Matching Processor for Micro Air Vehicle Navigation," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 63, no. 3, pp. 359-369, March 2016.

[23] M. Nomura et al., "0.5V image processor with 563 GOPS/W SIMD and 32bit CPU using high voltage clock distribution (HVCD) and adaptive frequency scaling (AFS) with 40nm CMOS," VLSI Circuits (VLSIC), 2013 Symposium on, pp. C36-C37, 2013

[24] B. Zhai et al., "Energy-Efficient Subthreshold Processor Design," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 17, no. 8, pp. 1127-1137, Aug. 2009.

[25] E. Krimer, R. Pawlowski, M. Erez and P. Chiang, "Synctium: a Near-Threshold Stream Processor for Energy-Constrained Parallel Applications," in IEEE Computer Architecture Letters, vol. 9, no. 1, pp. 21-24, Jan. 2010.