Design and Analysis of the HF-RISC Processor Targeting Voltage Scaling Applications

Felipe Todeschini Bortolon*, Sergio Johann Filho†, Matheus Gibiluka†,
Sergio Bampi*, Ney Laert Vilar Calazans†, Fabiano Passuelo Hessel†, Matheus Trevisan Moreira†
* Universidade Federal do Rio Grande do Sul – Porto Alegre – RS – Brazil
† Pontifícia Universidade Católica do Rio Grande do Sul – Porto Alegre – RS – Brazil
{ftbortolon, bampi}@inf.ufrgs.br
{sergio.filho, ney.calazans, fabiano.hessel}@pucrs.br {matheus.gibiluka, matheus.moreira}@acad.pucrs.br

***Abstract*—This paper presents the design and analysis of HF-RISC, a 32-bit RISC processor, targeting voltage scaling applications. We start proposing a design flow that enables the processor to operate at multiple voltage levels and explore how this flow enables designers to leverage the advantages of low voltage designs. Next, we present a set of case study designs of HF-RISC in a 28nm FD-SOI technology assessing their area, performance and power figures. Using the collected data we discuss how our flow can enable better design space exploration for voltage scaling applications and define guidelines for achieving lower power and better power efficiency. Accordingly, the obtained results indicate that the proposed flow allows 9.5% lower power overall and 25.5% better energy efficiency in HF-RISC design.**

***Keywords*—Voltage Scaling; Low Voltage; Energy Efficiency; MIPS; FD-SOI; Design Space Exploration; Embedded systems;**

I. Introduction

Internet-of-Things (IoT) applications typically require relatively high processing power combined to low energy consumption. In such applications, several architectural characteristics have to be evaluated and processors must be designed for specific operating conditions and scenarios. Due to the the need for integration of IoT devices with other computer systems through network protocols such as IPv6 [1], 32-bit processor cores provide better performance and energy trade-offs, compared to 8 or 16-bit devices [2]. In RISC designs it is advantageous to keep the number of pipeline stages low to reduce the penalties originated from hazards. A simpler pipeline improves the number of executed instructions per cycle (IPC), simplifies the design [3] and reduces energy consumption [4].

With this shallow pipeline strategy, HF-RISC, a 32-bit RISC processor was designed, as presented in [5]. HF-RISC greatly simplifies the instruction set architecture (ISA) implementation, as no interlocks or forwarding units are needed to fix hazards. This approach is useful for low power design, where applications aim at better energy/MHz ratios, rather than high clock frequencies at a high energy cost. The industry currently employs the same principle, using 32-bit processors like the ARM Cortex-M family [6] of processors with only 2 or 3 pipeline stages [2], in place of 8- and 16-bit microcontrollers. In this way, the design choices in HF-RISC aim to improve both performance and energy efficiency as detailed in [5].

At the circuit design level, an interesting approach to decrease energy consumption is to reduce the supply voltage while the device does not need to deliver symbolic performance. This is commonly known as supply voltage scaling or simply voltage scaling (VS). This technique is very effective because the power dissipation in an integrated circuit is quadratically proportional to the supply voltage [7]. Therefore, small voltage reductions lead to significant energy savings. For this reason several publications explore the design of VS systems, however a thorough discussion about the complete design flow is still missing. More specifically, there is no guidelines available considering how Static Timing Analysis (STA) tools can assist the design of a VS system, even though they are largely employed for circuit optimization.

This paper aims to fulfill this gap by analyzing the trade-offs offered by the synthesis tool when using a cell library characterized from sub to super-threshold. After a review of the state-of-art in Section II-A to demonstrate the novelty of this investigation, the target processor for the case-study, the HF-RISC, is presented in Section II-B . Next, Section III explains the synthesis flow and how it is used to implement several case studies. Section IV presents the results and delves into the details of the trade-offs that might be achieved using this flow. Accordingly, Section V uses the obtained results to present the final conclusions and to define guidelines for better exploring the synthesis flow using STA tools.

II. State-of-the-Art and Background

A. Processors for Voltage Scaling Applications

According to [8], VS leverages the computational burstiness of processors used in portable electronics devices, where typically only a fraction of the computation utilizes its full performance. Several works available in contemporary literature explore this profile to demonstrate the benefits of VS in integrated circuits. From a mathematical perspective, different authors analyzed at distinct angles how to stress VS and achieve its limits. In [9], for example, Yuan and Qu define three different types, called models, of VS systems based on how voltage can be scaled. Their purpose is to define under different constraints, e.g. voltage scaling range and delay, guidelines for the designer of how much energy savings might be achieved. Their analysis provides solid equations and simulation results to answer this question; however their results do not advance to the sub-threshold region.

This topic is of particular importance as stated in [10], where authors discover that the minimum operating voltage point for maximum energy-efficiency lies on that region. Although the optimal point depends on the workload characteristics of a specific processor, they determine that under typical load, operating voltage should scale to approximately

978-1-5090-2736-1/16/$31.00 ©2016 IEEE

30% (V_{min}) of its maximum value. A more recent work [11] shows an interesting investigation of VS benefits for future technologies using LEON3 processor as their case-study. Their analysis compare two different FinFET nodes against two traditional CMOS bulk nodes and conclude that the latter takes more advantage of scaling over the first due to their layout differences.

While some authors provide a theoretical perspective of this technique, there are other contemporary reports that demonstrate experimental applications of VS [12]. Accordingly, the work published by Craig et al. in [13] presents a 32-bits, 90 nm data flow processor that employs dynamic voltage scaling from sub-threshold to high performance. The design can scale at three different V_{DD} settings (0.7V, 0.8V and 1.2V) which can switch in a single clock cycle through selectively controlling a set of PMOS transistors. Each transistor has its source terminal connected to different V_{DD}s; therefore if it is in the on state, it lets the selected V_{DD} voltage to supply the data path. The authors present an interesting discussion about the special considerations taken for the sub-threshold PMOS switch design and the architectural modifications to allow blocks to operate at different voltages, i.e. memory on 1.2V and data path on 0.7V. Nevertheless, the synthesis process, e.g. cell pruning and selection, is not explored to optimize the design.

Authors in [14] fabricated a 65 nm, 32-bits sub-threshold RISC processor using a custom standard cell library which was designed to optimize noise margins, switching energy and propagation delay simultaneously. For comparison, an additional core based on conventional standard cells is added to the design. The chip implementation uses a hierarchical multi-mode multi-corner (MMMC) synthesis to account for different timing conditions under varying supply voltage. Even though conventional standard cell core instance achieves 260MHz at 1.2V it fails to operate bellow 700mV, while the custom can operate over a supply voltage range from 200mV to 1.2V with clock frequencies from 10 kHz to 94 MHz for the best samples. Another work that explores custom standard cell library is presented in [15] for a 32nm, 32-bits processor that operates from 280mV to 1.2V with clock frequency from 3MHz to 915MHz. To optimize their design for robust and reliable ultra-low voltage operation, they prune the cell library to eliminate circuits which exhibit DC failures or extreme delay degradation due to reduced transistor on/off current ratios and increased sensitivity to process variation. In contrast to [14], they argue that in the absence of MMMC tools that it is important to identify the optimal design such that the results achieved in one corner do not compromise operation at other corners. Therefore, their resulting library is characterized for three different supply voltages: 0.5V, 0.75V and 1.05V and it was identified that using 0.5V yield the best trade-offs for their case.

This increasing concern to achieve better performance and energy trade-offs with VS is crucial to new power-aware embedded systems, since stricter power dissipation constraints are imposed to devices that have higher transistor integration density. The works analyzed here explore different techniques to that extent. However only [14] and [15] delve into the synthesis process. Regarding library pruning, both of them confirm that considering noise margins and delay as cell selecting criterion provides more robust and reliable circuits. Nonetheless, [14] uses MMMC while [15] uses standard cell library characterization on different voltages. Even though, MMMC guarantees that constraints for all voltages are meet, this approach does not actually ensure that the tool will explore design trade-offs for performance and energy. On the other hand, [15] motivates this exploration by pointing to the importance of standard cell library characterization for different voltages to achieve those better trade-offs, however their approach is restricted to only three corners.

In conclusion, none provided a thorough exploration to determine how to deal with different voltage levels in synthesis flows or how to guide synthesis tools to achieve the best trade-offs between sub and super-threshold voltage operation, i.e. the synthesis optimal point. Because the majority of designs rely on STA tools to synthesize and optimize integrated circuits (ICs), this type of analysis can provide better insight to allow the designer to better explore VS design space. Therefore, this work stands off by analyzing the different performance, power and area outcomes from the synthesis tool with standard cell libraries characterization for different supply voltages considered in the design phase.

B. The HF-RISC Processor

The HF-RISC Instruction Set Architecture (ISA) comprises a small subset of the MIPS I ISA introduced in [16] targeting compatibility with existing tools and optimizing compilers. Additionally, its core has specific features for low-power design such as fewer pipeline stages and a more compact organization of its components. The most relevant differences between this implementation and the classic 5-stage MIPS are:

- Short, 3-stage pipeline, to simplify core design and to reduce chip area and energy consumption;
- No hazard or forward units, due to the short pipeline;
- Shared instruction and data memories, i.e. a Von Neumann organization.
- Fully synchronous, single clock edge design: registered memories are interfaced directly;
- 3-cycle branch delay when taken, with 2 branch delay slots;
- No unaligned loads/stores; no MMU; no exceptions;
- No co-processor; only memory-mapped peripherals (EPC, MASK, STATUS, VECTOR, and CAUSE registers);
- Configurable HW multiply unit; no HW division unit;
- A set of MCU-like peripherals: an optional UART, an interrupt controller, a running counter, two programmable counters, compare registers and a debug interface.

Most HF-RISC instructions take just one clock cycle, but load and store instructions take three cycles each, due to the memory bus multiplexing and pipeline refill. Also, multiply instructions take several cycles, depending on the chosen hardware configuration - 4 cycles for a parallel multiplier and from 11 to 35 for a serial multiplier. A side effect of

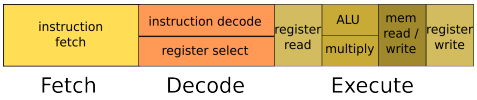


Fig. 1: The HF-RISC 3-stage pipeline and the stage tasks [5].

the simple pipeline is the absence of explicit load delay slots of conventional MIPS organizations. Two branch delay slots arise due to pipeline design, as the outcome of branches is discovered on the third pipeline stage. This simplifies the datapath, as the same ALU is used for both arithmetic and branch target calculation. Other advantages are the lack of a hazard detection unit and forwarding paths. The compiler can schedule instructions in the first branch slot, reducing branch penalty to 2 cycles. Although there are several different configuration possibilities for the processor, we chose one which offers the best performance/power ratio [5]. In this paper, we have conducted our experiments with the serial multiplier version, as it offers good performance (1.61 Coremark/MHz) and low area overhead.

Fig. 1 depicts the stages of the HF-RISC pipeline and the tasks executed in each of these. In the *fetch* stage, memory is accessed and an instruction becomes available in one cycle. In this same cycle the PC is updated. In the *decode* stage an instruction is fed into the decoding and control logic, so values are registered for the next stage. Pipeline bubble insertion is performed in this stage for memory and branch operations. In the *execute* stage the register file is accessed and the ALU calculates the result of the operation. Address and data are put on the data bus (on store operations) or data are copied to the register file (on load operations). On logic/arithmetic operations, the ALU result is written to the register file. Branch outcomes are computed in this stage. Multiply operations write the result to HI and LO registers. The register file is accessed only in the *execute* stage in order to simplify the design, as it allows avoiding the addition of extra hardware to cope with data hazards. To evaluate the performance of the architecture and build an adequate programming environment for measurements, we created a hardware abstraction layer (HAL), small C and runtime libraries, and used the GNU tools based on GCC 4.9.3 and Binutils 2.24. The compiler backend was modified to support all different processor configurations, including the absence of multiply and divide instructions and other microarchitecture features.

III. Voltage Scaling Aware Design and Analysis Flow

Electronic Design Automation (EDA) tools usually rely on Register-Transfer Level (RTL) descriptions to capture an IC specification, and on cell libraries models to synthesize and map this specification to a specific technology. During this process, the design goes through optimization steps that are driven by STA tools, which provide delay and power estimations for the synthesized circuit. These results are calculated using data available on the cell library models that describe cell's logic behavior and electrical characteristics for a set of operating conditions, e.g. temperature and voltage. In this way, an efficient synthesis and analysis of a circuit for VS applications requires cell libraries characterized for a specific set of voltage levels. However, these models are not conventional or easy to find, which impairs the capability of a

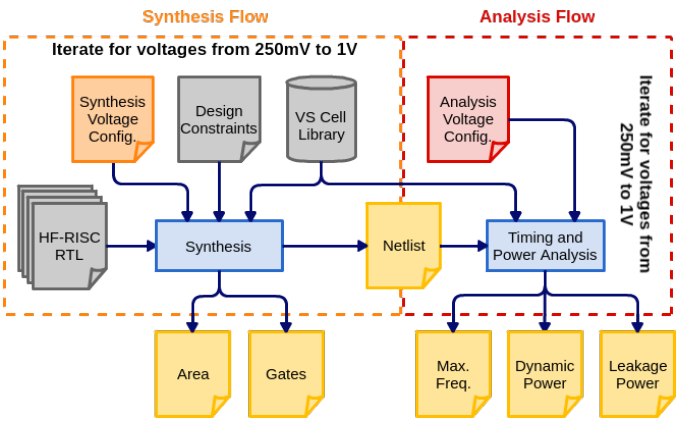


Fig. 2: Voltage scaling aware design and analysis flow.

designer to optimize a circuit for a given voltage level, or for a range. Hence, to explore the design space for VS applications this work characterized and pruned a library of cells targeting the STMicroelectronics 28nm FD-SOI technology, which will be referenced as VS Cell Library. Note that the selected node is FD-SOI because it allows operation in a wide range of voltages, suitable for VS applications, and its migration from bulk is relatively straightforward [17]. In summary, first the available library provided for nominal V_{DD} is reduced to a subset of cells that have a maximum of 3 inputs as advised by Kwong et al. in [18]. Then, the selected cells to compose the VS library were characterized using Cadence ELC^{TM} for voltages from 250 mV to 1 V in steps of 50 mV. For a complete description of this process refer to [19], [20].

With the availability of these different models, it was devised a voltage scaling aware design and analysis flow, as showed in Fig. 2, to evaluate the HF-RISC for different voltage levels. As the figure shows, the flow was divided in two main steps: synthesis and analysis. The synthesis has as its inputs the RTL description, a set of design constraints, the VS cell library and the voltage configuration. Note that the design constraints were defined to allow a realistic synthesis scenario, where the outputs of the design had realistic intra-chip capacitance values and the inputs had realistic transition and insertion delays. Such figures are all equivalent to those of a slice register, commonly employed in microprocessor SoCs. This step was iterated 16 times, sweeping the defined voltage configuration from 250 mV to 1 V in steps of 50 mV. At each iteration the HF-RISC is synthesized to its maximum frequency generating a netlist that is optimized for maximum performance at each voltage. This process is undertake for every library model previously generated, thus all 16 library models are combined to the 16 different operating voltages. To achieve maximum frequency it is necessary to perform a binary search in the defined clock and resynthesize the circuit multiple times. By adopting this approach is possible to stress the STA tool and explore its capability to optimize circuits for different voltages levels.

As Fig. 2 shows, after synthesis, each of the generated netlists are considered in the analysis step. Additionally to synthesis output, this step generates maximum operating frequency, dynamic power and leakage power reports. Because the flow can be configured for different voltage levels, it is possible to analyze how each synthesized version of HF-

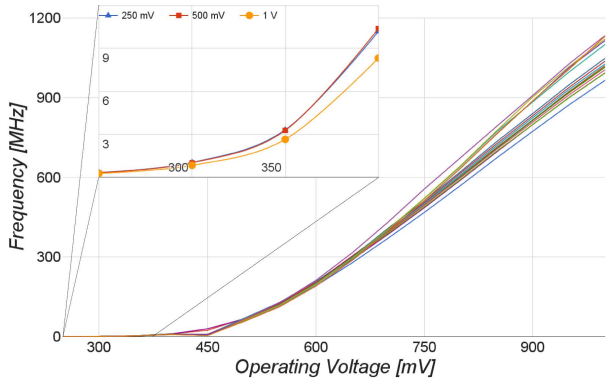


Fig. 3: Frequency vs. Operating Voltage for different Synthesis Voltage

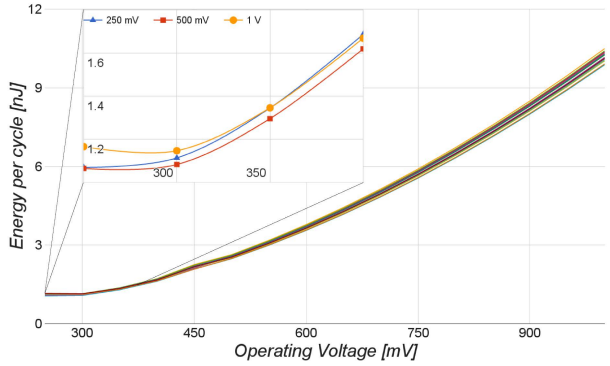
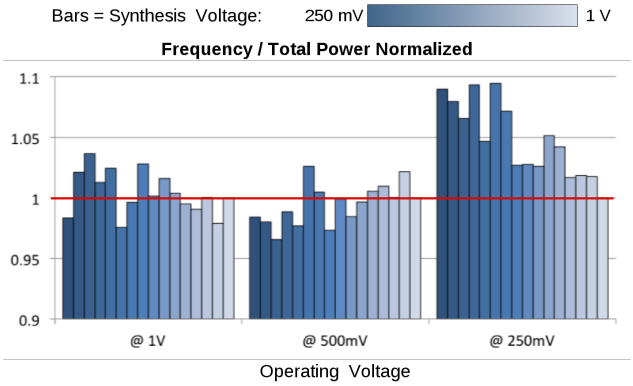


Fig. 4: Energy / Cycle vs. Operating Voltage for different Synthesis Voltage

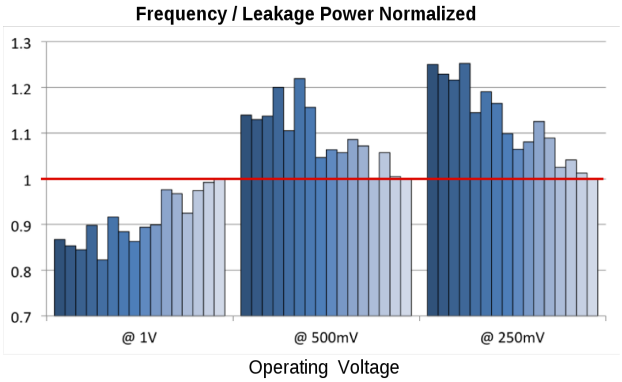
RISC behaves at each voltage. Furthermore, results help to understand how frequency and power metrics scale for each synthesized netlist with the variation in the voltage configuration. Maximum frequency was measured by analyzing the maximum clock period that allowed non-negative slack in the design as the voltage was scaled in the analysis step. Power figures were measured by simulating the CoreMark benchmark in the synthesized circuit, exporting the internal activity of the nets and performing static power analysis. After all iterations of the complete flows, the output collected a total of 16 designs (one for each voltage level) and the respective metrics of each for 16 distinct voltage levels.

IV. Results Analysis

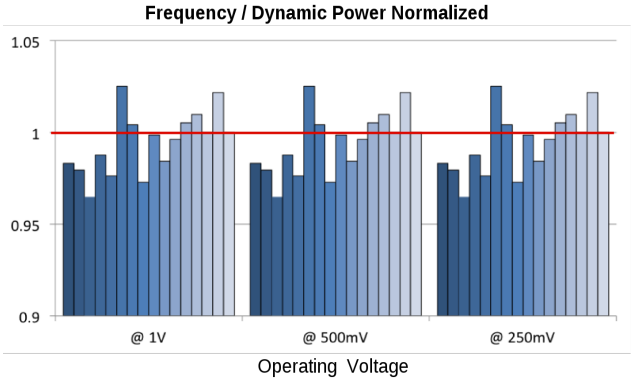
The first step to understand the trade-offs between the different generated netlists is to evaluate how their maximum frequency scales with voltage. This analysis, which is depicted in Fig. 3, shows how the frequency of a circuit synthesized at 250 mV varies as we sweep its operating voltage and how the same metric varies for the circuit synthesized at all other voltages. Therefore, the designer will have better insights on what is the best voltage to characterize a library for VS applications. Interestingly, the depicted results for the 250 mV synthesis exhibit an average of 24% higher frequency than 1 V synthesis while operating on the sub-threshold region, and a decrease of 11% on the super-threshold. An even better trade-off is obtained at the 500 mV synthesis, where the sub and super-threshold frequencies increase and decrease, respectively, by 22% and 5% compared against 1V synthesis.



(a)



(b)



(c)

Fig. 5: Frequency over (a) Total Power, (b) Leakage Power and (c) Dynamic Power, all normalized to the 1V synthesis. In each is depicted three different operating voltages: @ 1V, @ 500mV and @ 250mV, and synthesis voltages varies from 250mV to 1V (left to right) for each.

Extending this investigation to power consumption, the results depicted in Fig. 4 demonstrate that exploring voltage during synthesis can yield extra energy savings at the sub-threshold region. For example, as the charts show, the netlists synthesized at 250 mV and 500 mV present a better minimum energy point when compared to the netlist synthesized at 1 V. Note that these 3 voltage levels were isolated because they exhibit superior behavior for the 3 operating modes: sub-threshold, near-threshold and nominal. Diving more deeply into this scenario, Fig. 5 details this analysis for dynamic and leakage power and normalizes the results to the netlist synthesized at 1 V to determine the energy gains for each

different synthesis voltage. Fig. 5 (a) demonstrates that the netlist synthesized at 250 mV allows a 9% improvement on energy efficiency for an operating voltage of 250 mV, while presenting less than 2% degradation at both 500 mV and 1 V supply voltages. The 500 mV synthesis once again exhibits better improvements than 250 mV, with an increase of 9.5% and 2.5%, and a decrease of 2.5% at respectively 250 mV, 500 mV and 1 V. Interestingly, those energy savings derive from the leakage power improvements (Fig. 5 (b)) that significantly supersedes dynamic power (Fig. 5 (c)) deterioration. Since the off-current does not reduces as strongly as the on-current when the voltage scales below the threshold [10] optimizing leakage results in the depicted energy saving enhancement.

To further scrutinize energy optimization opportunities on the sub-threshold region, Fig. 6 plots the power efficiency measured as maximum coremarks divided by total power for three different synthesis: 250 mV, 500 mV and 1 V. As explored in [5], the maximum coremarks figure represents the processor efficiency in terms of its frequency since it correlates architectural and technology metrics of a design. Because all case studies share a common architecture, this analysis permits to explore the impact of technology specific figures on the core performance, in this case the voltage level used during synthesis. Combining this metric with total power, it is possible to understand the impact of such choice in a processor's design, more specifically, its effects on overall power efficiency. Results in Fig. 6 demonstrate that lower synthesis voltage achieves interesting improvements. Here the gains provided by 500 mV synthesis over 250 mV are more noticeable, which confirms the same results derived from previous analysis.

Another interesting metric is the relationship between design area and performance through synthesis variation. To that extent, Fig. 7 shows the Maximum Coremark achieved over design area (μm^2) for three supply voltages and four different synthesis normalized to the 1 V synthesis. The collected results are summarized in Table I. At each operating voltage the maximum performance is achieved at their respective synthesis voltages, i.e. 750 mV synthesis has maximum gain at 750 mV supply voltage. In the other cases the graph exhibits different improvements considering how close is the synthesis voltage to the supply voltage, i.e. staircase behavior at 1 V and 250 mV supply voltages. Nonetheless, similar to the previous graphs, overheads at super-threshold region for 500 mV and 250 mV are not so pronounced as the advantages that they offer for the sub-threshold. Therefore, the designer should carefully consider synthesizing at smaller voltages if the circuit targets voltage scaling.

The benefits demonstrated thus far rely on the synthesis tool capabilities to select gates that are more suited to operate at different voltage levels. Cell libraries characterized at lower voltage have precise information of the high delays that gates present when operating at lower voltage levels, thus allowing the tool to chose higher strength gates to minimize overall timing, i.e. maximize frequency. Fig. 8 supports this statement demonstrating the difference on cell strength selection for three distinct netlists using the 1 V as their reference. In the 500 mV synthesis, for example, 34.65% cells with strength higher than 30x is added comparing to the number of similar cells on the 1 V synthesis. Therefore, it is possible to conclude that in order

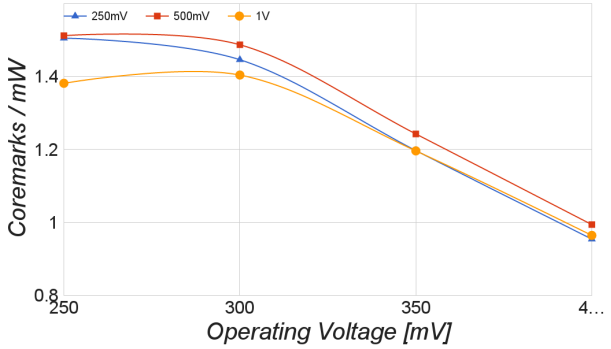


Fig. 6: Maximum Coremarks over Total Power vs. Operating Voltage for three different synthesis voltages: 250mV, 500mV and 1V

TABLE I: Maximum Coremark over Area performance comparison against 1V synthesis.

	Synthesis Voltage		
Supply Voltage	**@250mV**	**@500mV**	**@750mV**
@1V	↓ 13.2%	↓ 8.5%	↓ 1.3%
@750mV	↓ 7.2%	↓ 1.3%	↑ 6.9%
@500mV	↑ 13.3%	↑ 21.2%	↑ 8.7%
@250mV	↑ 25.5%	↑ 20.2%	↑ 12.5%

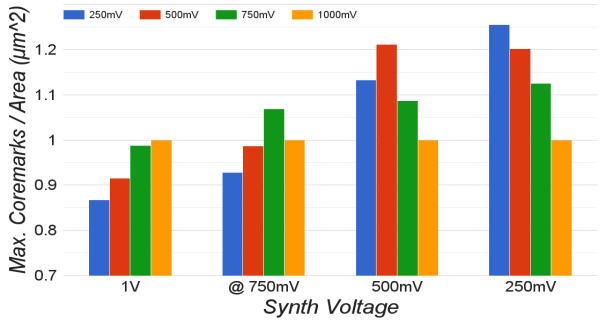


Fig. 7: Maximum Coremarks over Area vs. Synthesis for three different synthesis voltages: 250mV, 500mV and 1V

to meet the timing constraints the tool uses cells with higher strengths. This analysis is in agreement with previous works such as [21], which demonstrates that optimizing transistor sizing for the minimum energy operating point gives symbolic performance and power improvements. Moreover, presented results demonstrate the importance of having cell libraries characterized at different voltage levels for VS aware design and the capability of synthesis tools to leverage this data.

V. Discussion and Conclusions

Embedded designs often have to trade performance improvements for overall power and area. Thus, performance, power and area ratios, are more relevant than the traditional area, speed and power measurements isolated. To that extent, this work proposed a design flow that explores HF-RISC processor, i.e. target circuit, outcomes from synthesis using libraries characterized at distinct voltage levels. Apart from Jain et al. [15] and Lutkemeier et al. [14], which do not delve into the same details, the state-of-the-art does not explore the synthesis process to balance defined metrics. Results obtained

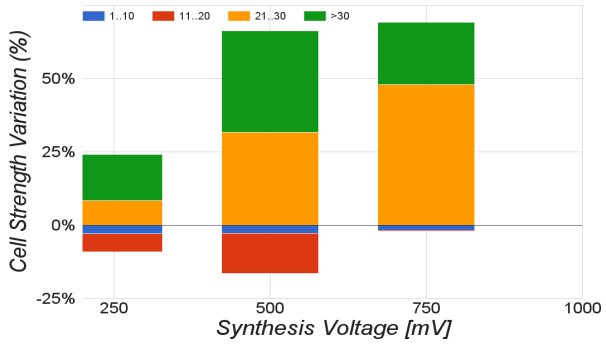


Fig. 8: Cell strength variation for 250mV, 500mV and 750mV synthesis compared to 1V.

from our VS aware design flow indicate that it is possible to achieve 20% higher clock frequencies in the sub-threshold with an overhead of only 5% at nominal voltage 1 V. Additionally, it is interesting to note that waiving 5% of performance at nominal supply voltage can offer further 9.5% and 2% more energy savings at the minimum and maximum voltage values, respectively. As depicted in Fig. 5, the dominant savings on energy derives from the leakage current. Considering that the target technology of this work, i.e. 28 nm FD-SOI, already offers reduced device off-current, we consider a fair assumption that traditional CMOS bulk technologies could achieve higher order of magnitude savings. Therefore, future work may explore other technology nodes or stress energy savings through body-biasing, technique of which is very suitable for FD-SOI.

To conclude, the results obtained in our analysis indicate that for synthesizing our processor for VS applications, extra cell library models are required in order to better explore the design space. Specifically for the HF-RISC processor, the 500 mV synthesis achieves the best trade-off between presented metrics, slightly superseding 250 mV. Note that the latter offers, in some cases, better improvements at a higher degradation cost, however. Examining Fig. 8 the probable reason for this outcome is the higher strength cells used for 500 mV, which ensure superior behavior at near and sub-threshold region. Altogether, this work demonstrates that proposed VS design flow can significantly enhance overall design efficiency regarding power and frequency. Moreover, results offer important insights for the IC designer on the behavior of the synthesis tool when considering different cell libraries. For example, exploring two extra voltage level models, one for sub- and another for near-threshold, can provide improvements for VS applications, without the overhead implied from finer grain voltage steps library characterization. This is an important observation, because this process is an extensive and laborious task.

REFERENCES

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions," *Fut. Gen. Comp. Syst.*, vol. 29, no. 7, pp. 1645–1660, Sep. 2013.

[2] J. Yiu, *The Definitive Guide to the ARM Cortex-M0*. Newnes, 2011.

[3] M. Labrecque, P. Yiannacouras, and J. G. Steffan, "Custom Code Generation for Soft Processors," *SIGARCH Comput. Archit. News*, vol. 35, no. 3, pp. 9–19, Jun. 2007.

[4] O. Azizi, A. Mahesri, B. Lee, S. Patel, and M. Horowitz, "Energy-performance Tradeoffs in Processor Architecture and Circuit Design: A Marginal Cost Analysis," in *37th Annual International Symposium on Computer Architecture*, 2010, pp. 26–36.

[5] S. J. Filho, M. T. Moreira, N. L. V. Calazans, and F. P. Hessel, "The HF-RISC Processor: Performance Assessment," in *LASCAS*, 2016.

[6] ARM, "Cortex-m0+ technical reference manual," Tech. Rep., 2012. [Online]. Available: http://infocenter.arm.com/help/topic/com.arm.doc.ddi0484b/DDI0484B_cortex_m0p_r0p0_trm.pdf

[7] J. Kwong and A. Chandrakasan, "Variation-driven device sizing for minimum energy sub-threshold circuits," in *International Symposium on Low Power Electronics and Design*, 2006, pp. 8–13.

[8] T. D. Burd and R. W. Brodersen, "Design issues for dynamic voltage scaling," in *International Symposium on Lower Power Electronics and Design (ISLPED)*, 2000, pp. 9–14.

[9] L. Yuan and G. Qu, "Analysis of energy reduction on dynamic voltage scaling-enabled systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 12, pp. 1827–1837, Dec 2005.

[10] B. Zhai, D. Blaauw, D. Sylvester, and K. Flautner, "Theoretical and practical limits of dynamic voltage scaling," in *Design Automation Conference (DAC)*, 2004, pp. 868–873.

[11] X. Lin, A. Shafaei, S. Chen, T. Cui, and M. Pedram, "Impact of technology and voltage scaling on LEON3 processor performance and energy," in *Microelectronics Technology Unified Conference (S3S)*, Oct 2015, pp. 1–2.

[12] M. Ashouei, J. Hulzink, M. Konijnenburg, J. Zhou, F. Duarte, A. Breeschoten, J. Huisken, J. Stuyt, H. d. Groot, F. Barat, J. David, and J. V. Ginderdeuren, "A voltage-scalable biomedical signal processor running ECG using 13pj/cycle at 1mhz and 0.4v," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*, Feb. 2011, pp. 332–334.

[13] K. Craig, Y. Shakhsheer, S. Arrabi, S. Khanna, J. Lach, and B. H. Calhoun, "A 32 b 90 nm Processor Implementing Panoptic DVS Achieving Energy Efficient Operation From Sub-Threshold to High Performance," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 2, pp. 545–552, 2014.

[14] S. Lutkemeier, T. Jungeblut, H. K. O. Berge, S. Aunet, M. Porrmann, and U. Ruckert, "A 65 nm 32 b Subthreshold Processor With 9t Multi-Vt SRAM and Adaptive Supply Voltage Control," *IEEE Journal of Solid-State Circuits*, vol. 48, no. 1, pp. 8–19, Jan. 2013.

[15] S. Jain, S. Khare, S. Yada, V. Ambili, P. Salihundam, S. Ramani, S. Muthukumar, M. Srinivasan, A. Kumar, S. K. Gb, R. Ramanarayanan, V. Erraguntla, J. Howard, S. Vangal, S. Dighe, G. Ruhl, P. Aseron, H. Wilson, N. Borkar, V. De, and S. Borkar, "A 280mv-to-1.2v wide-operating-range IA-32 processor in 32nm CMOS," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, Feb. 2012, pp. 66–68.

[16] J. Hennessy, N. Jouppi, F. Baskett, and J. Gill, "MIPS: A VLSI Processor Architecture," in *CMU Conference on VLSI Systems and Computations*, 1981, pp. 337–346.

[17] B. Nikoli, M. Blagojevi, O. Thomas, P. Flatresse, and A. Vladimirescu, "Circuit design in nanoscale fdsoi technologies," in *Proc. 29th International Conference on Microelectronics*, may 2014, pp. 3–6.

[18] J. Kwong, Y. Ramadass, N. Verma, M. Koesler, K. Huber, H. Moormann, and A. Chandrakasan, "A 65nm Sub-Vt Microcontroller with Integrated SRAM and Switched-Capacitor DC-DC Converter," in *IEEE International Solid-State Circuits Conference - Digest of Technical Papers*, Feb. 2008, pp. 318–616.

[19] "A standard cell characterization flow for non-standard voltage supplies," in *29th Symposium on Integrated Circuits and Systems Design (SBCCI)*, 2016, p. 16.

[20] M. Gibiluka, "Analysis of Voltage Scaling Effects in the Design of Resilient Circuits," Master's thesis, Faculdade de Informatica, Pontificia Universidade Catolica do Rio Grande do Sul, 2016.

[21] A. L. R. Rosa, L. B. Soares, K. H. Stangherlin, and S. Bampi, "Designing cmos for near-threshold minimum-energy operation and extremely wide v-f scaling," in *Proceedings of the 28th Symposium on Integrated Circuits and Systems Design*, 2015, p. 1:11:6.