Body Bias Control for Real-Time Systems

- Optimization and its Switching Parameters Analysis for FD-SOI Technology -

Keio University

Carlos Cesar Cortes Torres

A dissertation submitted in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

School of Science for Open and Environmental Systems
Graduate School of Science and Technology
Keio University

May 2020

I would like to dedicate this thesis to my loving mother Maria del Socorro Torres Ramirez. For her endless unconditional support and encouragement. Whose education and discipline have led me to the completion of my graduate education ...

Me gustaría dedicar esta tesis a mi querida madre María del Socorro Torres Ramírez. Por su interminable apoyo y aliento incondicional. Cuya educación and disciplina me han llevado hasta la finalización de mi educación de posgrado ...

Acknowledgements

This research has been conducted at Keio University Faculty of Science and Technology Department of Information and Computer Science. I want to thank my advisor Prof. Amano Hideharu, for accepting me in his laboratory and his support during this study.

I want to thank Prof. Yamasaki Nobuyuki, Matsutani Hiroki from the Dept. of Information and Computer Science, and Nakano Nobuhiko from the Dept. of Electronics and Electrical Engineering of Keio University Faculty of Science and Technology for kindly accepted to review this thesis and the discussion for my defense. I'm grateful with Keio University for granting me Kenkyu Susume scholarship and Keio Leading-edge Laboratory Ph.D. Program Research Grant. This study was part of the "Ultra-Low Voltage Device Project" supported by the Ministry of Economy, Trade, and Industry (METI), the New Energy and Industrial Technology Development Organization (NEDO), and Japanese Society for the Promotion of Science (JSPS) KAKENHI. I thank Cadence for the software and tools provided. I would like to also thank one of my role models Jesus Palomino for his support.

I would like to offer my special thanks to Steffen Bähr and Prof. Dr.-Ing. Dr. h. c. Jürgen Becker from Institutsleitung des Instituts für Technik der Informationsverarbeitung (ITIV) of Karlsruhe Institute of Technology (KIT), for inviting and welcoming me at KIT. For their valuable assistance to make possible my doctoral stay in Germany and the environment necessary for the completion of this thesis. To all other colleagues of KIT-ITIV I wish to thank for their engagement and the pleasant working atmosphere.

Finally, while I was carrying out this study, I was supported by many people, family to inspire me to pursue graduate education and friends from which I received encouragement and advice. People I met throughout this study, sincerely, I wish to express my gratitude to everyone that cooperated through and the conclusion of this journey.

Carlos Cesar Cortes Torres
Yokohama, Japan/Karlsruhe, Germany
May 2020

Abstract

In the past decade, real-time systems (RTSs), which must maintain time constraints to avoid catastrophic consequences, have been widely introduced into various embedded systems and Internet of Things (IoTs). It is essential for any of these embedded systems be energy efficient due to long battery life is important. Such systems tend to work intermittently and reducing leakage power in the idle state is essential. However, traditional energy models idealize overheads, they often require a significant amount of power since they must directly control the system supply voltage or cannot deal with the overhead of adjusting the Body Bias (BB) voltage. Moreover, when the power supply is powered down the data in the memory element is lost, thus the models are not accurate.

Dynamic Body Bias scaling is a promising approach to managing leakage energy and operational speed. In this study, I investigated the RTS energy efficiency by analyzing the ability of BB, applying dynamic body bias control in providing a satisfying tradeoff between performance and energy. Although BB is an efficient technique to reduce the leakage power, it has not been commonly used dynamically because of the large timing and energy overhead when a conventional CMOS process is used. However, recent System On Insulator (SOI) technologies enabled the use of dynamic body bias control with acceptable overhead. Here, I focus on Silicon On Thin Box (SOTB), a type of Fully Depleted (FD) SOI technology which can control BB widely with a small overhead. For the BB control, I analyzed the timing and energy overhead of two simple microcontrollers and a dynamically reconfigurable processor with SOTB technology. I propose a practical energy and timing model that includes switching transition and idle regions analysis. It is based on extracted real-chip parameters. First, I optimize VDD and BB voltages by using brute force coarse-grain method. Secondly, I propose a more accurate energy overhead model by using an analytical double exponential expression; I transform the real-chip physical parameters of the double exponential waveform into analytical function coefficients. Finally, I build an optimization model with a Non-Linear Programming.

The use of the proposed model resulted in an energy reduction of about 32% at lower

frequencies as compared with the conventional model. Moreover, the energy overhead was reduced to approximately 14% of the total energy consumption. This methodology provides a framework and design guidelines for real-time systems and CAD.

Table of contents

List of figures

List of tables

Nomenclature

Following are the symbols and abbreviations that are used within the body of this dissertation.

Symbols

$\alpha \cdot C$	Coefficient of dynamic energy corresponding to the switching activity factor of the capacitance
α	Velocity saturation coefficient for the MOSFET
δ	$Delta$ Factor related to tail time
ε	Process parameter
γ	$Gamma$ Factor related to raise time
κ	$Kappa$ Amplitude modifying factor
A	Coefficients of the exponential term
B	Coefficients of the exponential term
C	Capacitance
CPI	Clock cycles per instruction
D	Deadline
E_d	Dynamic Energy
E_s	Static Energy
E_T	Total Energy
E_{id}	Idle Energy

E_{ovs}	Energy overhead of the sleep-down transition
F	Coefficient related to frequency
f	Frequency
I	Leakage current
I_0	Current amplitude of the double exponential
I_{ovs}	Current at transition overhead period
I_{peak}	Current peak of the double exponential waveform
K_γ	Constant given by the technology process coefficient (back gate biasing)
P_d	Dynamic power consumed during the active state
P_s	Static power consumed during the active state
P_{ideal}	Ideal power consumption
P_{id}	Power consumed during the idle state
t_s	Sleep-down time
t_w	Wake-up time
T_{exe}	Execution time
T_{id}	Idle time
T_{ovsT}	Overhead sleep time. Time needed to establish the necessary VDD and VBN
t_{rise}	Raise time. Time it takes to reach the current peak of the double exponential waveform
t_{tail}	Tail time. Time it take to settle the double exponential waveform

Subscripts

n	n-well
p	p-well

Acronyms / Abbreviations

BB	Body Bias
BET	Break Even Time
CAD	Computer-Aided Design
CPI	Cycles Per Instruction
DPM	Dynamic Power Management
DVS	Dynamic Voltage Scaling
FBB	Forward Body Bias
FD	Fully Depleted
FPGA	Field Programmable Array
GIDL	Gate Induced Drain Leakage
IoT	Internet of Things
IPM	Interior-Point Method
ISA	Instruction Set Architecture
MIPS	Microprocessor without Interlocked Pipeline Stages
MOSFET	Metal-Oxide Semiconductor Field-Effect Transistor
NLP	Non-Linear Programming
PE	Processing Element
PG	Power Gating
RBB	Reverse Body Bias
RISC	Reduced Instruction Set Computing
RTS	Real Time System
SI	Switching Impulse wave
SOI	Silicon On Insulator
SOTB	Silicon On Thin Box

SRAM	Static Random Access Memory
TCI	Through Chip Interface
TLB	Translation Lookaside Buffer
V850	V850 E-Star Microcontroller
VBN	Voltage Body bias to a n-well
VBP	Voltage Body bias to a p-well
VLSI	Very-large Scale Integration
Vth	Threshold voltage
ZBB	Zero Body Bias

Chapter 1

Introduction

1.1 Background and Purpose of the study

Real-time systems (RTSs) are part of our daily lives, being used in different domains, such as home appliances, medical systems, robotics, security, aeronautics, and many others. One class of these systems is used for highly time-critical tasks that should be executed in a predefined deadline. When failing to meet this deadline, the executed task's results can be corrupted, or even the entire system might fail, possibly leading to catastrophic consequences.

At the same time, and with the increasing popularity of the Internet of Things (IoT), the need to design RTSs that can be embedded in small devices has become a necessity.

These devices tend to work intermittently, working periodically with a certain time interval or kicked up by some events. It means that they are in a sleep or waiting mode a long time, and the leakage power reduction is essential, in such a period, especially in battery driven systems.

Such systems have two modes: (a) an active mode working at high clock frequency and (b) a sleep mode for the idle time. In such a sleep mode, the clock is stopped, that is, the dynamic power is completely suppressed. Thus, the problem is how to reduce the leakage power in such idle time. This issue is especially important for battery driven devices (i.e., embedded RTSs require a battery life of a few years) and should operate on a limited power budget in the order of milliwatts. As technology continues to scale, the leakage current will keep increasing and, therefore, strict control is needed to find an optimal operational region. Hence, the energy consumption should be kept minimum while making sure that the timing constraints are met.

The RTS energy efficiency has been extensively studied, and some have focused on very-large-scale integration (VLSI) designs. Various techniques, including power gating (PG) [**?**] for dynamic power management (DPM) [**?**], and dynamic voltage scaling (DVS) [**?**] have been introduced in RTSs. Although these techniques improve energy efficiency, they often require a significant amount of power since they must directly control the system supply voltage; e.g. (PG) which shuts down the power supply with high threshold transistors, is a commonly used technique. First, the area overhead of power switches and isolation cells is not negligible, especially for simple embedded systems. By shutting down the power, the data in the registers or memory are disappeared. It causes another overhead, and circuits to save and restore the data when it goes back to the active mode. Finally, turn-on and off power switches (consisted of many transistors) often cause electric noise on the power grid.

Body bias (BB) control is another solution that currently attracts the attention of designers because it can improve RTS energy efficiency as it can manage the tradeoff between power leakage and performance, without affecting the power supply [**? ? ? ?**]. It increases the threshold by giving a strong reverse bias to the substrate of transistors, being especially efficient in fully depleted silicon-on-insulator (FD-SOI) technology [**? ? ?**], which is commonly used for low-power systems. Additionally, it can reduce the leakage power in the sleep mode without touching the power supply. Thus, power switches nor isolation cells that require a large area for implementation and energy overhead for switching are needed. Although a pair of body biasing wires and triple gate structure are needed, the overhead is usually smaller than that of the PG. Also, the data in registers or memory cells are kept in the sleep mode. Although, for accomplishing this, the effect of leakage reduction is not enough in the traditional bulk process, and recent Silicon on Insulator (SoI) technologies can suppress its leakage similar or more compared to the PG. Its effect is further endorsed when systems are enabled with silicon on thin box (SOTB) technology[**?**], which is a novel and advanced fully depleted silicon on insulator (FD-SOI) technology.

However, changing the transistor threshold by BB requires a significant timing overhead, which is the main reason why the dynamic BB control is not commonly used. In this case, almost no power is needed to keep the reverse BB from the body bias supply, but it also consumes some energy when the state is changed from zero or forward bias to reverse bias for charging the substrate. Thus, such a BB control must be done when

the expected gain of energy is larger than the overhead.

Therefore, combining the benefits of SOTB and adaptive BB can drastically suppress the leakage current. However, when controlling the BB on RTSs, the BB must be controlled dynamically so as not to miss the deadline. Although the energy for statically maintaining the BB voltage is quite small, the dynamic control of the BB requires considerable energy. In this direction, several studies on dynamic BB control have been conducted [**? ? ? ?**], but they were not based on accurate real-chip measurements that include the BB switching-voltage overhead.

Furthermore, a minimum idle period to get a gain in energy savings is referred to as the Break Even Time (BET). Hence, here the BET is the time at which the energy saved by applying BB to enter into idle state becomes equal to the energy overhead.

In the scenarios of SOTB and adaptive BB previous studies on dynamic BB control [**? ? ?**] have two limitations. Firstly, they were not based on a model of BB switching-voltage overhead. Most of them ignored it or included the energy consumption in the active state. Secondly, a method has not been proposed for finding the optimal BB voltage and the optimal supply voltage for meeting task deadlines.

Based on the above, we investigated RTS energy efficiency by analyzing the dynamic BB control on performance and energy, including both the physical energy and timing overheads when executing the voltage transitions. To this aim, we propose (a) a *practical timing* and (b) a *power mathematical models* capable of determining the energy consumption based on the task execution while taking into account a given deadline constraint. Additionally, we have now devised a standard full switching impulse voltage (double exponential) expression [**? ? ?**], a mathematical expression of the transient, for estimating the switching energy. We also devised an interior point method (IPM) based on our power model that can be used to obtain optimality in nonlinear programming (NLP).

1.1.1 Research background

Silicon on thin box - SOTB

The target technology is silicon on thin box (SOTB) technology, a novel FD-SOI technology [**?**]. It features latch-up immunity, superior high temperature tolerance, high performance, radiation hardness, and high BB sensitivity. These characteristics are possible due to its insulating "buried oxide" layer widely used in SOI devices [**? ?**].

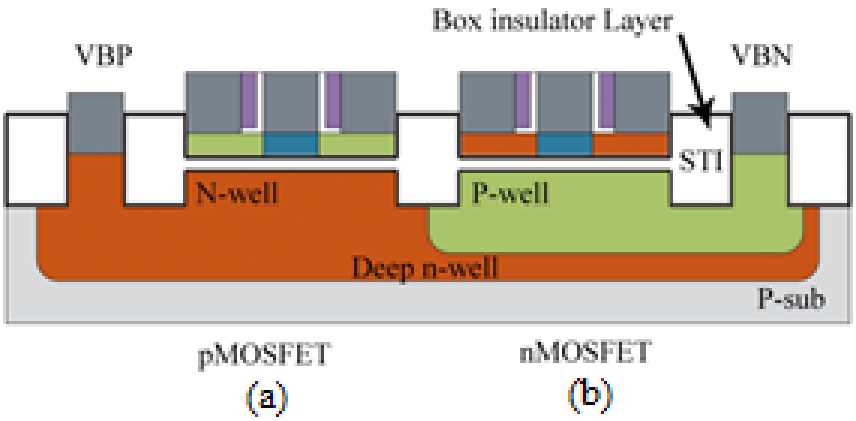


Fig. 1.1 Cross-sectional view of SOTB MOSFET: (a) pMOS and (b) nMOS

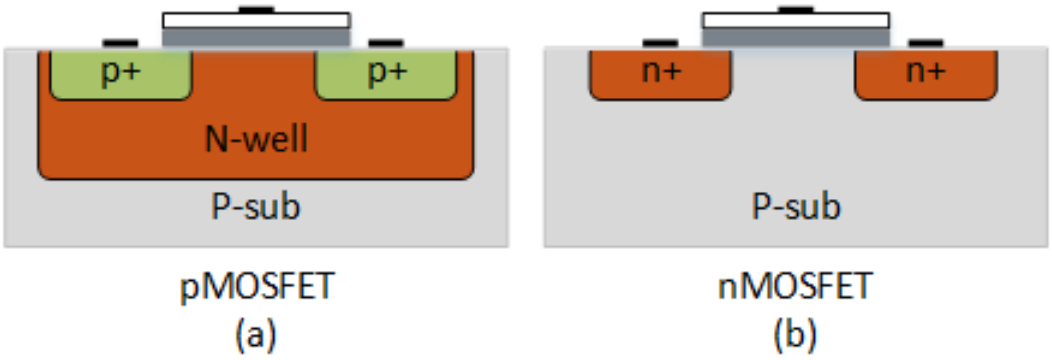
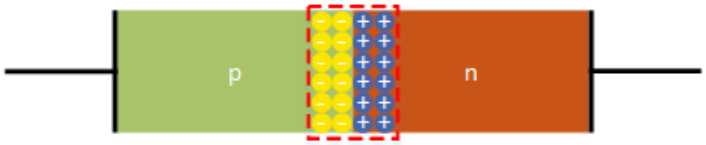


Fig. 1.2 Cross-sectional view of traditional MOSFET: (a) pMOS and (b) nMOS

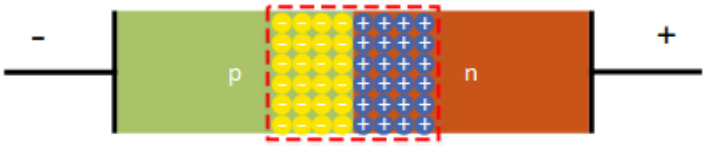
These body-driven characteristics enable high caliber energy reduction using the BB. Unlike other conventional FD-SOI devices, a SOTB device is formed on an ultra-thin box layer (about 10 nm), as shown in Fig. 1.1, additionally, contrasting with traditional transistors, in the SOTB, the BB can control the leakage power and delay of the transistor in the wide range even with the low VDD voltage. Consequently, SOTB ensures a more efficient reduction in leakage current using BB control than other conventional metal–oxide–semiconductor field-effect transistors (MOSFETs). The Fig. 1.2 illustrates the traditional MOSFET.

Body Bias Control

The states of SOTB are classified according to the nMOS BB voltage VBN, the pMOS BB voltage VBP, and the supply voltage as VDD. As with other FD-SOI technologies,



(a) Zero Body Bias.



(b) Reverse Body Bias.

(c) Forward Body Bias.

Fig. 1.3 Depletion region according to the body bias applied.

the default state of a given MOSFET (VBN=0 and VBP=VDD) in SOTB technology is called Zero Body Bias (ZBB). If a lower voltage is applied to the nMOS body (VBN$<$0) and a higher voltage is applied to the pMOS body (VBP$>$VDD), the depletion width increases, so the threshold voltage increases. This condition is known as reverse body bias (RBB). In contrast, if a higher voltage is applied to the nMOS body (VBN$>$0) and a lower voltage is applied to the pMOS body (VBP$<$VDD), the depletion width decreases, so the threshold voltage decreases. This condition is known as forward body bias (FBB). These characteristics are illustrated in Fig. 1.3

SOTB-MOSFET body biasing behavior characteristics are summarized in Table 1.1. As shown in Table 1.1, with RBB, since the threshold voltage is higher, the leakage current is lower at the expense of an increase in the delay time, so performance is degraded. With the FBB the threshold voltage is lower, the delay is reduced, hence, it can achieve high operating speeds, increasing the performance at the cost of leakage current.

Although the RBB for RTSs is useful in reducing the leakage current in the sleep mode, it should be carefully selected; and accurate timing and energy models should be elaborated to avoid any unnecessary leakage overhead (steep FBB) or timing require-

Table 1.1 Body Bias characteristics. Tradeoff between performance and power.

Characteristic	Body Bias voltage	
	Reverse Body Bias	**Forward Body Bias**
Voltage	VBP $\geq$ VDD	VBP $\leq$ VDD
	VBN $\leq$ 0	VBN $\geq$ 0
Leakage Power	Low	High
Performance	Low	High
Delay	Increase	Decrease

ment unsatisfactory (excessive RBB) as not to miss the deadline.

1.1.2 Related Work

A significant number of studies have been carried out to enhance the energy efficiency in embedded RTSs, including voltage algorithms, and device-level solutions.
In this direction, the DPM is a technique that reduces the energy dissipation of RTSs with low power idle states over sufficiently long intervals [**?**]. DPM consists of shutting down the power supply so that in idle states, the chip does not consume any power. It usually employs header switch transistors (sleeping transistors, where a high threshold device is connected in series with low threshold transistors). DPM alone cannot guarantee the overall system energy consumption is minimized. It must ensure that the non-trivial energy switching overheads do not offset the energy savings obtained during the device intervals. DPM might lead to excessive CPU power consumption. DPM presents non-trivial difficulties in real-time settings. Hence, this technique introduces a number of design issues. Under the condition that a power supply is cut-off for idle states, volatile data are discarded. When data need to be preserved, a certain level of voltage has to be supplied as a power supply. Hence, the power-leakage reduction is restricted to such conditions. Additionally, it requires switching on/off time and energy for saving register values and restoring cache contents. Also, this causes an execution overhead, being its performance degradation unpredictable [**?**]. In the Table 1.2, I summarize the DPM versus the RBB characteristics.

Leakage reduction methods of systems working with a certain interval tend to use PG, and the design methods for quick transition between wake-up mode and sleep mode have been widely investigated in [**?**]. Additionally, a processor that provides functional

Table 1.2 Dynamic Power Management VS Reverse Body Bias main characteristics.

Dynamic Power Management (DPM)	**Reverse Body Bias (RBB)**
Power supply cut-off in idle state, volatile data are discarded	Data stored in registers and memory modules are saved.
Reduces energy leakage in idle states.	Reduces energy leakage in idle states.
Widely used.	Has not been commonly used.

units with nano-second order mode transitions is reported [**?**]. Hence, this kind of power gating method must employ algorithms as well to meet RTSs deadlines.

Therefore, the leakage power dissipation increases exponentially due to the sub-threshold leakage current, although there are some techniques to minimize this effect, called Multiple-Threshold Voltage CMOS (MTCMOS), to create a virtual power supply and ground rails whose voltage levels are very close to the real ones.

Another problem is that they are prone to reduced performance and noise [**? ? ?**]. Hence, these kinds of solutions are impractical for embedded RTS, being the reason why compared with such activities, much less number of dynamic BB control has been reported.

D. Duarte et al. [**?**], made a compendium and summarized some leakage reduction techniques with its corresponding characterization. It introduces an interesting gating supply voltage alternative technique, by using Phase-Locked Loops (PLLs) as voltage regulators, which at the same time can support leakage reduction by power supply gating. He expressed the importance of the minimum idle time to avoid exceeding the given deadline, targeting embedded systems.

In Fig. 1.4 we illustrate Ikebuchi et al. [**?**] design proposal for PG to manage energy consumption. We could see any PG design uses PG-cells, Power switches and Isolation cells. Ikebuchi et al. showed a good performance versus Discrete Cosine Transform, Dijkstra and Quick Sort benchmarks shown in Fig. 1.5. In his study, the PG goes from 160μW up to 900μW. Nevertheless, the data is lost when the power supply is shut down and it needs a large area for additional circuitry. In the Table 1.3, I summarize the PG versus the RBB characteristics.

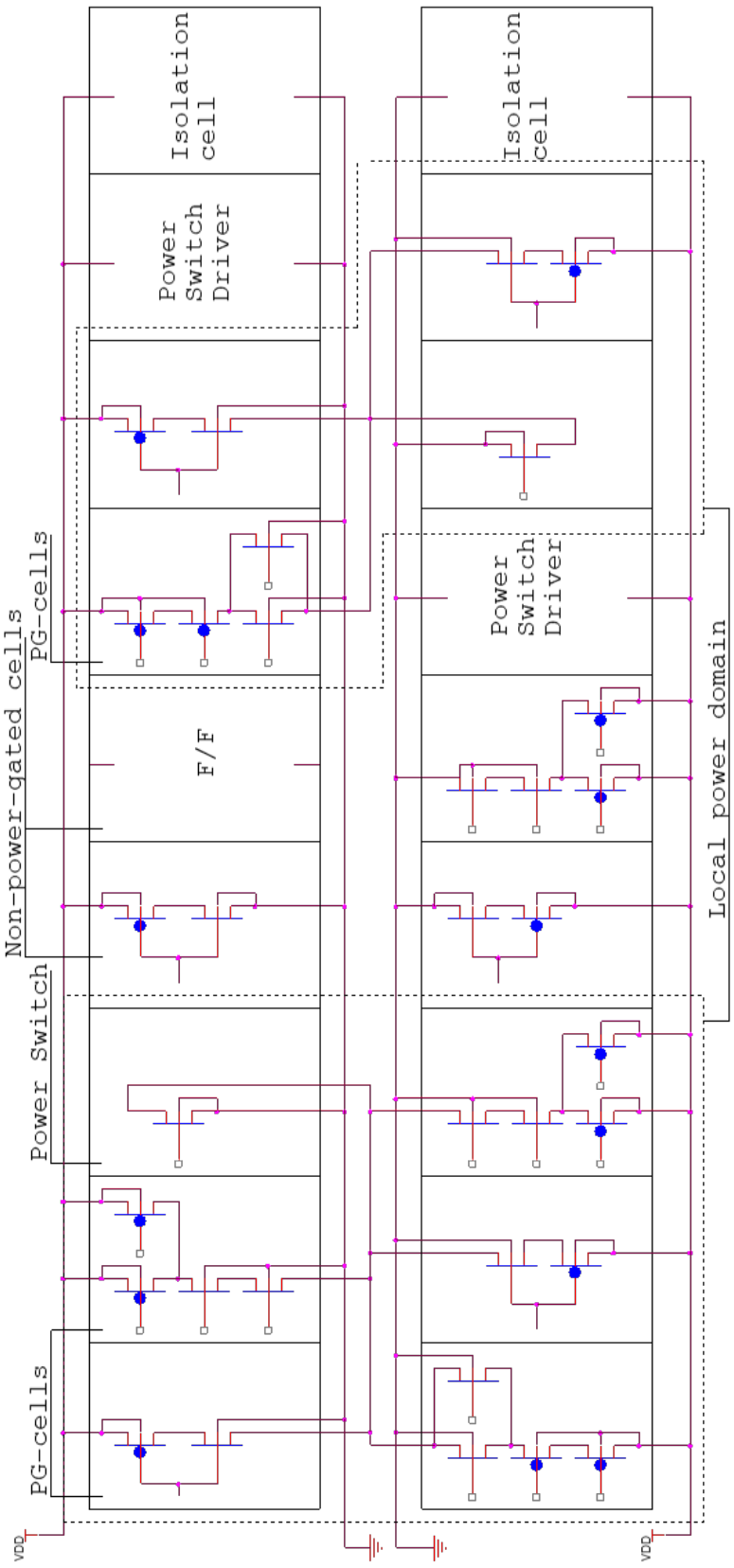


Fig. 1.4 PG design proposed by Ikebuchi et al. [**?**].

Table 1.3 Dynamic Power Management VS Reverse Body Bias main characteristics.

Power Gating (PG)	**Reverse Body Bias (RBB)**
When shutting down power supply, data is lost.	Data stored in registers and memory modules are saved.
Timing overhead nano second order.	Timing overhead micro second order.
Additional transistors for power switches and isolation cells.	No additional circuitry is needed.
Large area for additional circuitry.	NA
Generates electric noise when turn on/off.	NA
Widely used.	Has not been commonly used.

Otherwise, DVS algorithms have been widely employed in modern computer systems. Such algorithms are techniques that decrease the power supply voltage while keeping application deadlines [**?**]. These algorithms can drastically reduce the dynamic power due to the quadratic power-supply dependency while providing the necessary peak computation power in general purpose systems. However, the range for power-supply scaling is highly restricted when the power supply voltage is near the threshold region [**?**].

DVS reduces the dynamic CPU energy consumption. The CPU clock frequency and the supply voltage can be adjusted dynamically on-the-fly. DVS leads to short device idle intervals (it might limit DPM). It might spend more time in performing the same computation. DVS presents non-trivial difficulties in real-time settings. The Fig. 1.6 illustrates the basic concept for DVS scheme.

Additionally, they are overlooked for time constraint applications (RTS), because the scaling of processor frequency could be prejudicial. Such limitations can drastically impact the efficiency of energy saving. To get the benefits of the DVS in RTSs embedded systems (RT-DVS), schedulers must be implemented to ensure the tasks are executed in time [**?**].

Some of the RT-DVS algorithms proposed are not well balanced enough. One of the problems is that these algorithms lower the frequency in one cycle that in the next cycle, high voltage and frequency are required to meet the deadline, resulting in a performance penalty (scheduling overheads). Hence, lowering the frequency is obtained by comparing the worst case specification and idle utilization. This condition could cause excessive and conservative assumptions and provoke a non-deterministic behavior.

For RT-DVS schedulers commonly used are Earliest Deadline First (EDF) or Rate

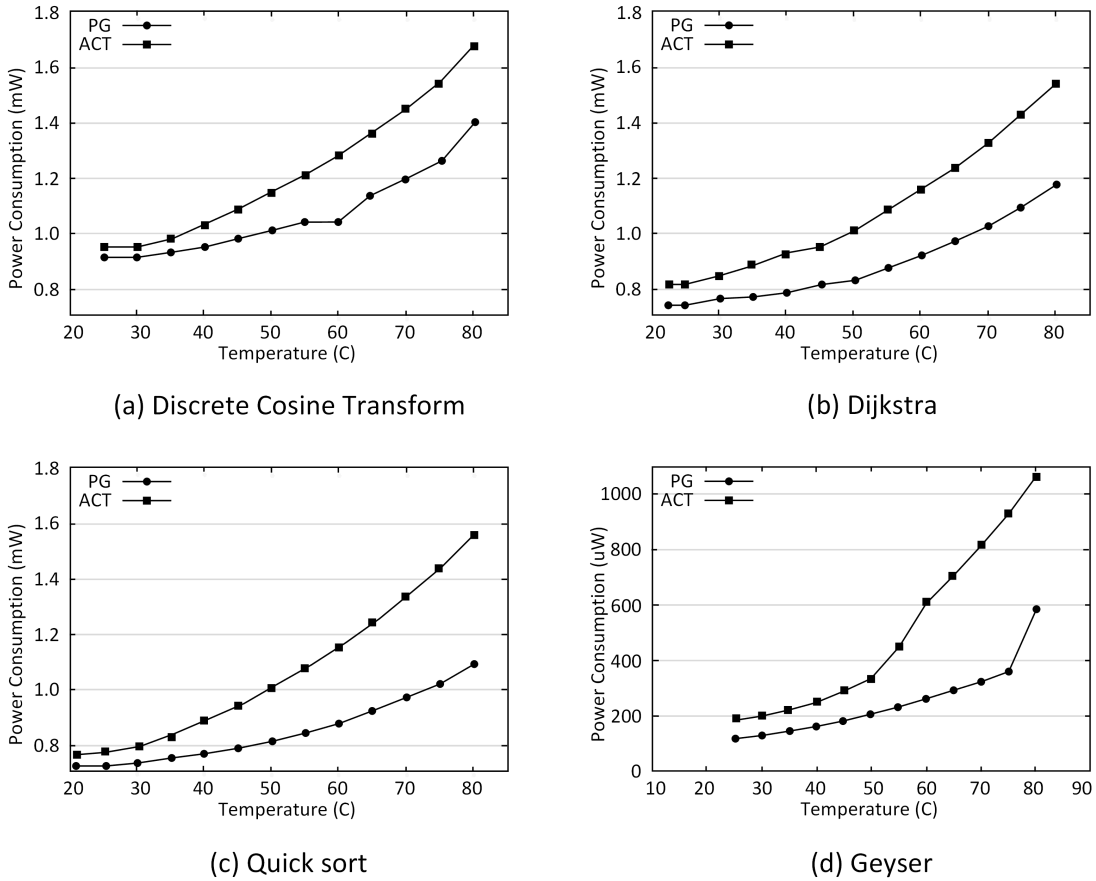


Fig. 1.5 Examples of the PG technique applied to different benchmarks [**?**].

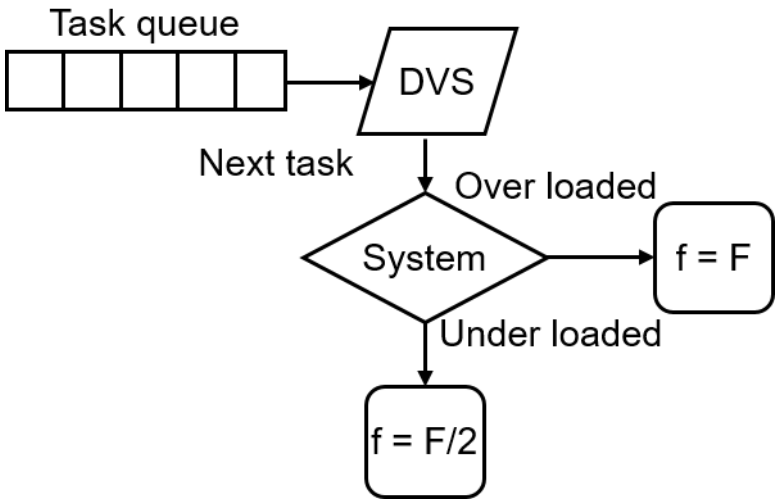


Fig. 1.6 Simple concept for DVS scheme.

Monotonic (RM). Both algorithms consist of prioritizing the tasks by time execution, either statically or dynamically, and delay the task executions as late as possible; thereby, group idle periods. In this way, the processor can be idle for a longer period with a smaller number of power transitions.

Devadas et al. [**?**] proposed a unified framework for DPM and DVS. It considers the trade-offs between the DPM and DVS policies. Typically, DVS and DPM components tend to favor power management configurations with opposing features. Reducing the processing frequency (to favor DVS) will scale up task execution times and hence seriously constrain the DPM opportunities. On the other hand, configurations that favor DPM will create long idle intervals that require high CPU frequencies and will also need to take into account the device transition overheads. Detecting and reclaiming unused CPU time (slack) has been a major tool for dynamic DVS schemes. The most novel aspect of DFR-EDF allows the use of the same dynamic slack for DVS and DPM. However, this proposal is not based on real-chip values. This unified framework DPM-DVS is shown in Fig. 1.7.

Other proposed techniques are based on either timeout mechanisms or stochastic methods. However, they cannot be applied to RTSs due to its unpredictability [**?**].

The Variable-Threshold CMOS (VTCOMS) is another technique to reduce leakage energy. It allows us to dynamically adjust the threshold voltage of the transistor by applying a body bias [**?**]. Some of the methods to modulate the threshold voltage are Adaptive Supply Voltage (ASV) and BB, which both of them have been proven to be

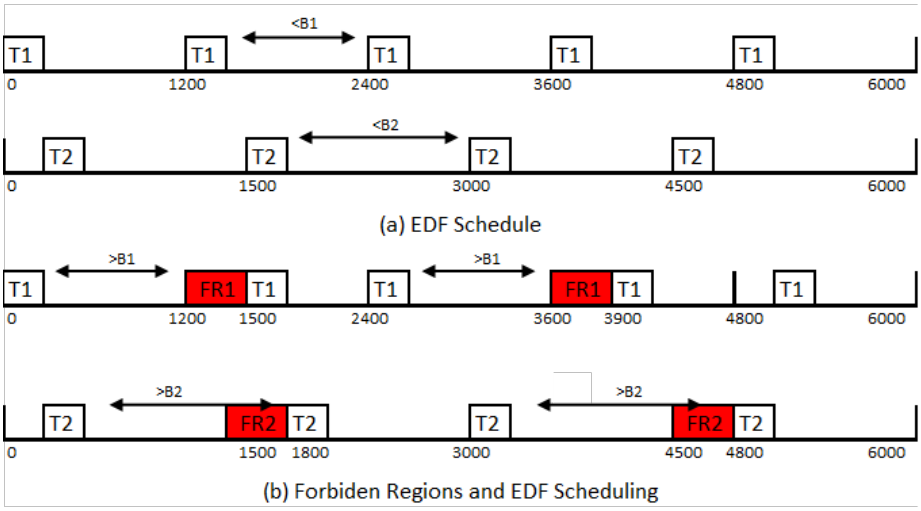


Fig. 1.7 Unified framework for DPM-DVS [**?**]. DFR-EDF allows the use of the same dynamic slack for DVS and DPM.

effective methods to trade performance versus power and vice versa, with similar energy efficiency. They have been characterized in [**? ?**]. Although BB needs additional on-chip power distribution networks for the body voltage and also adds design complexity, ASV shows some reliability issues. Hence, from a design perspective, for embedded low power systems, these problems cannot be overlooked, making BB the best choice for our study [**? ?**].

Some studies have analyzed the benefits of combining DVS/DVFS and adaptive BB for energy reduction achieving successful results by developing algorithms and power models [**? ? ? ?**]. These models can calculate an optimal power supply and BB voltage for each operational frequency. The authors assumed ideal voltage regulators that can output any voltage obtained from the models. However, the actual voltage drivers have a certain output-voltage resolution limitation.

In the Table 1.4, I summarize the DVS/DVFS versus the RBB characteristics.

To maximize energy reduction and meet the deadline, Yan et al. proposed a task-scheduling algorithm and energy models for RTSs usage [**?**]. It identifies the optimal trade-off point between a supply voltage and BB voltage. It also considers the trade-off between energy consumption and clock period using heuristics equation.

It computes the optimal energy consumption at a given clock frequency.

To meet the time criteria, it evaluates the validity of the generated schedule by checking the Earliest Start Time (EST) and Latest Finish Time (LFT).

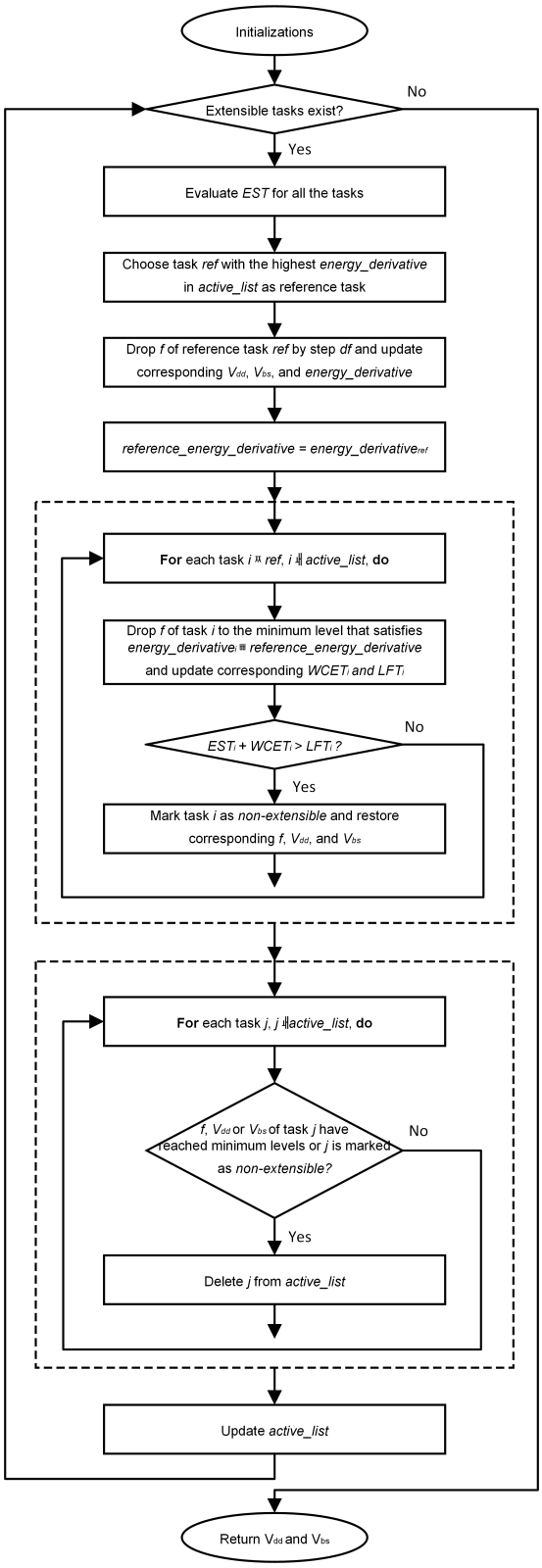


Fig. 1.8 Joint dynamic voltage scaling and DBB algorithm [**?**].

Table 1.4 Dynamic Voltage Frequency Scaling (DVS/DVFS) VS Reverse Body Bias main characteristics.

Dyn. Voltage/Frequency Scaling (DVS/DVFS)	**Reverse Body Bias (RBB)**
Range for power-supply scaling is highly restricted when the power supply voltage is near the threshold region. [**?**]	Work at near threshold region.
Provide significant energy savings (proportional to f^2)	Provide significant energy savings (Endorsed by FD-SOI/SOTB) [**?**].
Widely used.	Has not been commonly used.

Nevertheless, Yan et al. targets CMOS technology, it assumes a transition time of VDD and BB a conservative estimate based on the current circuit technology (Not based on real-chip values) and do not fully analyze switching overheads. In Fig. 1.8 it illustrates the joint dynamic voltage scaling and DBB algorithm.

We could also improve the accuracy for these frequency scaling modes by choosing unique voltage settings that maximize the average power reduction of the whole frequency range (maximizing the operating points, identifying BB and supply voltages and frequency). This selection is done by filling the gap between the discrete operating points [**?**]. These models can calculate an optimal power supply and BB voltage for each operational frequency. By using the obtained voltages, the algorithm schedules a task so as not to violate the deadline. Namely, the authors assume ideal voltage regulators that can output any voltage obtained from the models. However, the actual voltage drivers have a certain limitation in terms of output-voltage resolution.

Akgul et al. proposed a power-management method considering these voltage constraints [**?**]. The authors assumed discrete power-supply voltages and succeeded in reducing the energy even under the restrictions mentioned earlier.

Akgul et al. used a Piece-Wise Convex Subset (PWCS) method. Given a target frequency, their method has two modes: 1) The target frequency F_{target} can be applied with a PM belonging to the PWCS. In this mode, the performance requirement will be achieved by applying directly F_{target}. 2) F_{target} does not belong to the PWCS. The task is processed in hopping execution by applying the highest frequency F that belongs to the PWCS with F < F_{target} and the lowest frequency F belonging to the PWCS with F > F_{target}.

However, these studies were not based on parameters from real chips, and the

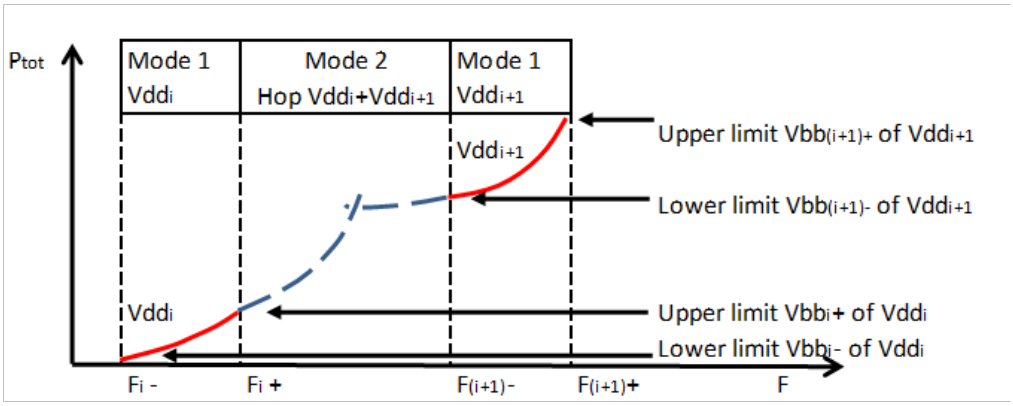


Fig. 1.9 Piece-Wise Convex Subset (PWCS) method, Akgul et al. [**?**]. Given a frequency, it uses 1) A predefined frequency, 2) The maximum frequency in a hopping execution.

switching overhead of adjusting the BB was not considered. The Fig. 1.9 illustrates the PWCS method.

Another aspect to consider Power-aware resource management techniques decrease the energy consumption by selectively placing idle components into low power states. An ideal power management policy would place a device in the sleep state only when the idle period is longer than the BET. Unfortunately, in most real systems, such ideal prediction of idle period is not possible.

Although methods to compute BET have been reported in several literatures, these trials of analysis are done for traditional MOSFET technologies[**? ?**]. [**?**] analyzed BET using a nonvolatile SOTB SRAM (NV-SRAM). However, these studies do not specify the energy overheads and utilize the PG technique, which cannot preserve volatile data when a power supply is gated for leakage reduction. Also, Kondo and his colleagues proposed a scheduling method with the operating system and compiler to keep the BET on functional units of a microprocessor[**?**]. However, it also used the PG technique.

There are a few trials for dynamic BB control and the BET, while the BET has not been evaluated. Kuhen and his colleagues proposed the dynamic body bias control for dynamically reconfigurable systems[**?**], but their algorithm also lacks of the overhead analysis for changing the BB voltage.

In our previous study [**? ?**], we developed a power model using BB control. The model is based on real-chip measurements in terms of leakage current, switching current, and maximum operational frequency. However, ideal BB switching is also assumed. Several approaches have been proposed to improve energy efficiency. When considering

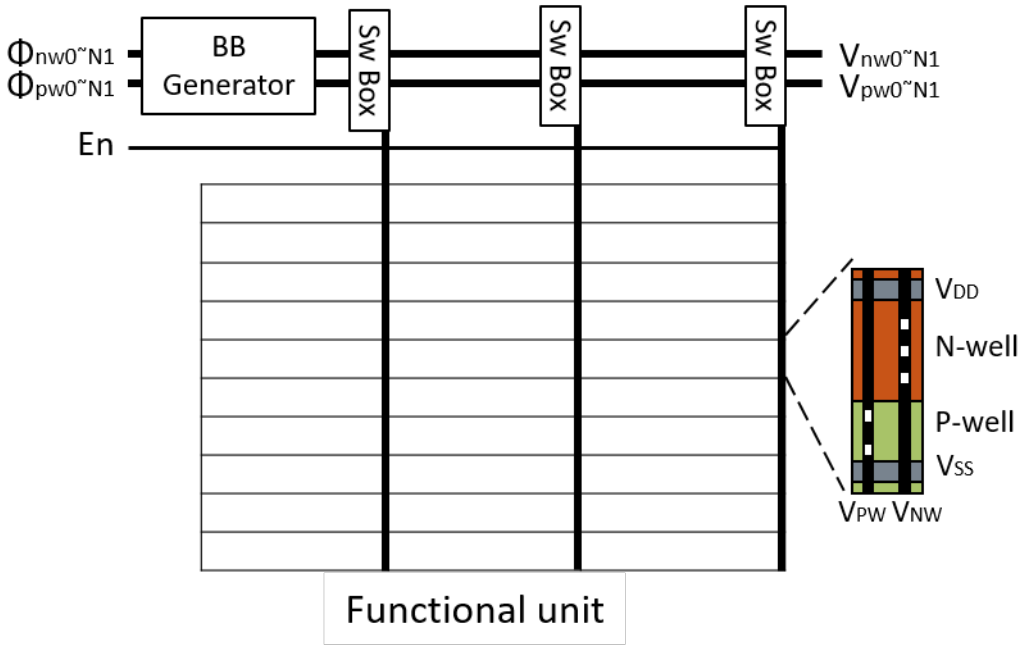


Fig. 1.10 DBB circuit level approach, step-wise sleep depth control [**?**].

overhead conditions or analyzing idle regions, all these approaches are based on the circuit level information [**? ? ?**].

Takeda et al. [**?**], propose a DBB Step-wise sleep depth approach to control leakage current. It automatically applies deeper sleep depth in a step-by-step manner. It applies the next sleep depth when the elapsed time reached to each pre-set time, called threshold, after an idle period starts. The stages are switched by switch boxes by selecting N-well and P-well biasing voltages from multiple generated voltages. When the number of stages is N, 2N voltage generators for N-well and P-well are necessary. It uses charge pump circuits for these voltage generators. The Fig. 1.10 illustrates the step-wise sleep depth control. However, this is a circuit level approach.

Nevertheless, we do not have any circuit level information of the target processor instead, we can measure the real chip. Very few works have system level and under ideal conditions without considering overhead conditions. Our goal is to obtain a realistic model just by the parameters from the simple evaluation of the real chip and process parameters.

Hence, to the best of our knowledge, none of the studies presented above incorporated these timing and energy overhead conditions in their energy-saving approaches targeted for RTSs. In this direction, we have established a functional mathematical model for power and timing by using parameters extracted from real chips [**? ?**]. Moreover, we

introduced several of the first studies that included energy overhead parameters also extracted from real chips [**? ? ?**]. Additionally, we devised a model that includes BB switching overhead. Finally, we performed an optimization in our model to obtain the optimized power supply and body bias voltages.

1.2 Hypotheses and the significance of this study

It is essential for any embedded systems to be energy efficient. Such systems tend to work intermittently and reducing leakage in the idle time is essential.

Based on the aforementioned in previous sections, we investigated RTS energy efficiency by analyzing the dynamic BB control on performance and energy, including the physical energy and timing overheads when executing the voltage transitions. To analyze the minimum idle period to get a gain in energy savings (BET). To this aim, we propose a practical timing and a power mathematical models capable of determining the energy consumption based on the task execution while taking into account a given deadline constraint.

In this sense, we define ourselves as hypotheses for this work that it is possible to insert mechanisms for efficient energy reduction for RTS based on:

- Real-chip physical electrical parameters.
- Mathematical power model that does include the switching overheads.
- Optimize the calculation for the power model.

To properly design efficient algorithms and schedulers, we must include such overheads. Therefore, in this study, we present the first studies to examine the Energy and BET using accurate parameters extracted from a real chip using SOTB technology employing BB control energy saving technique.

The significance of this study is compounded by two main points. The first one is switching region analysis to understand the key elements in such region. The second is the voltage optimization. Understanding the characteristics involved in the switching region, we investigate the optimal points for the operational region.

1.2.1 Switching region analysis

This analysis is based in two key aspects:

- Energy assessment.
- Timing assessment.

We investigate the energy components and its duration for the transition region. We correlate these two components with the supply voltage (VDD) and BB voltage (VBN). By understanding the characteristics in this region, we obtain information to design more accurate algorithms.

In this analysis, we summarized the activities as follows:

- To obtain the timing and energy overheads when switching the BB voltages for various types of modules implemented with SOTB technology, used for embedded processing: a processor core, memory modules, ALUs, and processing elements, of a dynamically reconfigurable processor.
- By using these measurements, a practical power model for scaling the BB according to the switching behavior, operational frequency, clock cycles per instruction (CPI), and time for a deadline is proposed. The proposed model can calculate the energy consumption for each task of a given RTS application.
- A mathematical model for calculating energy overhead.
- Performing Break Even Time analysis, providing design parameters.

1.2.2 Voltage optimization

We describe the tradeoff between saving energy and switching overhead. We define the problem and propose a solution in terms of Non-Linear Programming paradigm. By automating this evaluation, we can clearly analyze the trends and find design parameters and coefficients applicable to a number of platforms.

In this optimization, we summarized the activities as follows:

- A method for optimizing energy consumption by optimizing the BB voltage and supply voltage.
- A method for increasing the accuracy of the energy model.

Our method provides design guidelines for RTSs and computer-aided design (CAD).

1.3 Outline of the present study

This study is organized as follows. Chapter 1 provides a background of BB, FD-SOI technology, and related work of energy reduction. Chapter 2 is dedicated to explaining the timing and power mathematical models. We introduce the target systems in chapter 3. The energy and timing analysis is presented in chapter 4. We optimize the mathematical models in chapter 5. Finally, chapter 6 summarizes the findings and outlook of this study.

Chapter 1

We describe the background of this research. We mentioned about the importance of the RTSs, which are applicable to various fields. We explain the state of the art and the SOTB technology along with the Body Bias control. We described the benefits of using these combined technologies and introduced the issues of previous research had, such as: switching overhead, the tradeoff between voltage, and saving energy. We describe the conventional research on this field and the advantage of the present research.

Chapter 2

We describe the scenario to analyze. We explain in detail the mathematical base model we use for power and time evaluation. We also describe the VDD calculation based on alpha power law.

Chapter 3

We introduce the target systems we use for the analysis, describe its architecture and its electrical coefficients. We present the characterization of its principal energy features and electrical coefficients.

Chapter 4

We describe the methodology and the evaluation for this study. We present the energy consumed in different cases (deadlines). We explain in detail how the different components of the energy (Static, Dynamic, Overhead energy) behaves under such circumstances. We evaluate the energy model by using the Brute Force method. Finally, we discuss the accuracy of this model.

Chapter 5

We describe the mathematical optimization model. We introduce the concept of the Double Exponential waveform that we use to build an optimization model. We discuss

the tradeoff between low power and performance. From these concepts, we optimized the model based on the interior-method Non-Linear Programming. We compare the results found in chapter 4 with the results found in this chapter. Finally, we discuss the accuracy of this model.

Chapter 6

We state the conclusions obtained by the present research results, and we propose as well the possible future work.

Chapter 2

Analytical Model

2.1 Proposed Base Model

Typical RTSs are multi-processor and multi-task. Using DVFS technique, to meet the deadline, schedulers must be implemented. In Fig. 2.1, we show a single processor multi-task in which schedulers can program the task back to back and adjust the frequency to finish the task before the deadline or at the deadline. However, in this study, we focus in single task.

Without power-saving control, a task is executed in time t_{exe} and finishes at the given deadline. The frequency and voltage are constant all through the deadline. Hence, the power leakage consumed in the idle region is wasted, as shown in Fig. 2.2 (a). To reduce power leakage, most conventional models lower or shut down the power supply in the idle time. However, this requires a power-management circuit for controlling power supply, which requires a certain amount of current. Also, when the power gating is used, the data in the storage are lost without special mechanisms to save them. They are sometimes too heavy for small RTSs used for IoT. Instead, we investigated power-leakage control using the BB. Specifically, when the SOTB is used, power leakage with the strong RBB is extremely low, yet all data in the memory and registers are kept. Since the static power required for maintaining the BB is also small, it can be controlled by straightforward low-power circuits using the charge pump [**?**]. The power-supply voltage and clock frequency can be adjusted for each application but assumed to be constant during the execution. Since the clock gating is applied during the idle time, the dynamic power during this time is assumed to be zero.

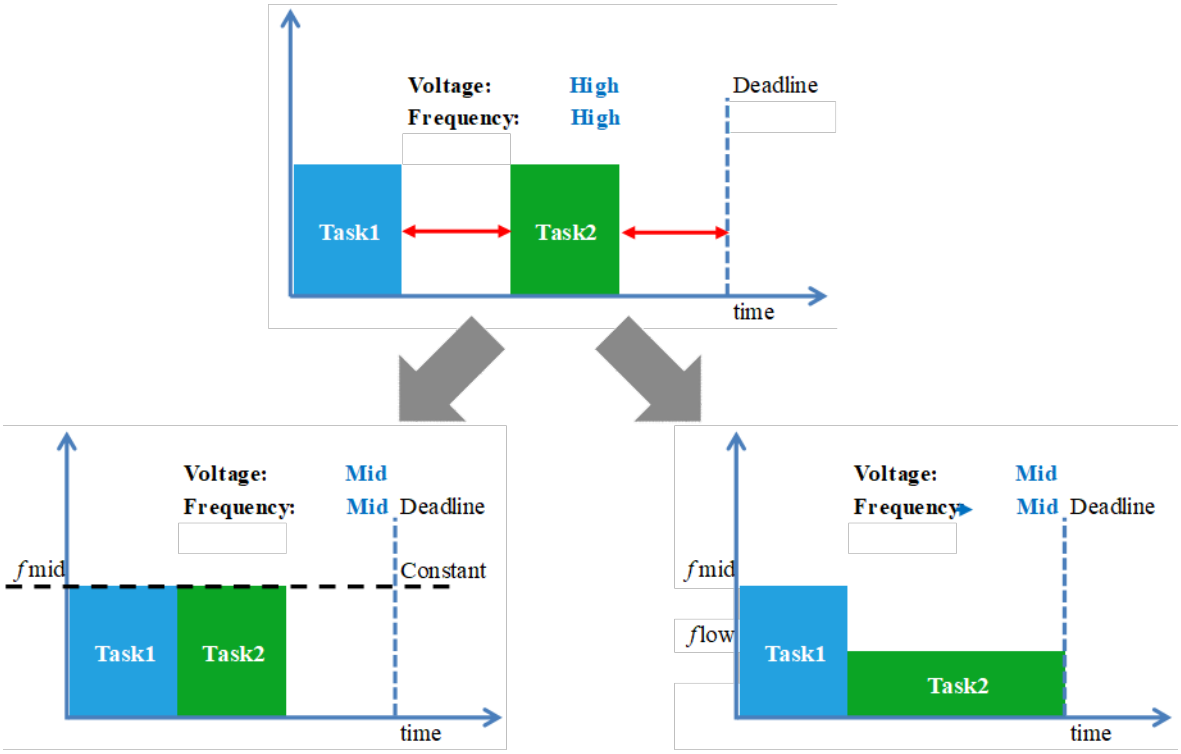
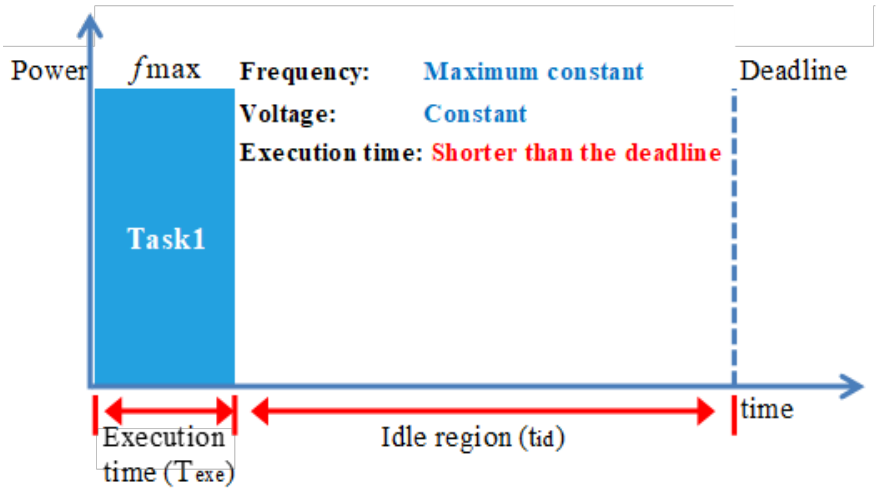


Fig. 2.1 In a single processor multi-task in which schedulers can program the task back to back and adjust the frequency to finish the task before the deadline or at the deadline

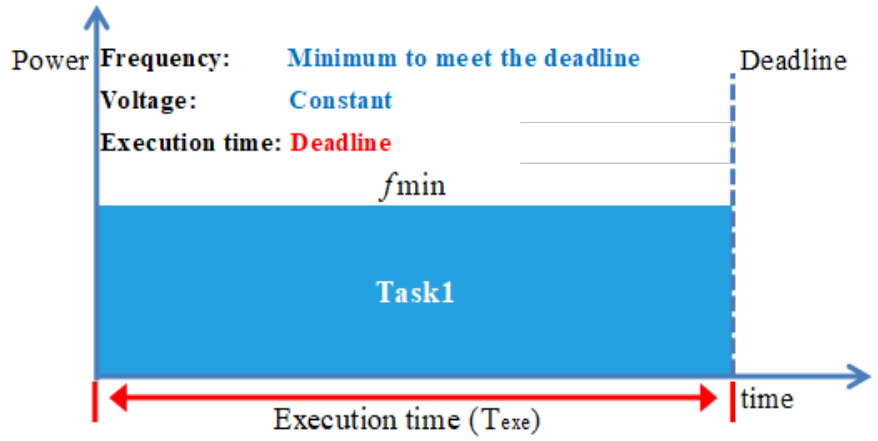
In this study, we focus on two possible scenarios to execute a given task on an RTS while considering a predefined deadline. In the first scenario, the system works at the minimum frequency at which the task execution finishes at the deadline. This means that the minimum VDD and ZBB voltages are supplied with a minimum frequency to satisfy the deadline (for example, 10MHz), as illustrated in Fig. 2.2 (b). This scenario is our baseline. In the second scenario, shown in Fig. 2.3, consists of optimizing the VDD to boost the frequency according to the alpha power law; hence, the task is executed in much less time than the first scenario. This is our test scenario. During the time remaining until the deadline, RBB is applied to reduce the leakage power. If the BB voltage is fixed and the substrate has been charged, almost no current is required for giving the biasing. If the voltage changes dynamically, energy is lost due to substrate charging and discharging.

The goal of this study was to obtain optimized VDD and RBB control for a given task and deadline.

We first present a functional mathematical RTSs timing model followed by our power and energy model, and we use parameters extracted from real chips [**?** **?**], to illustrate



(a) Conventional Real Time Execution



(b) Frequency scaling

Fig. 2.2 Conventional real-time execution. (a) The frequency and voltage are set to maximum, so task finishes in a period shorter than the deadline, frequency and voltage remain constant, and power is wasted at the idle region. (b) The frequency scale is set to minimum and remains constant, which allows the task to finish at the deadline, saving power.

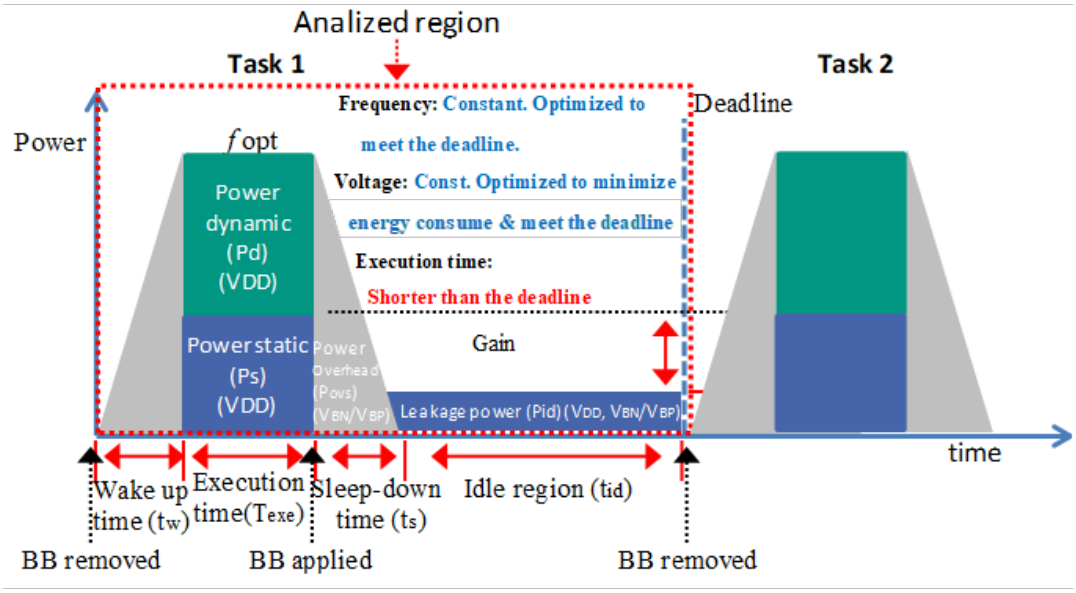


Fig. 2.3 Evaluation. Dynamic Power Management (DPM) with BB, which works at maximum frequency. Dynamic and static energy are consumed only at execution time. There are switching overheads and leakage current during idle period.

the energy characteristics of each scenario. The target is a microcontroller consisting of a processing unit and a memory module. Both components are optimized by being separately controlled with different BB. On the other hand, to avoid level-shifter overhead, a common VDD is used for both components.

We give the same BB to the nMOS and pMOS transistors under the assumption that both transistors are designed so that their characteristics are balanced, which normally results in the best performance per energy [**?**]. That is, the following equation holds.

$$VBP = VDD - VBN \tag{2.1}$$

Hereafter, the BB voltage is represented simply as VBN.

2.1.1 Timing model

We define T_{exe} as the execution time of a given critical task, which is executed with *N* instructions. Assuming that each instruction is executed in *CPI* cycles and the clock period is *T*, T_{exe} can be represented as:

$$T_{exe} = N \cdot CPI \cdot T \tag{2.2}$$

Under the RTS paradigm, T_{exe} should satisfy Eq. (5.3):

$$T_{exe} + T_{ovs} \leq D \tag{2.3}$$

where *D* is the given deadline at which the critical task must be completed. For clarity, we assume T_{exe} = *D* in the first scenario. The term T_{ovs} represents the additional time required for acquiring the necessary operational frequency. In other terms, it is the time to establish the necessary VDD and VBN when switching to and from active and idle states. It can be defined as the sum of the wake-up and sleep-down times, t_w and t_s, represented as:

$$T_{ovs} = t_w + t_s \tag{2.4}$$

2.1.2 Power model

Considering the timing constraints, we propose a power and energy model. The ideal power consumption of a VLSI system P_{ideal} is generally defined as:

$$P_{ideal} = P_s + P_d \tag{2.5}$$

where P_s and P_d are the static and dynamic power, respectively, which can be obtained from the following equations [**?**]:

$$P_s = I \cdot 10^{A \cdot VDD + B \cdot VBN} \cdot VDD \tag{2.6}$$

$$P_d = \alpha_{at} \cdot C \cdot VDD^2 \cdot f \tag{2.7}$$

In Eq. (2.6), *I* is the leakage current, and *A* and *B* are coefficients of exponential terms for VDD and VBN, respectively. In Eq. (2.7), α is the switching-activity factor, *C* is the capacitance, and *f* is the minimum operating frequency (minimum frequency required to meet the deadline).

The static energy at the execution time E_s, represented in Figs. 2.2 and 2.3, can be calculated as:

$$E_s = I \cdot 10^{A \cdot VDD + B \cdot VBN} \cdot VDD \cdot T_{exe} \tag{2.8}$$

Using Eqs. (2.2) and (2.7), the dynamic energy E_d can be expressed as:

$$E_d = \alpha_{at} \cdot C \cdot VDD^2 \cdot N \cdot CPI \tag{2.9}$$

By applying the above equations, the total energy consumption for the first scenario is:

$$E = I \cdot 10^{A \cdot VDD} \cdot VDD \cdot Texe + \alpha_{at} \cdot C \cdot VDD^2 \cdot N \cdot CPI \quad (2.10)$$

Furthermore, referring to the second scenario, another portion of energy should be considered. When the task execution is completed prior to the deadline, the system can enter into an idle state. Although the RBB can reduce the leakage current, it is still consumed, as shown in Fig. 2.3. Besides this leakage energy, the BB switching also consumes some energy E_{ovs}, which can be calculated as:

$$E_{ovs} = \int_{t_{si}}^{t_{sf}} VBN(t)I(t)dt \quad (2.11)$$

where t_{si} is the BB-transition starting point and t_{sf} is where it ends. The term *I(t)* is the current flowing in the BB terminal. It is important to mention that only the sleep-down energy was considered. This is due to the fact that the sleep-down energy represents the current charging while the wake-up voltage refers to the current discharge.

As shown in Eq. (2.8), the energy consumption at the idle state E_{id} for the second scenario can be:

$$E_{id} = I \cdot 10^{A \cdot VDD + B \cdot VBN_{id}} \cdot VDD \cdot T_{id} \quad (2.12)$$

The VBN_{id} is the applied RBB and T_{id} is the idle time. Using Eq. (5.3), T_{id} can be calculated as:

$$T_{id} = D - T_{exe} - t_w - t_s \quad (2.13)$$

Finally, considering the overhead caused by dynamic BB control (transition period) and the leakage energy at T_{id}, the total energy is:

$$E = E_s + E_d + E_{ovs} + E_{id} \quad (2.14)$$

2.1.3 Optimal VDD calculation

We optimized the VDD for the active state and applied BB control. First to decide this optimal supply voltage appropriate to each frequency f_{max}, according to the alpha power law, expressed as:

$$f_{max} = F \cdot \frac{(VDD - Vth)^{\alpha}}{VDD} \quad (2.15)$$

where F is a coefficient related to frequency and $1 \leq \alpha \leq 2$ is the saturation coefficient, which is equal to 2 in the case of the SOTB technology [**?** **?**]. From the above equation, the optimal VDD at the operational state can be expressed as [**?**]:

$$VDD = \frac{(V_{th} + \frac{f_{max}}{F}) + \sqrt{(V_{th} + \frac{f_{max}}{F})^2 - 4V_{th}^2}}{2} \tag{2.16}$$

Some of these parameters are obtained from real-chip evaluation, and the others come from the characteristics of the SOTB. The details of obtaining the parameters are described in the following chapter 3.

Chapter 3

Target Systems and Methodology

In this chapter, we introduce the target systems we use to evaluate the energy and timing model introduced in chapter 2. Additionally, we present the characterization of its principal energy features and electrical coefficients.

The proposed models can be applied to any RTS. Nevertheless, to evaluate its efficiency, we used V850 E-Star (V850), Geyser-SCM, and MuCCRA-4 as the target systems.

3.1 Target system (1) V850 E-Star: Microcontroller

V850 E-Star, a photograph of the chip is shown in Fig. 3.1, is a high-performance low-power 32-bit RISC microcontroller for car electronics, digital signal processing, and digital servo-motor control. It is composed of a five-stage standard pipeline with 46.2-k gate logic cells and 128-kb instruction/data memory modules [**?**]. Fig. 3.2 illustrates the block diagram. The chip used was implemented with LEAP 65-nm FD-SOI SOTB technology. Chip measurement was done with an evaluation board, as illustrated in Fig. 3.3. The VDDs can be statically altered using DC-DC converters, and the V850 state can be controlled using the attached field-programmable array (FPGA). The V850 basically executes one instruction per clock cycle; hence, CPI = 1. The V850 contains a processing core and on-chip memory. These two components have different timing and power characteristics; thus, different BB voltage terminals, called VBN and VBP for the core, and VBNM and VBPM for the memory, respectively.

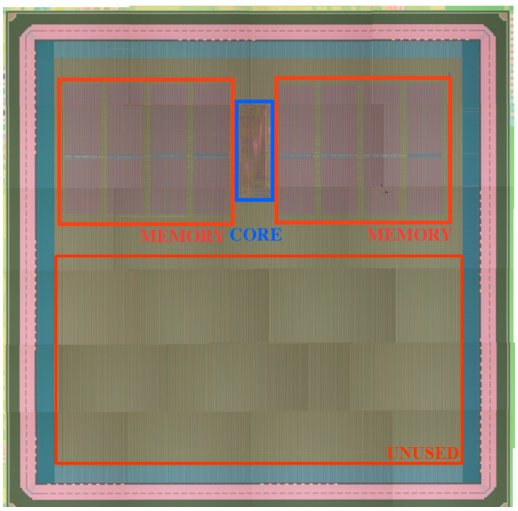


Fig. 3.1 Chip photograph of V850E-Star microcontroller.

Table 3.1 V850 Coefficients for proposed power model.

Parameter	Core	Memory
I	0.2587×10^{-3}	3.0523×10^{-3}
A	0.51921	0.45172
B	1.7926	2.1563
F	6.6641×10^{8}	6.8350×10^{8}
$K\gamma$	0.1110	0.0681
$\alpha_{at}C$	0.6247×10^{-10}	1.3669×10^{-10}

3.1.1 Model coefficients

Both core and memory components should be modeled independently; hence, the total energy consumption of the target microcontroller for both scenarios E_{sc1} and E_{sc2} can be represented using Eq. (3.1) for the first scenario and Eq. (3.2) for the second scenario.

$$E_{sc1} = E_{score} + E_{smem} + E_{dcore} + E_{dmem} \tag{3.1}$$

$$\begin{aligned} E_{sc2} &= E_{score} + E_{smem} + E_{dcore} + E_{dmem} \\ &+ E_{ovscore} + E_{ovsmem} + E_{idcore} + E_{idmem} \end{aligned} \tag{3.2}$$

For these equations, parameters I, A, B, $K\gamma$ and $\alpha_{at}C$ can be obtained from real-chip measurements, as shown in [**?**]. Table 3.1 lists these power-model coefficients.

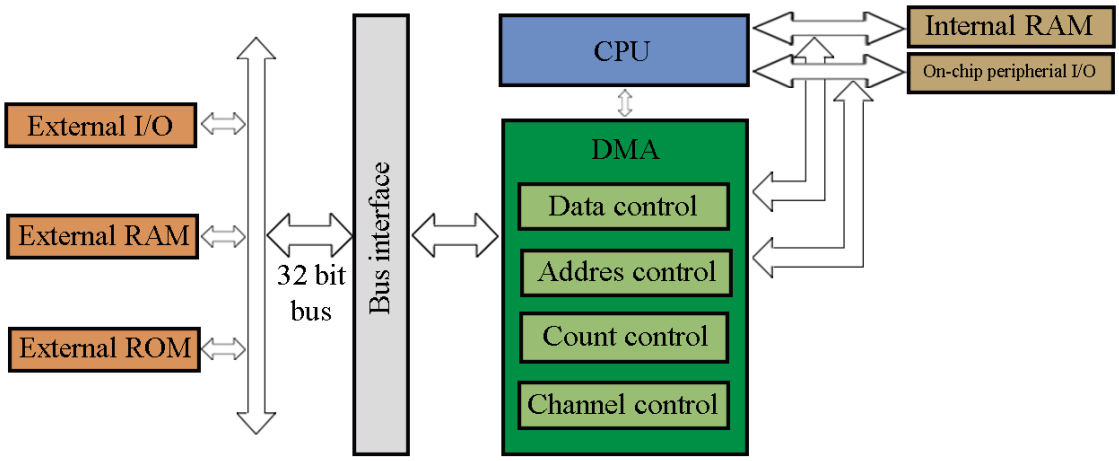


Fig. 3.2 V850E-Star block diagram.

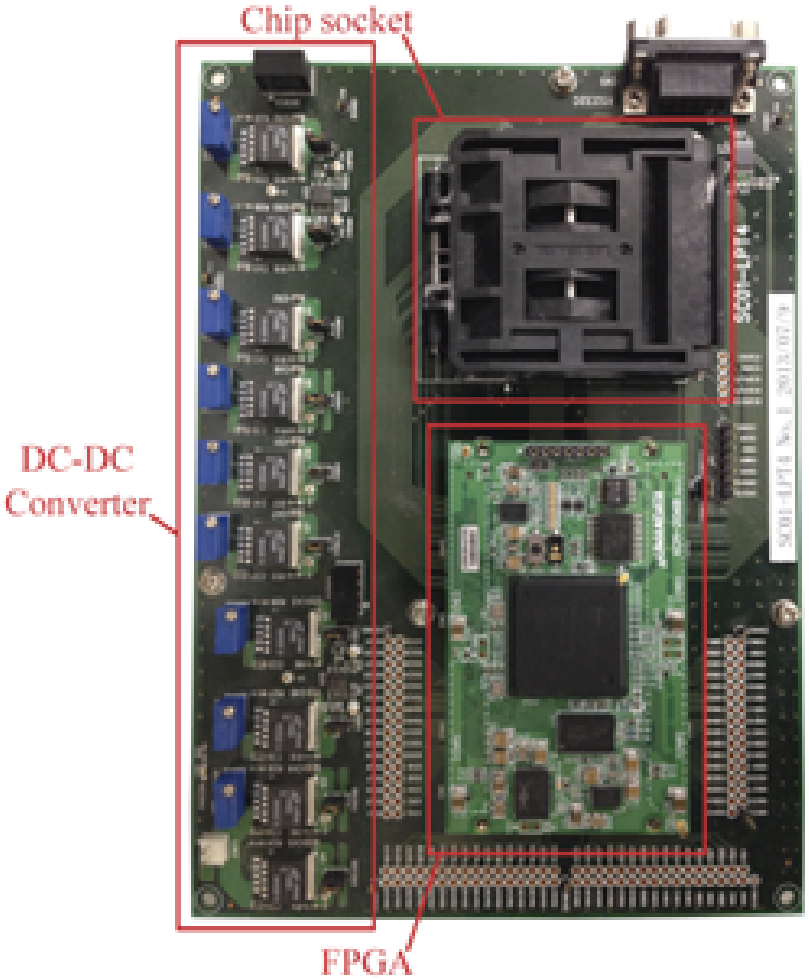


Fig. 3.3 Evaluation board of V850 E-Star.

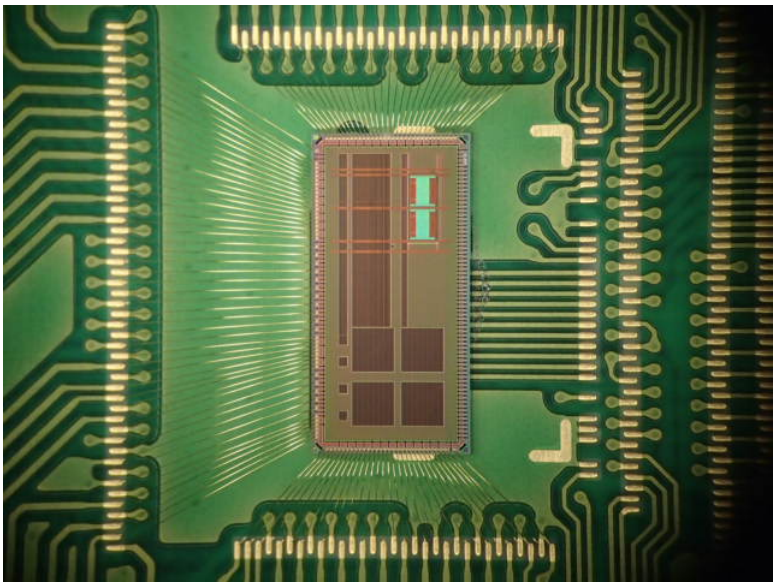


Fig. 3.4 Chip photograph of Geyser-SCM compatible processor.

3.2 Target system (2) Geyser-SCM: A MIPS R3000 compatible processor

Geyser-SCM is a fine-grained run-time BB control mechanism MIPS R3000, a 32bit RISC compatible processor. It is composed of a five-stage standard pipeline with separate instruction cache and data cache both with 4KB 2way set associative mapping. OS with a virtual memory system can be ported using a unified Translation Lookaside Buffer (TLB) with 16 entries [**?** **?**]. Unlike a previous implementation Geyser [**?**] which provides run-time power gating mechanism, Geyser-SCM relies its leakage reduction on the body biasing of the SOTB process. The chip used was implemented with SOTB 65-nm technology. The chip photo is shown in Fig. 3.4. Four bix boxes located in the lower half are for cache memory modules. The specification of Geyser-SCM is shown in Table 3.2.

3.2.1 Model coefficients

The power model coefficients are summarized in Table 3.3.

Fig. 3.5 illustrates the block diagram. Since the appropriate size of embedded memory has not been available in the SOTB process, we made them with a collection of registers.

Table 3.2 Specification of Geyser-SCM

Architecture	ISA	Fully MIPS R3000 Compatible
	Cache	4KB 2 way separated
	TLB	16-entry shared
Chip	Process	LEAP 65nm SOTB 7-metal
	Size	5mm $\times$ 5mm
	I/O	208pins
Tools	Design	Verilog HDL
	Synthesis	Synopsys Design Compiler 2011.09-SP2
	P&R	Synopsys IC Compiler 2010.12-SP5

Table 3.3 Geyser-SCM coefficients for the proposal power model.

Parameter	**Core**	**Memory**
I	0.186×10^{-3}	0.650×10^{-3}
A	0.5096	0.3953
B	2.3805	2.3805
F	6.6641×10^{8}	6.8350×10^{8}
$K\gamma$	8.2874×10^{-2}	6.1342×10^{-2}
$\alpha_{at}C$	0.1771×10^{-10}	0.1771×10^{-10}

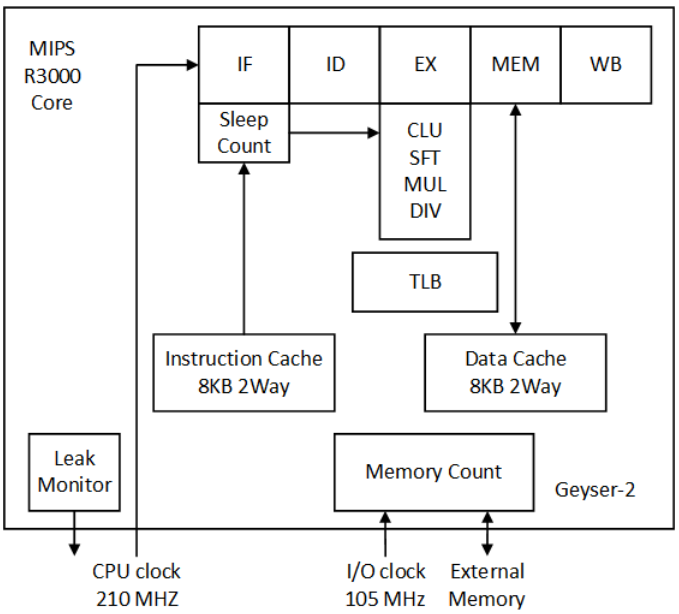


Fig. 3.5 Geyser-SCM block diagram.

It provides a wireless inductive coupling with through chip interface (TCI) on the upper right side of the chip. The TCI technology falls out of the scope of this study, but we refer the interested reader to [**?**] for further information.

Chip measurement was done with an evaluation board, as illustrated in Fig. 3.6.

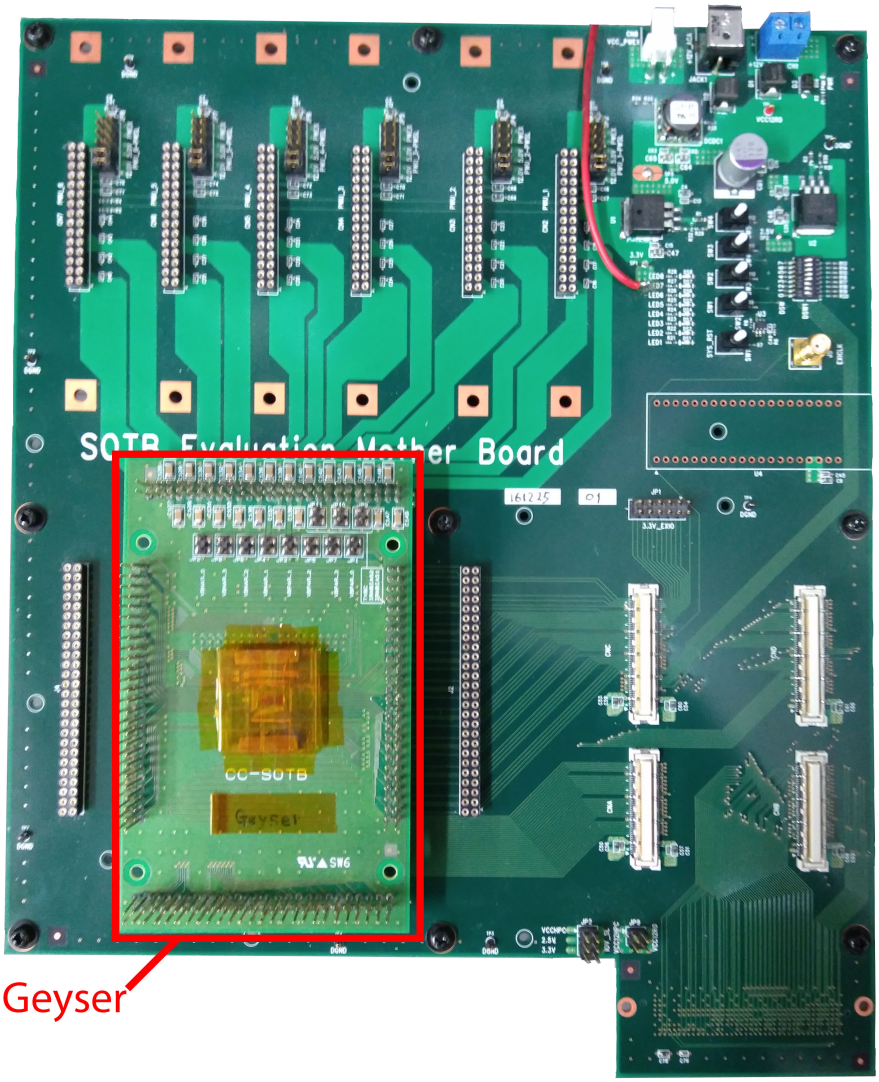


Fig. 3.6 Evaluation board of Geyser-SCM.

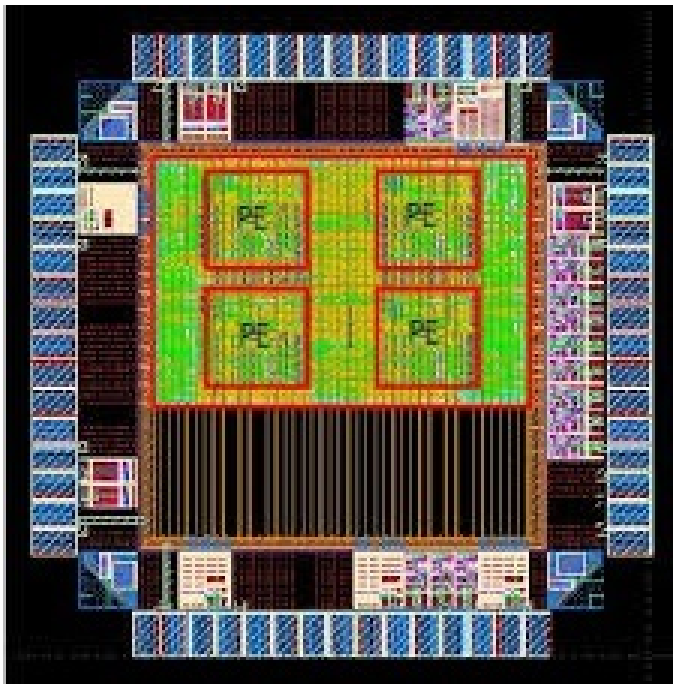


Fig. 3.7 Chip photograph of MuCCRA-4 dynamically reconfigurable processor.

3.3 Target system (3) MuCCRA-4: a dynamically reconfigurable processor

MuCCRA-4 is a prototype of a dynamically reconfigurable processor [**? ? ? ?**]. It is a performance centric dynamically processor which provides tiny vector instructions and pipelined Processing Element (PE) architecture. A photograph of the chip is shown in Fig. 3.7, and as shown in Fig. 3.8, a PE is consisting of a simple 32-bit ALU equipped with simple arithmetic/logic/shift instructions and register file with eight registers. PEs are connected with mesh structure with direct links, and multiplexers for exchange data from other PEs are provided. On the top and bottom of the PE array, data memory modules are provided to store and deliver data to be computed. The configuration data which define the operation in PEs and interconnection between them can be switched to a clock cycle. It has 32-hardware context controlled by the state machine in the context controller. The target MuCCRA-4 (65mm $\times$ 65mm) is implemented with the same LEAP 65-nm process as V850 E-star. For the limitation of the chip size, a 2x2 small PE array was implemented.

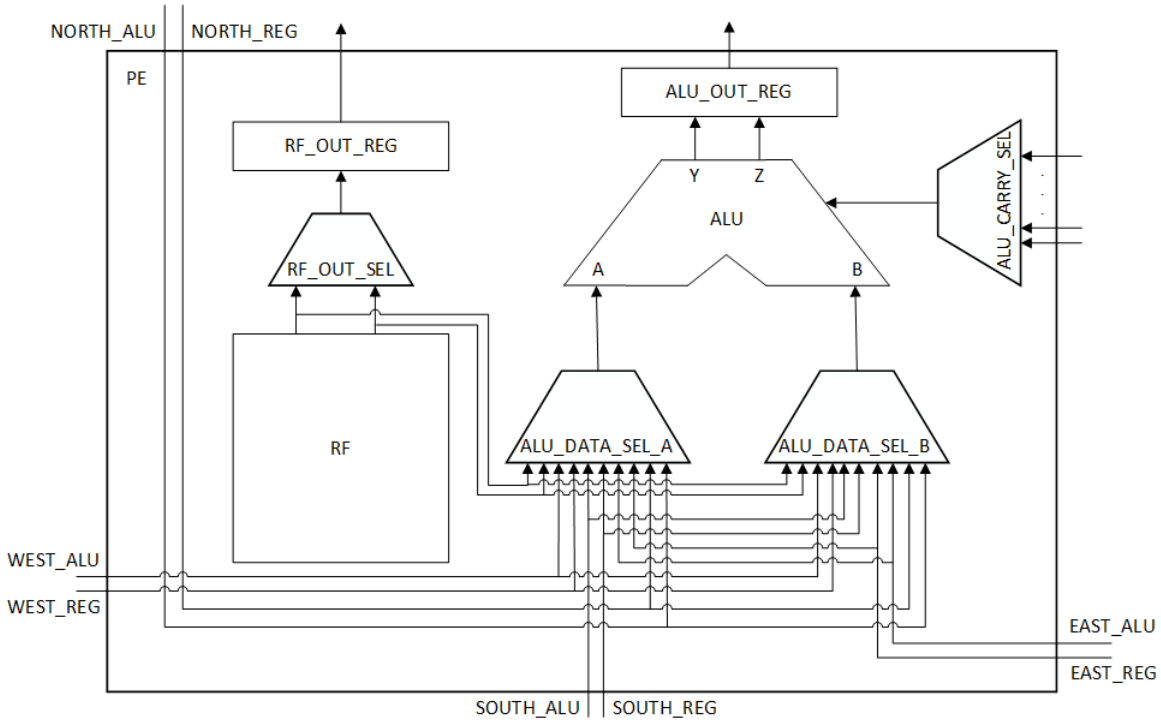


Fig. 3.8 MuCCRA-4 Processing Element diagram.

3.3.1 Experimental set up

To investigate the effect of BB overheads, we conducted a real-chip evaluation with the MuCCRA-4 reconfigurable processor. The chip requires several power supplies to supply body bias voltages to the individual body bias domains. To do this, six four channel power supply boards can be used on the motherboard, where each is controlled by a computer through a USB to UART bridge on the Micro-Zed board. For these experiments, we use Agilent N6507A power analyzer, Agilent 34410A multimeters, and Agilent E3631A power supplies [**?**].

Chip measurement was done with an evaluation board, as illustrated in Fig. 3.9.

3.3.2 Model coefficients

The power model coefficients are summarized in Table 3.4.

The Fig. 3.10 illustrates an evaluation setup for the target systems.

Table 3.4 MuCCRA-4 coefficients for the proposal power model.

Parameter	PE
I	3.2250×10^{-5}
A	0.5648
B	1.9490
$K\gamma$	0.1351
$\alpha_{at}C$	2.1100×10^{-10}

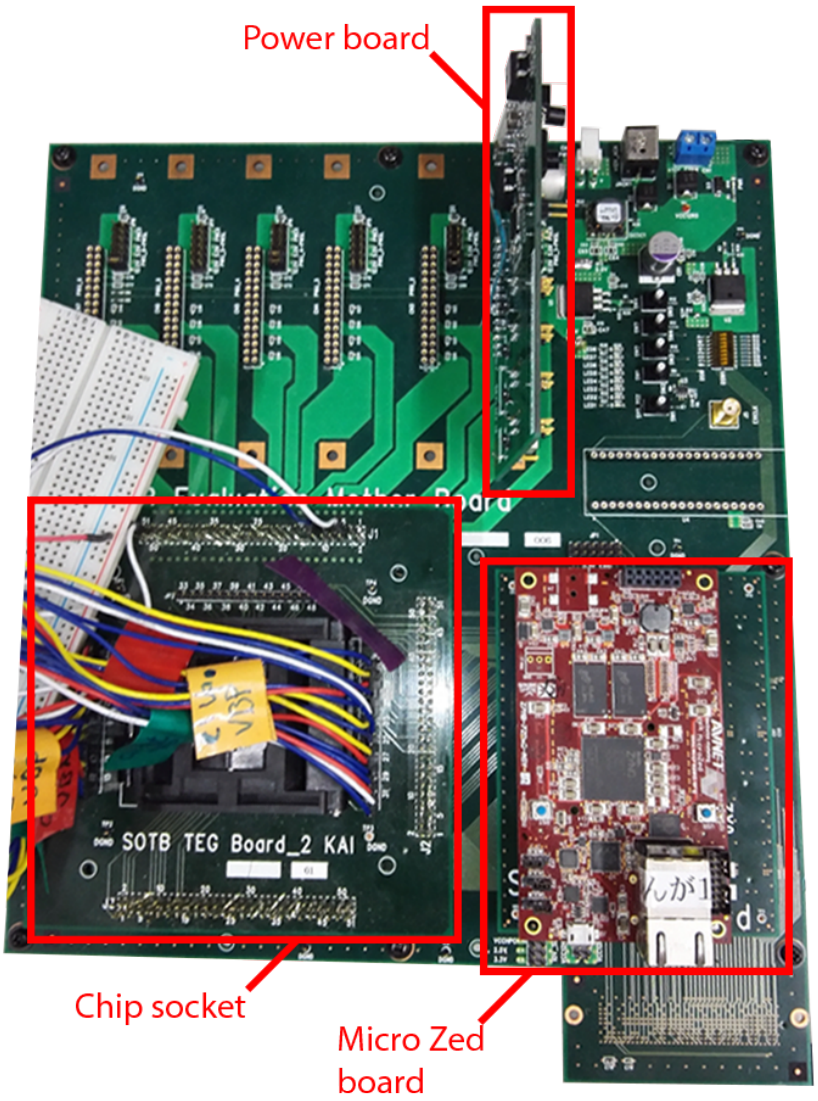


Fig. 3.9 Evaluation board of MuCCRA.

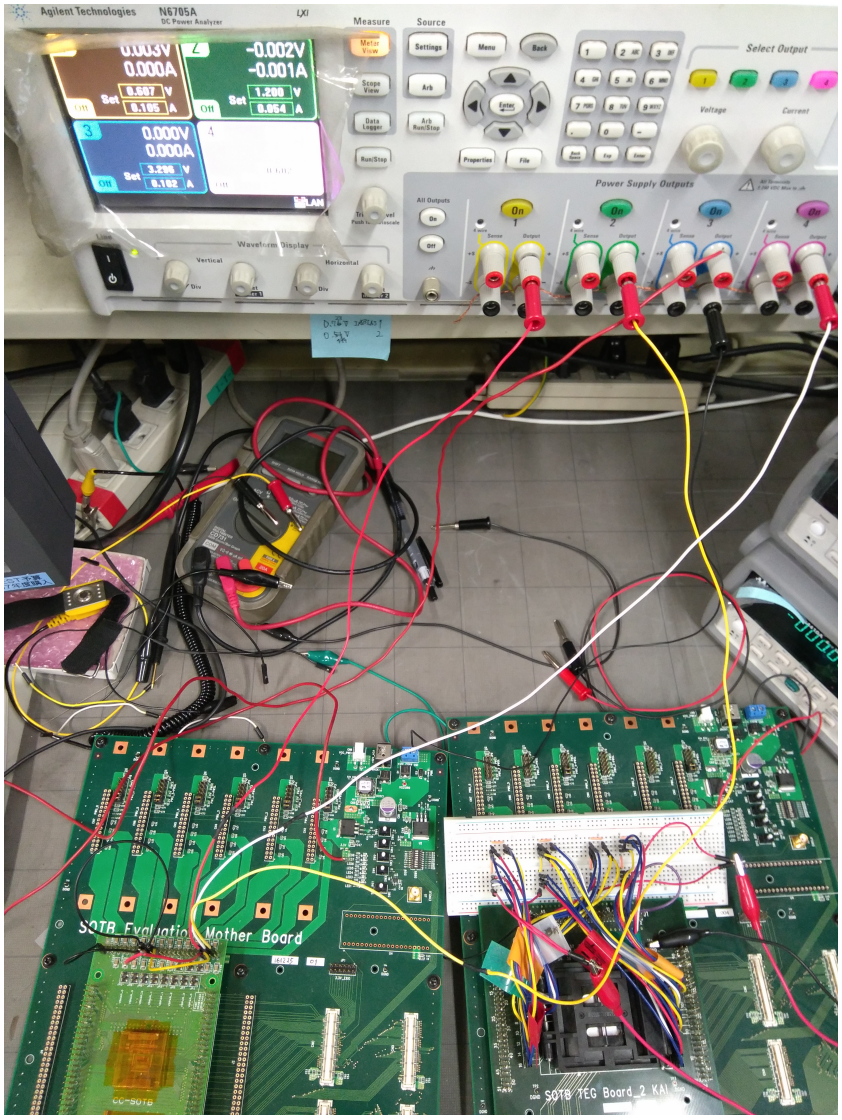


Fig. 3.10 Example of evaluation board setup.

Chapter 4

Energy and Timing Analysis

In the first part of this chapter, we evaluate the target systems mentioned in chapter 3 with the aim of finding the optimum VDD and VBN by using a brute force method. We present an empirical analysis of the energy and timing transition region for Body Bias (BB) control. We measure the timing overhead parameters directly from real chip and analyzed these overhead conditions and their relation to the total energy at the active state. In the second part of this chapter, we discuss the optimal RBB for the second scenario presented in chapter 2 and evaluate its energy reduction.

4.1 Switching region analysis of dynamic BB scaling overheads

4.1.1 V850. Experimental set up

To investigate the effect of BB overheads, we conducted a real-chip evaluation with the V850Estar microcontroller. For these experiments, SG-4322, which is a function generator provided by Iwatsu Electric Co. Ltd., was used as the BB generator. Both VBP and VBN were changed simultaneously. The energy and timing overheads were measured using the Keysight MSOX 4104A oscilloscope and N2820A current probe.

For the timing-overhead measurement, the N2820A was connected between the VDD terminal of the V850 and an off-chip power supply driver for determining whether the effect of the BB is obtained by observing the leakage-current behavior. In this experiment, we defined the timing overhead as the period of the leakage-current transition. Fig.4.1 (a) illustrates the actual behavioral response obtained from this experiment.

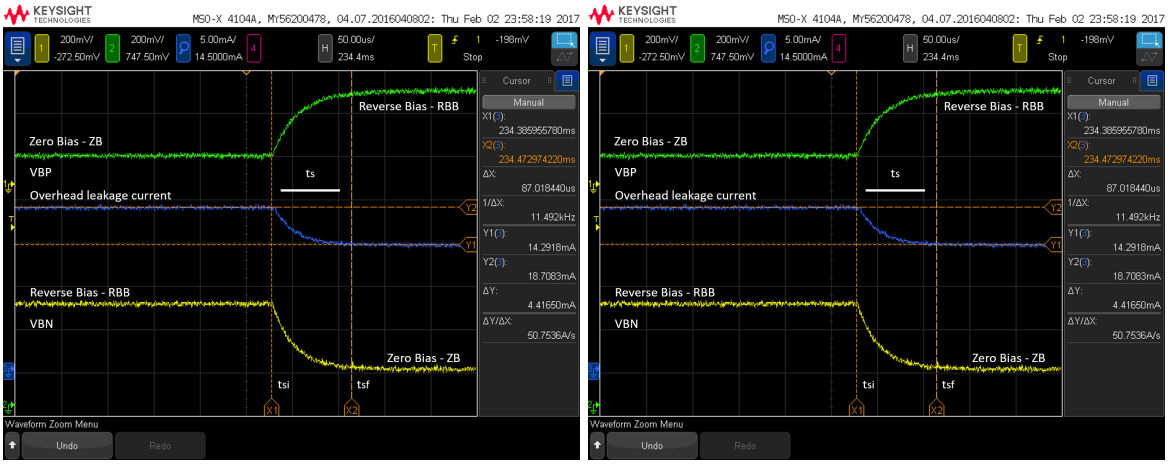
For the overhead-energy measurements, the current probe was connected to the

function generator and BB terminals of the V850. Therefore, immediately when BB was changed, we observed a current spike when charging the well capacitor, as shown in Fig.4.1 (b), which is decreased accordingly, the capacitor charges. This period lasted during the sleep time. We integrate the BB voltage and the leakage current, as shown in Fig. 4.1 (b) by "Leakage current-BB voltage".

In SOTB chips, all I/O pads for the BB are just metal without any resistors or capacitors. Also, all decoupling capacitors on the board are removed so that the power and timing overhead can be measured without any influence from outside the chip. The BB voltage was changed from RBB (ZBB) to ZBB (RBB) voltages. The RBB voltage for pMOS (nMOS) swept from 1300 mV (-700 mV) to 800 mV (-200 mV). We applied the same range of voltages to the core and memory for modeling purposes. However, these voltages were applied separately and analyzed in the same fashion.
We present some of the results of the measurements (maximum and minimum) in Fig.4.2 for the energy of the core and memory of p-body, in Fig.4.3 for the energy of the core and memory of n-body, and in Fig.4.4 for the timing of the core and memory for both p-body and n-body.

4.1.2 V850. Body-bias energy-transition evaluation

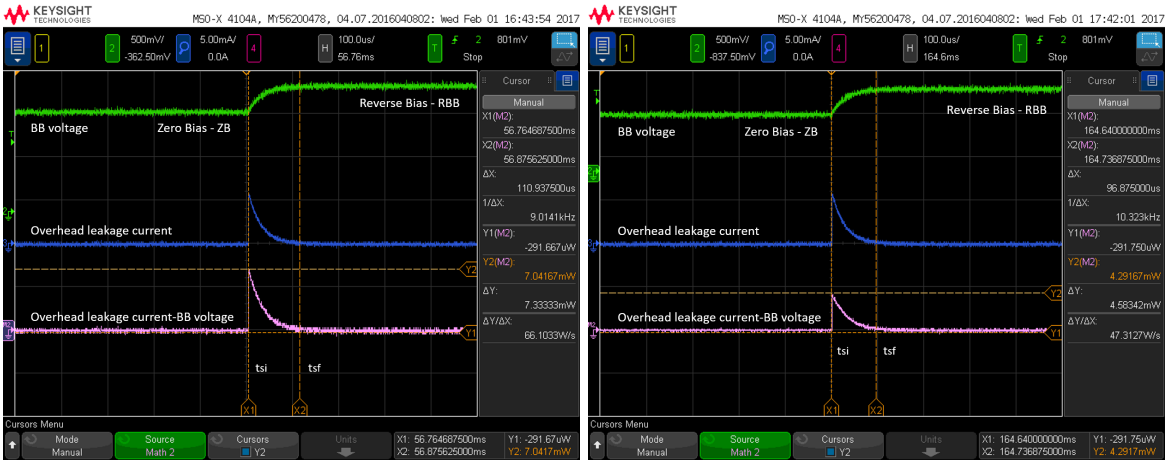
As previously mentioned, since the target system consumes energy when applying the RBB, only the sleep-down transition was considered for analysis. The results are shown in Fig. 4.5 for VBP and Fig. 4.6 for VBN. From these graphs, we can observe that the amount of charge for pMOS is larger than that of nMOS. According to the well structure of SOTB, the pMOS (nMOS) was formed on the n-well (p-well), as previously represented in Fig. 1.1. This means that the p-well has a larger area of the p-n junction and larger capacitance. In fact, in Fig. 4.5, the pMOS shows the maximum value of 400 nJ and a minimum value of 190 nJ. While in Fig. 4.6, the nMOS shows the maximum of 210 nJ and minimum of 47 nJ. The energy of pMOS is twice that of nMOS at its highest settings and four times the energy at its smallest. Furthermore, there is a clear pattern where the energy overhead decreased when the RBB voltage decreased, having values in the nano-Joule (nJ) order. Finally, when averaging the results of both pMOS and nMOS, the core consumed more energy than the memory modules. The reason for this is that the core shares the BB for all the chip area except the memory region as well as the core region shown in Fig. 3.1. Such an area includes many filler cells and buffer cells, which consume static power.



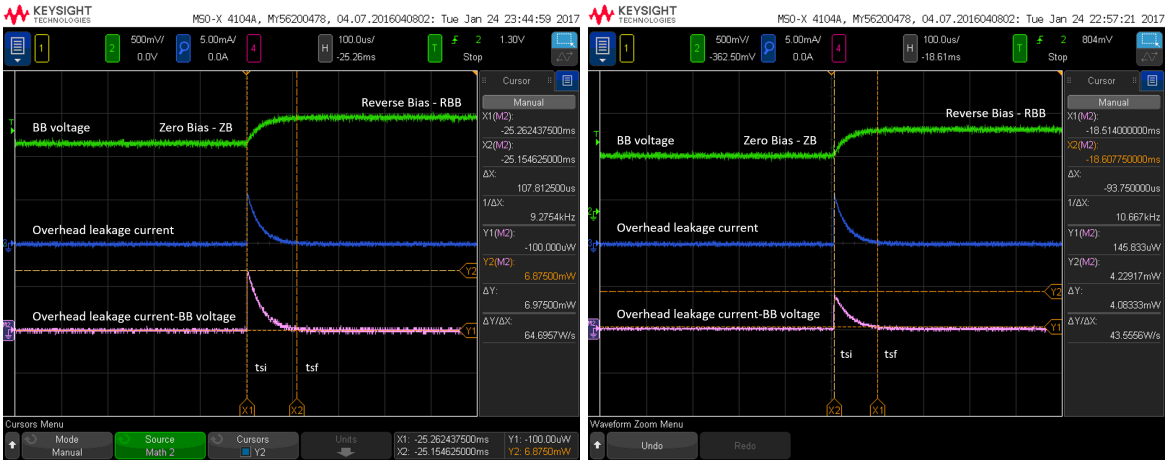
(a) Timing

(b) Energy

Fig. 4.1 Waveforms obtained with real chip: (a) Timing-overhead evaluation. (b) Energy-overhead evaluation.

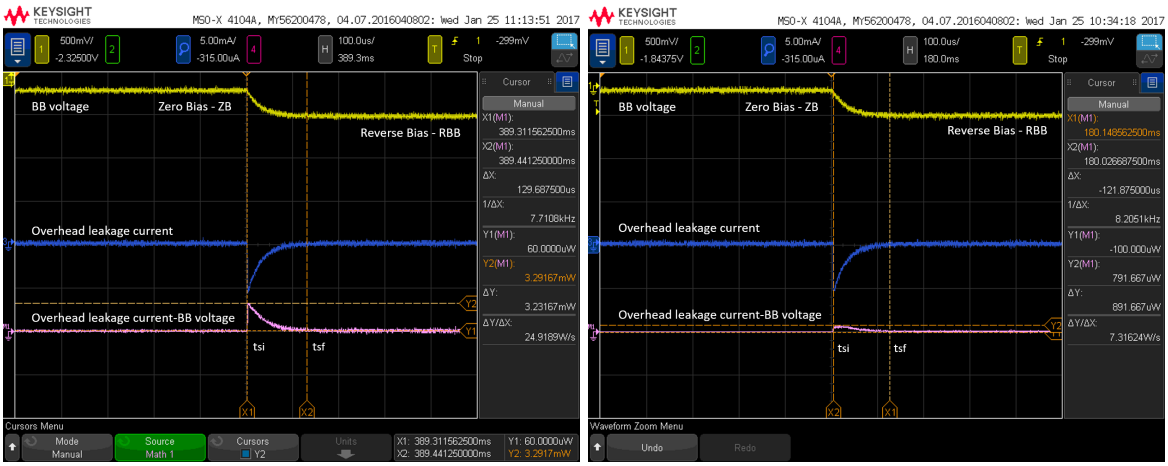


(a) Core p-body waveform Maximum VS Minimum

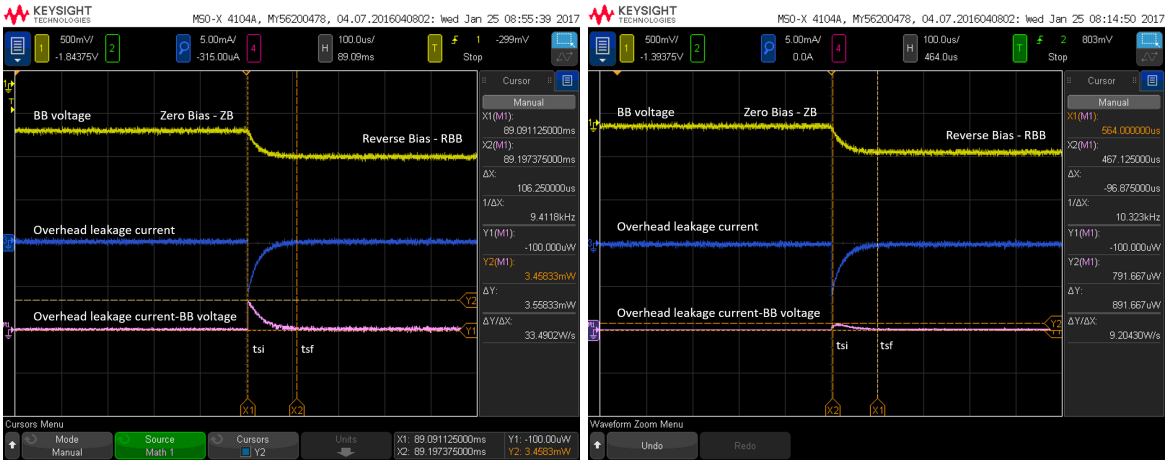


(b) Memory p-body waveform Maximum VS Minimum

Fig. 4.2 Waveforms obtained with real chip; b-body: (a) Core energy overhead evaluation Maximum-Minimum. (b) Memory energy overhead evaluation Maximum-Minimum.

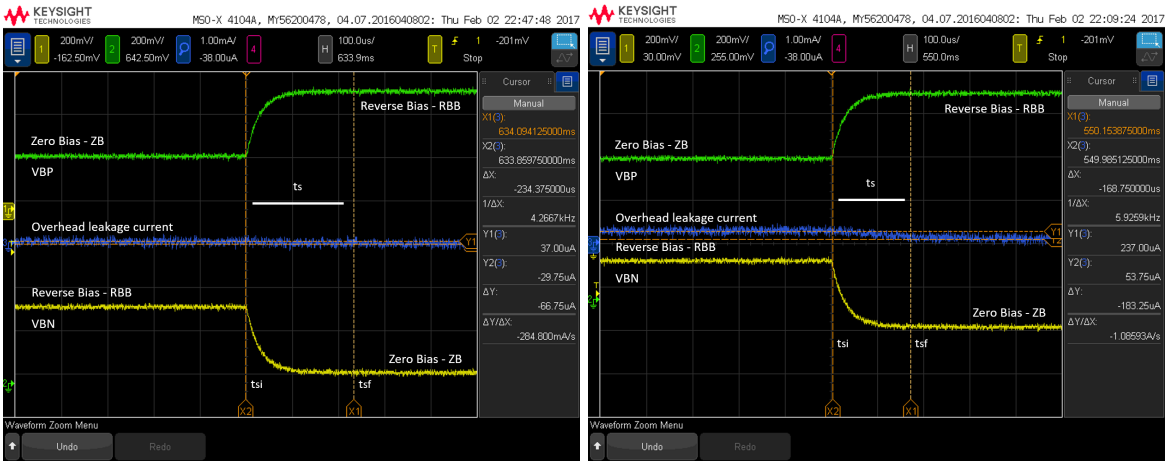


(a) Core n-body waveform Maximum VS Minimum

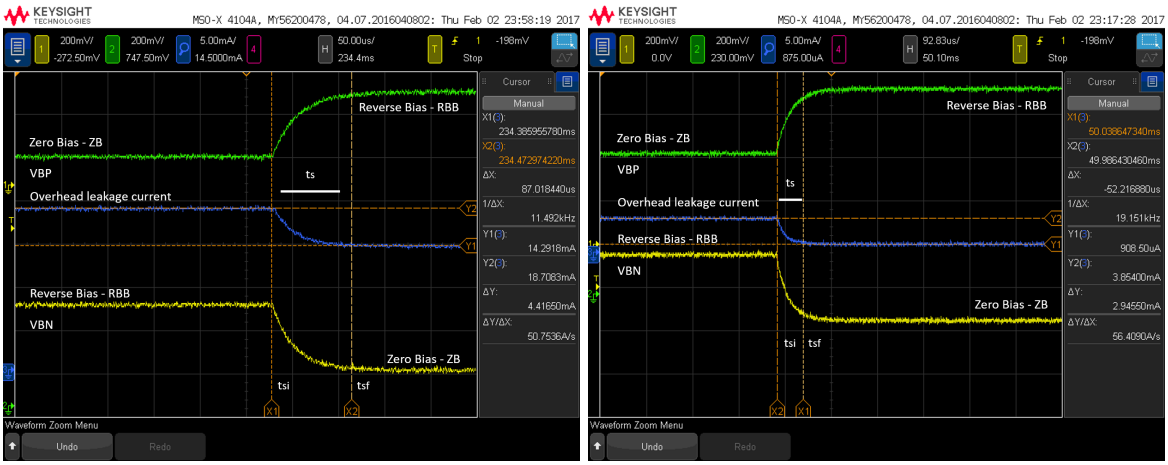


(b) Memory n-body waveform Maximum VS Minimum

Fig. 4.3 Waveforms obtained with real chip; n-body: (a) Core energy overhead evaluation Maximum-Minimum. (b) Memory energy overhead evaluation Maximum-Minimum.



(a) Core p-body & n-body waveform Maximum VS Minimum



(b) Memory p-body & n-body waveform Maximum VS Minimum

Fig. 4.4 Waveforms obtained with real chip; p-body and n-body: (a) Core timing overhead evaluation Maximum-Minimum. (b) Memory timing overhead evaluation Maximum-Minimum.

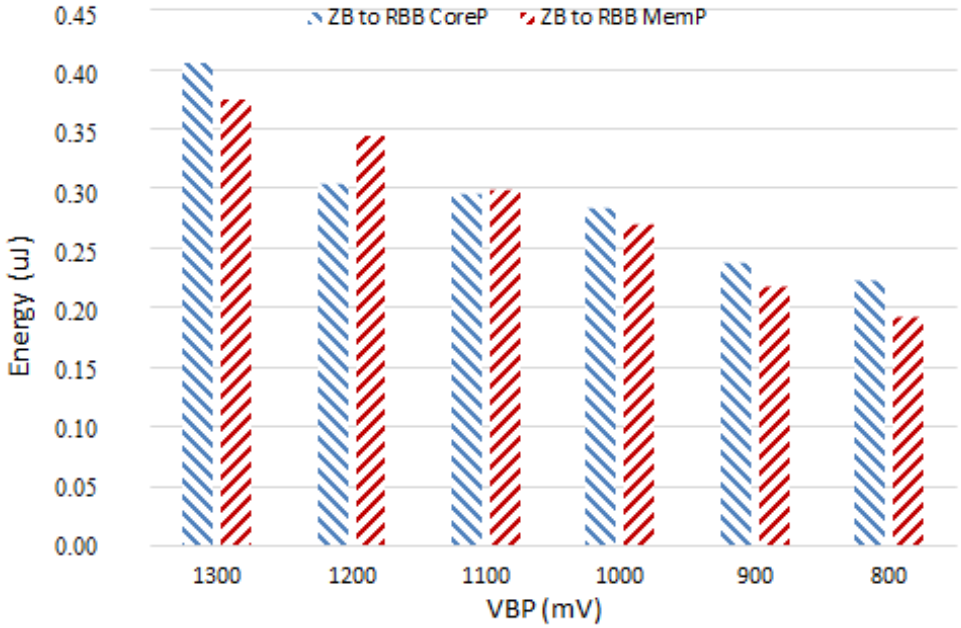


Fig. 4.5 Energy consumption for ZB-RBB transition relationship between core and memory for pMOS transistor.

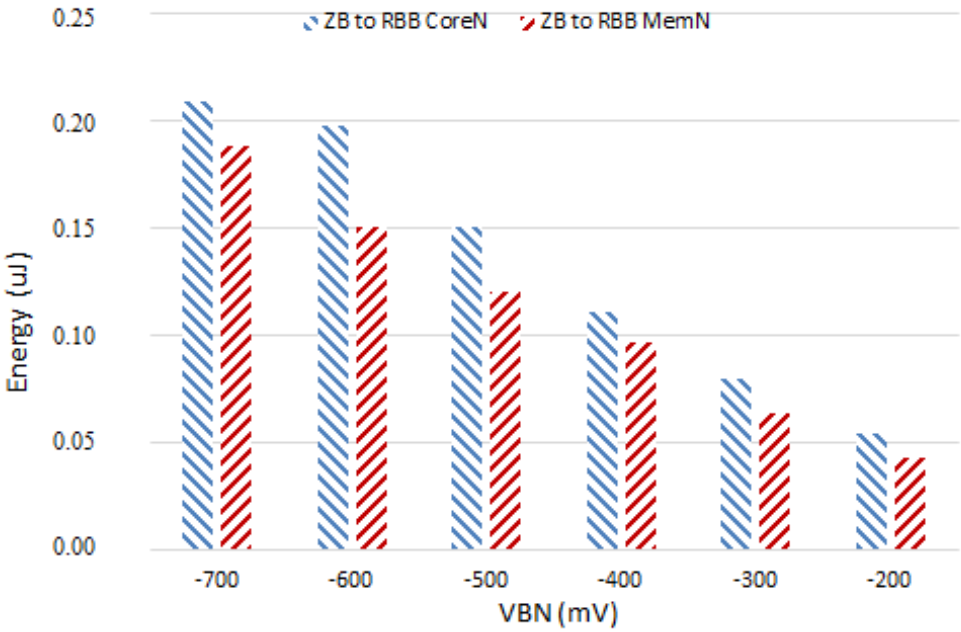


Fig. 4.6 Energy consumption for ZB-RBB transition relationship between core and memory for nMOS transistor.

4.1.3 V850. Break Even Time Analysis

As in our earlier paper [**?**], the efficiency of the dynamic BB scaling can be characterized by the Break Even Time (BET) and can be used as a rule of sum. Devices should enter into the low-power state only when the idle duration is long enough to compensate for the energy overhead necessary to switch to the low-power state [**?**].

The BET is a function of the static power consumed at the active state (P_s), the amount of power consumed during the idle state (P_{id}), and the transition overheads including sleep-down E_{ts} and wake-up E_{tw}, as illustrated in Fig. 2.3 (b). This relationship can be calculated by Equation (4.1):

$$BET = \frac{E_{ts} + E_{tw}}{P_s - P_{id}} \tag{4.1}$$

However since the wake-up energy E_{tw} is for the capacitor discharges, hence, this energy term must not be included and the Equation, (4.1) is simplified as:

$$BET = \frac{E_{ts}}{P_s - P_{id}} \tag{4.2}$$

By evaluating this equation across several voltages, we can determine the BET behavioral region. Therefore, we include in this equation our power-model coefficients (listed in Table 3.1) and shown in Eq. (5.2), measured leakage current, switching activity, VDD, and VBN voltages and incorporate our obtained overhead coefficients. Fig. 4.7 illustrates this working region. We assume a VDD = 600 mV, which is a nominal value.

While a high RBB saves significant static power in the idle state, the switching overhead becomes larger. We can see a bell curve that has its lowest points at -300 mV of VBN and 900 mV of VBP voltages. Under this condition, about 0.23 ms of BET is obtained. However, when averaging, we can observe a trend that a bell curve having lower voltage points from -500 mV to -300 mV, this means around 0.25 ms of BET.

4.1.4 V850. Body-bias transition time evaluation

As in our earlier paper, [**?**], a certain time (timing overhead) is needed for sleeping down or waking up by changing the BB voltage. We measured the BB transition time with the same sleep-down conditions, as mentioned in the previous subsection. The results are shown in Fig. 4.8. The transition timing trended to decrease when BB voltage

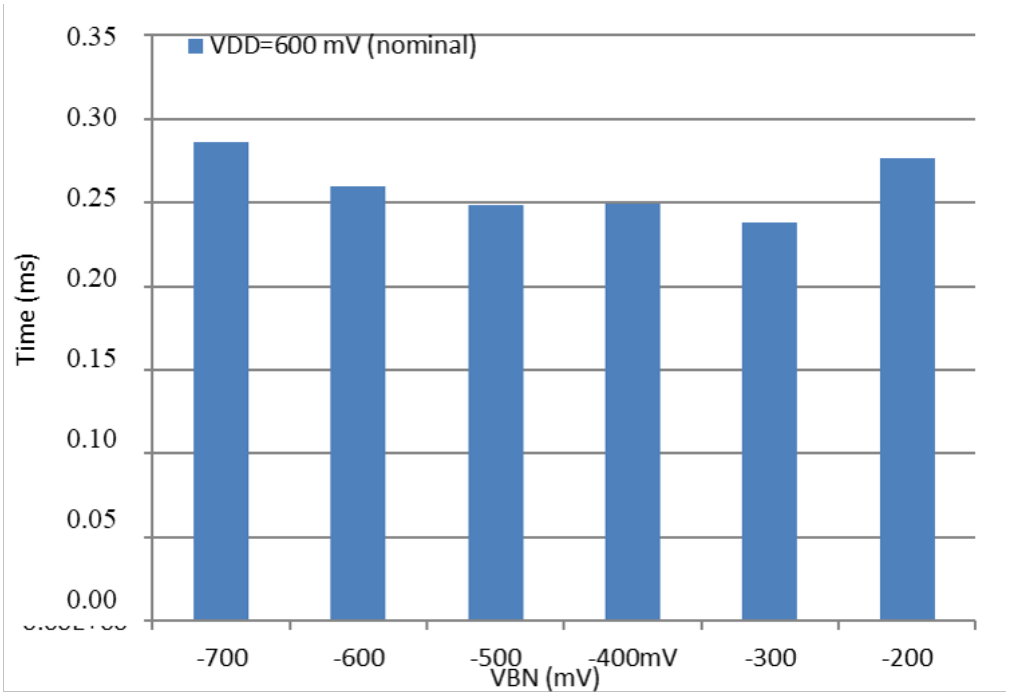


Fig. 4.7 Break Even Time curve, set VDD = 600 mV.

decreased. The memory transition time was about half that for the core; thus, a strategy of only sleeping memory might be advantageous in some situations. We count the slowest transition time as an overhead and assume the worst-case energy saving, that is, leakage is not reduced during the transition. The slowest transition time increases with a large BB voltage; thus, reducing the leakage with a large reverse BB voltage requires timing overhead as well as energy overhead.

4.1.5 Geyser-SCM. Experiment set up

We change VBP-VBN simultaneously. The BB changes to and from Reverse Body Bias (RBB) and Zero Bias (ZB). The RBB voltage for pMOS (nMOS) sweeps from 1300mV (-700mV) to 800mV (-200mV).

4.1.6 Geyser-SCM. Body Bias energy transition evaluation

The period is shown in the screenshot Fig. 4.9 is when BB is changed (green signal), we can observe a current glitch when charging the well capacitor (blue signal) and the leakage energy signal Current-BB (pink signal). In this case, we calculate the E_{ovs} as:

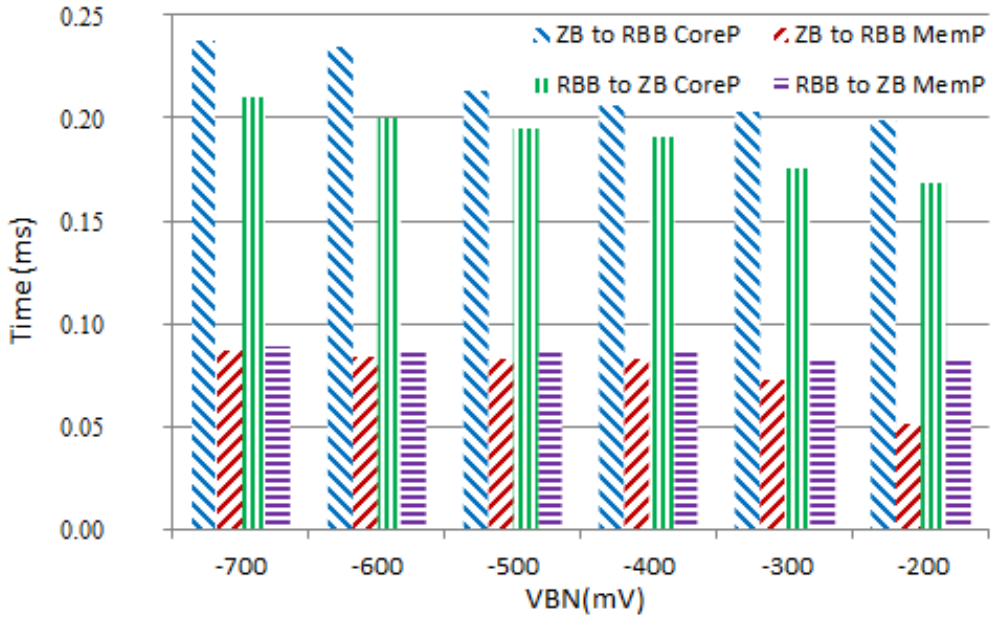
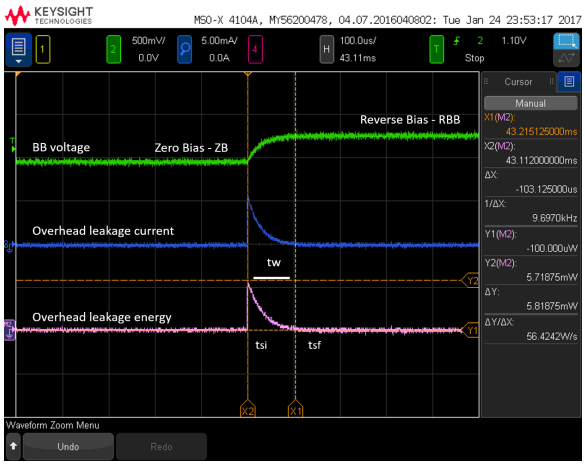


Fig. 4.8 Timing-transition relationship for wake-up (ZB) and sleep-down (RBB).

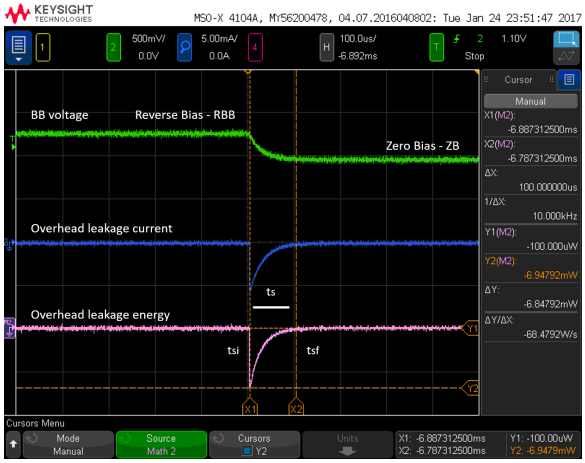
$$E_{ovs} = \frac{VBN \cdot I(t_{sf} - t_{si})}{2} \tag{4.3}$$

where t_{si} is the starting point and t_{sf} is the ending time of the BB transition, in such way it provides the energy overhead (J). This energy is the product of the glitch and the time that takes to settle, as shown in Fig. 4.9 by "Leakage energy" signal. The results are presented in Fig. 4.10 (a) for pMOS and Fig. 4.10 (b) for nMOS.

According to the well structure of SOTB, pMOS (nMOS) is formed on n-well (or p-well). As described in Fig. 1.1, p-well is deposited on the n-well. It means that the p-well has a larger area of the p-n junction and consequently, larger capacitance, as we can corroborate in our target systems, the amount of charge for pMOS is larger than that of nMOS. As for the V850 we can observe there is a trend where the energy overhead decreases when the RBB voltage decreases, having values in the nano-Joule (nJ) order. On the other hand, Geyser-SCM has parameters in the micro-Joule (uJ) order.
We obtain similar results for V850 and Geyser-SCM. However, as worth to note there is a significant difference regarding the energy consumed between the memory blocks of these two microcontrollers. The difference in the energy consumption it is expected because Geyser-SCM is an experimental chip, in the other hand V850 is gate level optimized chip. The difference in the energy consumption it is expected because Geyser-



(a) Wake-up Energy overhead



(b) Sleep-down energy overhead

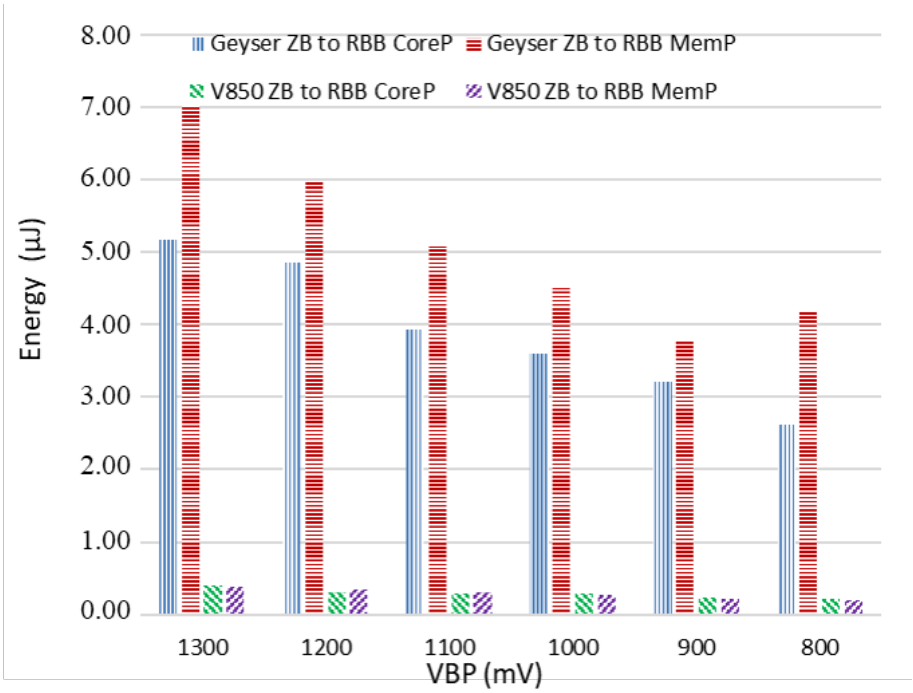
Fig. 4.9 Geyser-SCM Real chip oscilloscope screenshot.

SCM is an experimental chip, in the other hand V850 is gate level optimized chip. Geyser-SCM consumes more energy due to use registers as memory and a large multiplier and a divisor and while V850 uses real memory modules.

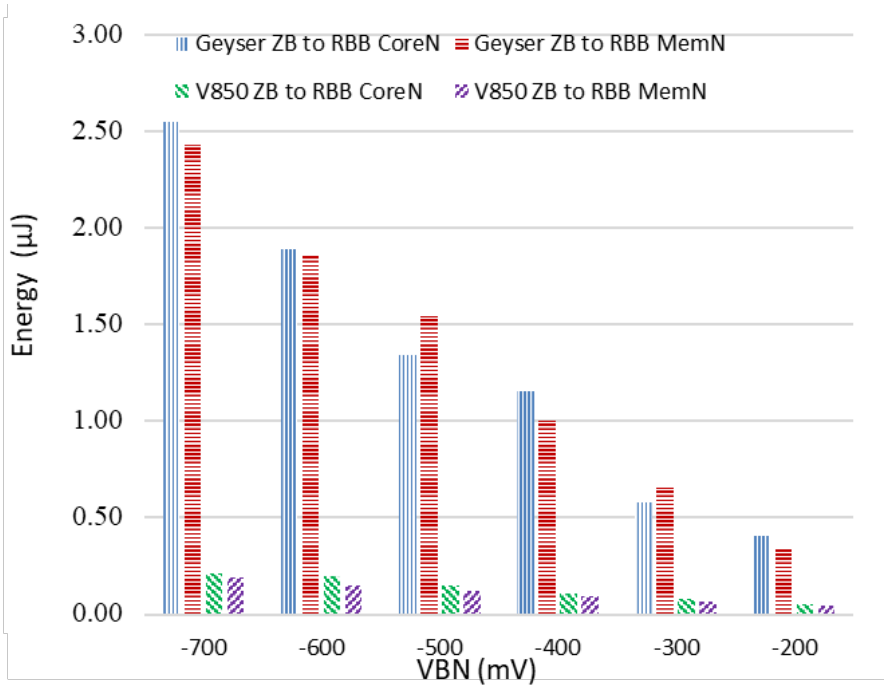
4.1.7 Geyser-SCM. Break Even Time Analysis

Now, with the obtained energy results and transition parameters, we can calculate the BET to efficiently use the dynamic BB scaling.

We calculate the BET by using Equation (5.17) as illustrated in Fig. 4.11 and using the coefficients of our power model (inscribed in Tables 3.1 and 3.3), Equation (5.2), measured leakage current, supplied VDD and VBN, and the switching activity. Fig. 4.12 shows the BET values for V850 and Geyser-SCM when we apply Equation (5.17) to



(a) pMOS component



(b) nMOS component

Fig. 4.10 Energy transition overhead chart for Zero Body Bias - Reverse Body Bias. For pMOS and nMOS components.

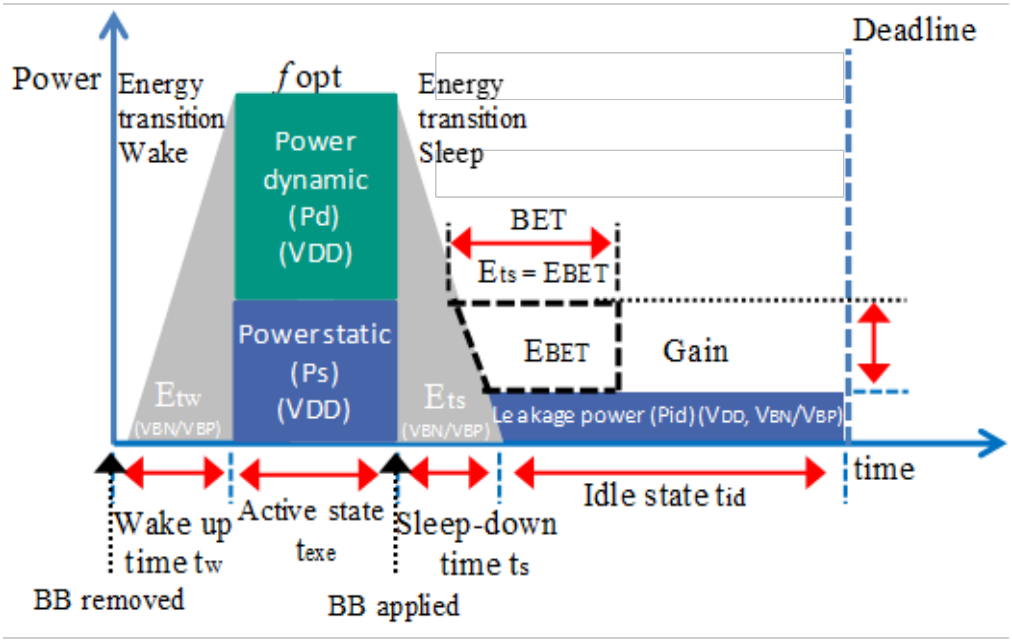


Fig. 4.11 Break Even Time. When the idle duration is long enough to compensate the cost of the energy overhead.

a hypothetical case where the VDD is fixed. We assumed VDD = 600mV, which is a nominal value. While a high RBB saves significant static power in the idle state, the switching overhead becomes larger. We can see a bell curve that has its lowest points at -300mV of VBN and 900mV of VBP voltages. At these conditions, about 0.23ms and 1.81ms of BET are obtained for V850 and Geyser-SCM respectively. The BET can be used as a rule of thumb to select the BB voltage; however, the efficiency of leakage reduction in embedded systems highly depends on the duration of the idle state. On the other hand, faster operational frequency and high VDD are required for a longer idle time. Consequently, there is a tradeoff between the voltage switching overhead and the leakage current and the BET.

4.1.8 Geyser-SCM. Body Bias timing transition evaluation

The timing overhead is the leakage current transition period. The memory block has a shorter transition time than the core (with a minimum of about half the time) [**?**]. Shorter transition times are better for saving energy however, it means a tradeoff with a moderate reduction of energy due to small BB is applied.

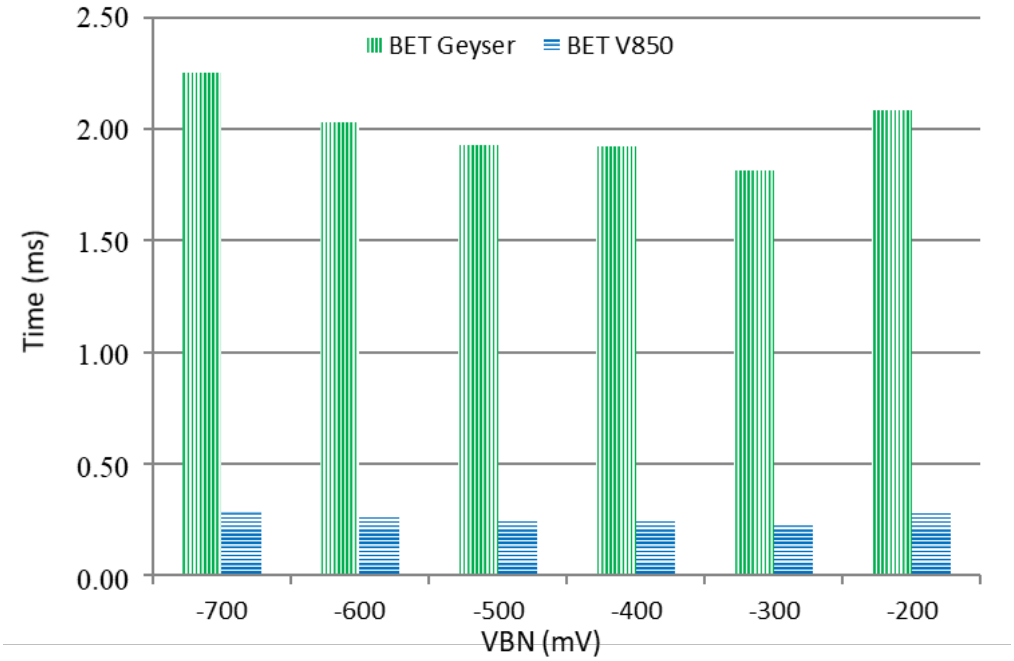


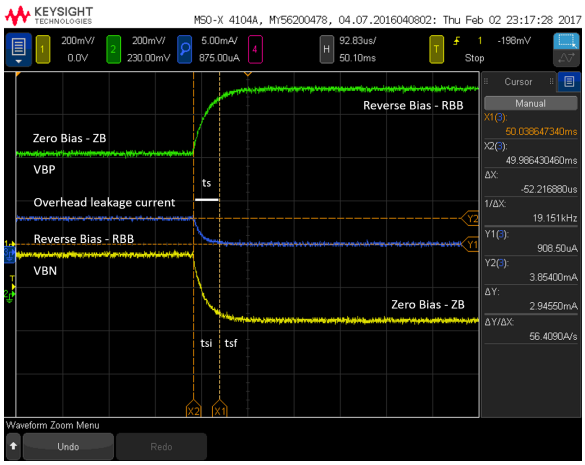
Fig. 4.12 Break-Even Time characteristic behavioral curve. Nominal VDD = 600mV. Analyze using the extracted parameters.

4.1.9 MuCCRA-4. Experiment set up

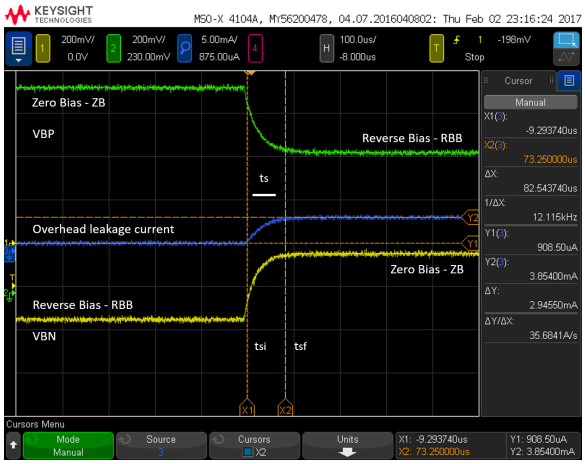
We change VBP-VBN simultaneously. The BB changes to and from Reverse Body Bias (RBB) and Zero Bias (ZB). The RBB voltage for pMOS (nMOS) sweeps from 1300mV (-700nV) to 800mV (-200mV).

4.1.10 MuCCRA-4. Body Bias timing transition evaluation

We defined the timing overhead as the period of the leakage current transition (wake-up sleep-down) by changing the BB voltage. Fig.4.13 (a, b) depicts the actual response, when BB changes: VBP (green signal), VBN (yellow signal), and the leakage current (blue signal). The results are shown in Fig. 4.14 (a) for V850 and (b) MuCCRA-4.
We can observe that the memory has a faster switching timing than the core in both. Regarding the V850 the changing reason between the core and the memory is about 50% when using high BB and up to 75% when using low BB. As for the memory, the discharge reason remains around 50% across all the BB evaluated. Thus, the worst-case scenario is presented by the core from 200us up to 240us, as for the best-case is presented by



(a) ZB to RBB Timing



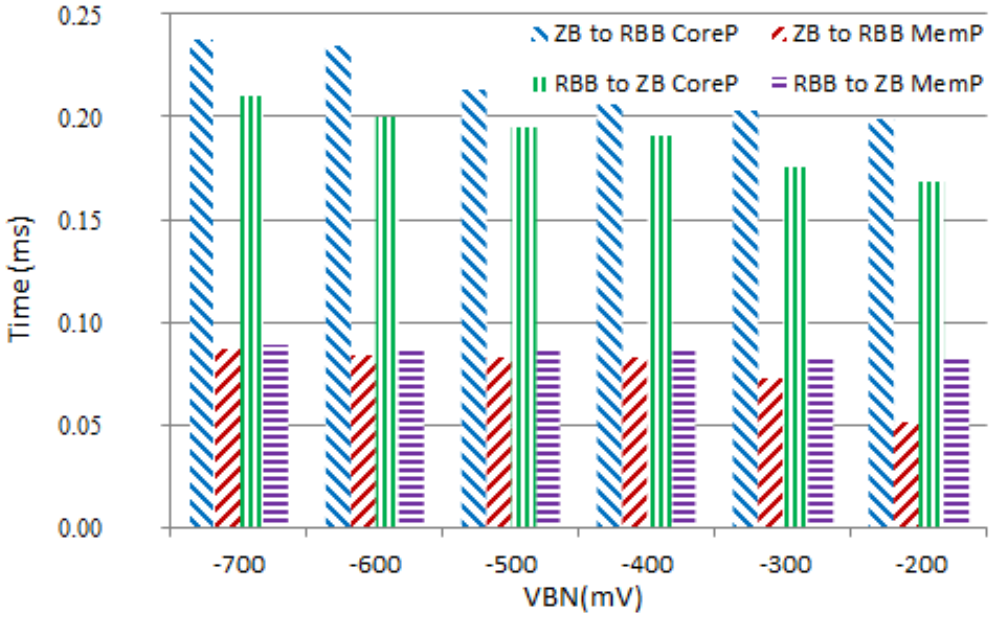
(b) RBB to ZB Timing

Fig. 4.13 Real-chip oscilloscope screenshot. Proof of the measurement taken. Timing overhead: (a) wake-up and (b) sleep-down.

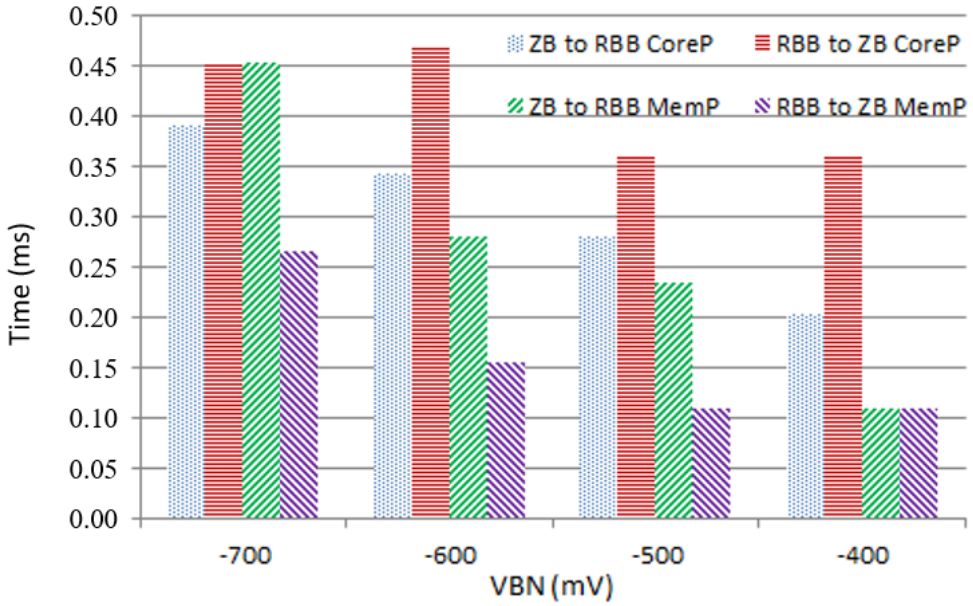
the memory from 52us up to 82us, corresponding to BB voltages.

Regarding the MuCCRA-4, timing is slightly higher compared to V850, however, in this case, we can appreciate the timing decreases as the BB decreases. Nevertheless, the timing order keeps in the same order. In this case, the worst-case scenario is also presented by the core from 390us up to 450us and the best case also presented by the memory around 109us.

The transition timing trend increases when BB voltage increases, thus, saving the leakage with a large RBB requires timing and energy overhead. The core component has the worst-case time scenario. In our evaluation, the RBB to ZB transition is when the



(a) V850 Timing



(b) MuCCRA-4 Timing

Fig. 4.14 Timing overhead conditions graph for Zero Body Bias - Reverse Body Bias transition relationship.

RBB is released, this period should be considered for tasks schedulers and algorithms to schedule the next task. From the evaluation results, we can observe the timing overhead required for SOTB modules is on us order. In contrast, typically deadlines are in the order of ms.

Compared with the PG, the timing overhead of the BB is long. [**?**] suggested that functional units of the PG can turn on and off within 200nsec. The speed of switching can be controlled by the size and the number of transistors used as power switches. On the other hand, the switching time for state transition of the BB control is hundreds of microseconds. It is not dependent on the semiconductor area of the BB. That is, although the area of MuCCRA-4 is smaller than that of V850, the switching speed is longer than that of V850. Since it depends on the layout issues, it seems difficult to control the timing overhead. However, the dynamic BB control has the following benefits compared with the PG.

- Unlike the PG which shuts down the power supply, the data stored in the registers or memory modules are saved in the sleep mode with the deep reverse bias. Thus, an application which needs to keep the data, the BB control is more advantageous.
- The overhead of the BB control is much smaller than the PG, which needs many transistors for power switches and isolation cells.
- The PG generates electric noise on the power grid when the power switches are turned on. For quick switching, the influence becomes especially significant.

These results suggest that the dynamic BB control can be advantageous for the IoT devices , which work with millisecond order intervals and require to keep the data. The difference in the energy consumption it is expected because MuCCRA-4 is an experimental chip, and V850 is gate level optimized chip.

4.2 Effect of dynamic Body Bias scaling

In this section is presented an energy evaluation using a brute force coarse grain search.

4.2.1 Optimal VDD for active state

First, we focus on the active state. As stated earlier, the number of instructions N was determined from the first scenario with *CPI* = 1. Since the V850 includes a single local memory, one instruction is executed in a clock cycle [**? ?**]. Hence, higher operational

Table 4.1 Scenario 2 voltage settings. VDD optimized for given frequency according to alpha power law.

Freq. (MHz)	VDD (mV)	VBN-Active State.
10	304.11	
20	340.99	
30	371.97	Zero Bias
40	403.52	
50	437.84	
60	470.87	

frequencies than that of the first scenario allow the instruction execution of each task to finish prior to the deadline. When a periodic real-time task finishes execution, the system can be put into idle state by the next active state.

The VDDs for each operational frequency are obtained with Eq. (2.16). We compute and use the optimized voltage conditions for VDD that are appropriate to each frequency according to the alpha power law. The VDD is determined beforehand and fixed through all the active and idle periods. We do not change it dynamically due to the high cost of doing so, as described earlier. In the active state, the BB is set to the ZBB. This VDD optimization method does not have any penalties regarding the target microcontroller or the platform [**?**]. These settings are summarized in Table 4.1.

4.2.2 Optimal RBB and power reduction by BB scaling

The BET can ensure static energy reduction; however, we must select the optimal VBN considering all energy combinations in the second scenario. As an exemplification, let us assume a scenario in which the deadline is 3 ms. We use Eqs. (2.8), (2.9), and (2.12) to calculate E_s, E_d, and E_{id}, respectively. For E_{ovs}, we simplify the use of Eq. (2.11) and use the measurements from our evaluation. Fig. 4.15 shows the results of this evaluation. It describes the change in the energy consumption with various VBNs for such a deadline. The horizontal line is the optimized energy of scenario 1, which works at a 10 MHz clock frequency. For a large frequency corresponding to a short active state, a strong RBB is advantageous. However, as we have a tradeoff between switching power and operational frequency, 60 MHz of operational frequency cannot be the optimal point. In this figure, the best reduction ratio was achieved with -500 mV VBN at a 40 MHz clock frequency in the active state. To show the tradeoff in simple terms, the energy efficiency breakdown of the second scenario with a 3 ms deadline is shown in Fig. 4.16 (a).

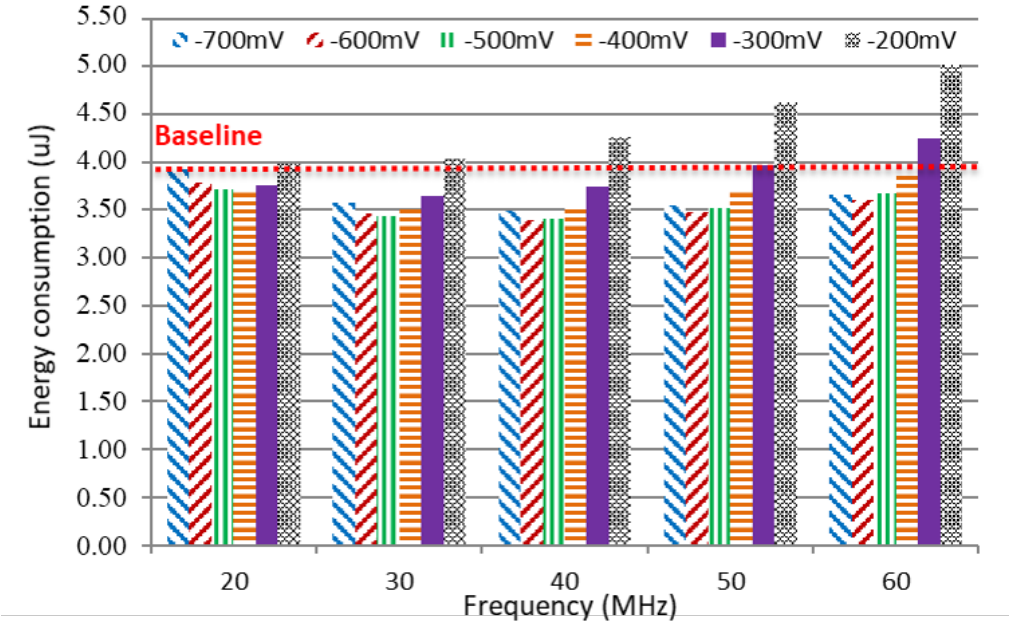


Fig. 4.15 Total energy consumption including energy transition. Scenario 1 vs. scenario 2. Sweep across several frequencies and VBNs. Deadline = 3 ms.

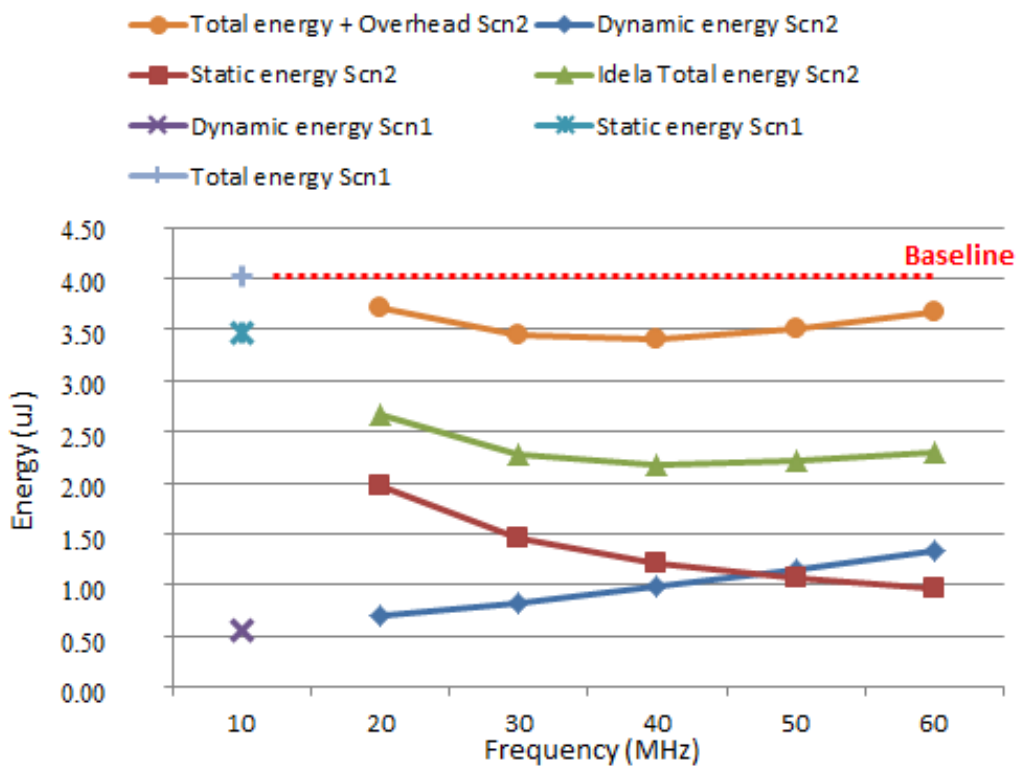
The optimal VBN = -500 mV was used. For a large frequency corresponding to a short active state, the dynamic energy in the active state increased, while the total static energy decreased thanks to the energy reduction in the idle state.

Moreover, the energy breakdown proves that we cannot ignore the energy overhead when designing a system, especially RTSs. In fact, the total energy is almost doubled by the overhead of dynamic BB scaling. However, the dynamic BB is still useful for lowering system energy. In this case, we achieved 15.31% energy reduction when using a 40 MHz clock frequency. The energy saving by dynamic RBB scaling is efficient only when the deadline is long enough since shorter deadlines reduce the idle duration. In fact, at a 2 ms deadline, we achieved only 5% energy saving, as shown in Fig. 4.16 (b).

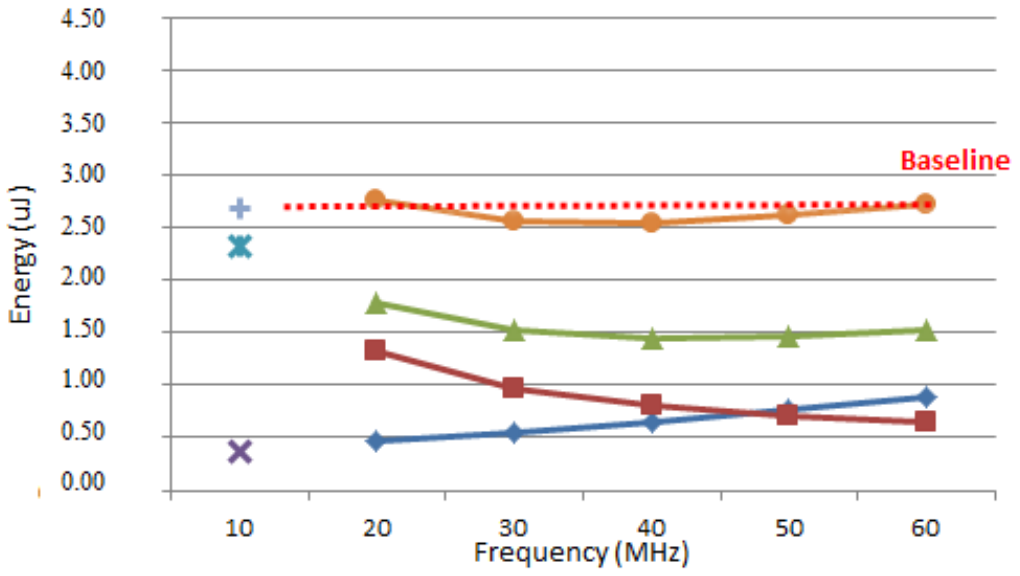
To determine the optimal operating region for VDD and BB control, we applied brute force in the same fashion as described earlier to find the optimal point for a 3 ms deadline (Fig. 4.15). Our brute-force approach involves calculating with a granularity of 1 ms and -100 mV increments for the deadline and BB control, respectively. The outcome is that 40 MHz remains the optimal frequency, regardless of the deadline length. However, for short deadlines, the optimal BB point is -500 mV. As the deadline increases, the optimal point moves to a stronger RBB. As we can see in Fig. 4.17, the decreasing energy rate is not linear, e.g., at around 4 ms, the -700 mV, -600 mV and -500 mV reach a similar

value, from this point -500 mV is no longer optimal. Hence, the next RBB step becomes the optimal value of the region. Increasing the deadline decreases the energy consumed across strong RBBs. Though simple, this method can be practical as a quick reference for design.

For further analysis, energy reduction with various deadlines (2 ms, 3 ms, 4 ms, and 12 ms) is shown in Fig. 4.18. Since the idle time is stretched when a longer deadline is given, stronger RBB can reduce further leakage. At the 12 ms deadline, a stronger VBN (-700 mV) achieved better energy reduction than VBN=-500 mV, which we previously considered as the optimal voltage setting at the 3 ms deadline. About 35% of energy reduction was obtained at the 12 ms deadline.



(a) Deadline = 3 ms



(b) Deadline = 2 ms

Fig. 4.16 Total energy consumption. Comparison between 1st scenario (Baseline at 10 MHz) and 2nd scenario (20 MHz–60 MHz) including overhead conditions. Optimal BB -500 mV.

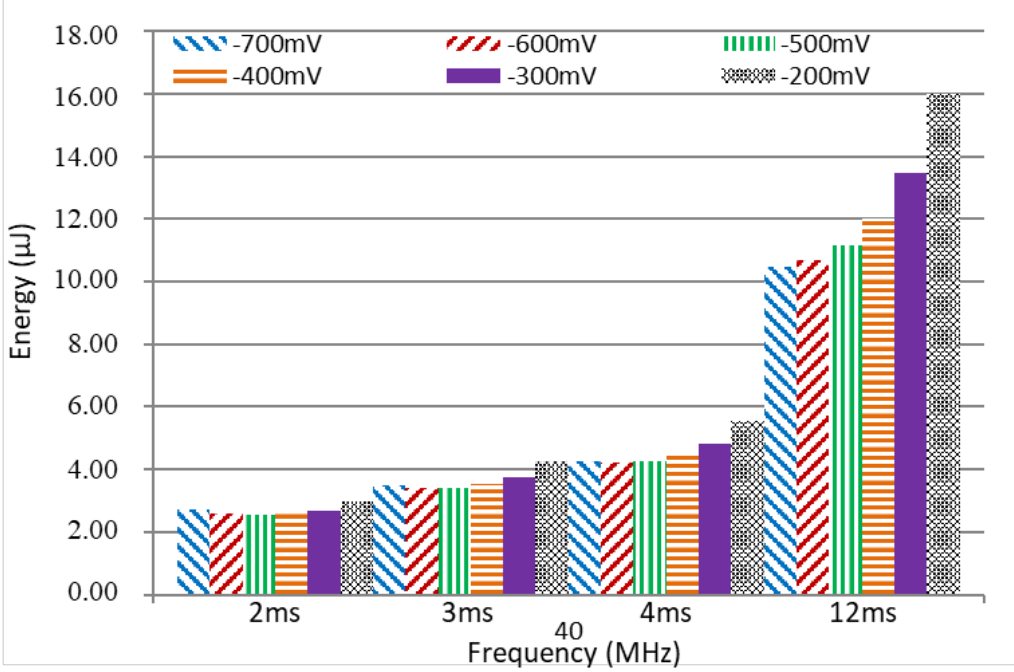


Fig. 4.17 Brute-force results to find optimal VDD-BB control optimal point. For illustrative purposes, we leap from 4 ms to 12 ms to show trend in energy behavior.

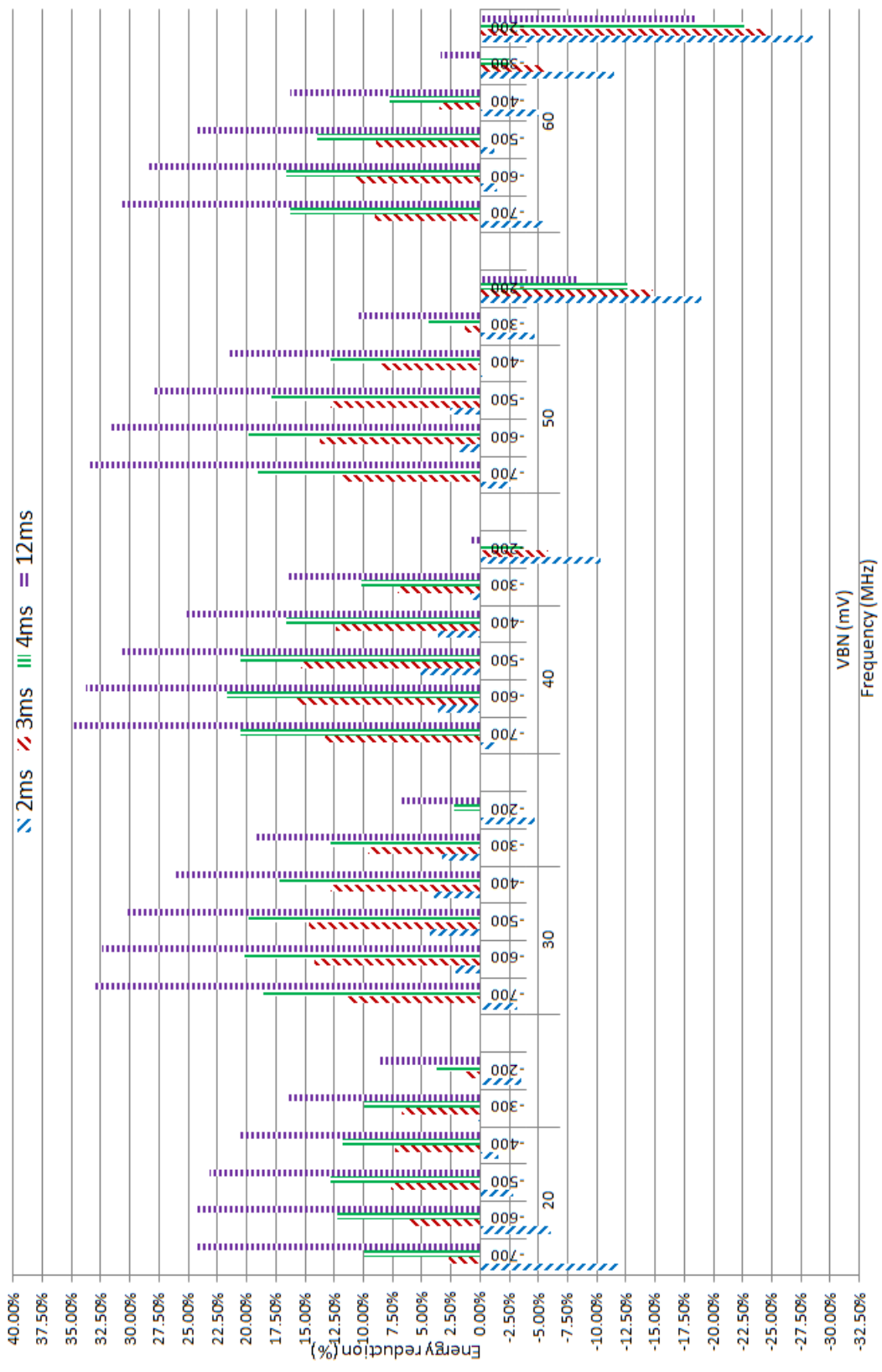


Fig. 4.18 Energy-reduction ratio considering leakage current at idle state across 2 ms, 3 ms, 4 ms and 12 ms deadlines. Each column represents each VBN voltage grouped by frequency.

4.2.3 Accuracy of the model

The models proposed here are based on the timing model in Section 3.1 and the energy model in Section 3.2 and 3.3. The following errors are considerable.

- We expressed the execution time T_{exe} as a simple Eq. (2.2). Since a microprocessor includes the overhead of pipeline stall caused by the cache misses and various kinds of hazards, it is too optimistic in general. However, V850E-star used in this evaluation is a simple micro-controller which provides local memory modules instead of the cache. All instructions and data are preloaded the memory before execution. Also, V850E-star can execute most of the instructions without the pipeline stall, which is one clock cycle. So, we can ignore the error from this part. When more sophisticated processors have treated, this part of the model must be elaborated.

- The base energy E shown in Eq. (5.2) and the maximum frequency f_{max} are based on the model proposed in [**?** **?**]. According to the paper[**?**], the error of the model under room temperature is about 2.7%, yet it can be increased by the temperature variation, process variation and the GIDL (Gate Induced Drain Leakage) effect. Although the GIDL effect was appeared to be less than 1%, the process variation and temperature variation must be compensated to adjust the power supply voltage. The supply voltage adjustment method proposed in [**?**] can also be applicable to the model used here.

All other values used in the study come from the evaluation results from the real chip.

Chapter 5

Optimization

For the Body Bias optimization part, we use Non-Linear Programming (NLP), we use a convex optimization. Convex problems have the property of having a single exact optimum. Interior point methods (IPM) are both theoretically solid and computationally efficient.

In chapter 2, we established a power model and in the chapter 4 we introduced one of the first studies that include real-chip extracted energy overhead parameters.

However, there is no way to mathematically represent E_{ovs} in Eq. (5.2). That is why we could not apply an optimization method to the above expressions. Although there are several models for representing transition behavior [**?**], they are mostly for controlling the supply voltage of PGs.

Hence, we have now devised a standard full switching impulse voltage (double exponential) expression [**? ? ?**], (conventional method in power electronics) a mathematical expression of the transient, for estimating the switching energy. We also devised an interior point method (IPM) based on our power model that can be used to obtain optimality in nonlinear programming (NLP).

5.1 Proposed Energy Consumption and Overhead Calculation Models

5.1.1 Baseline Model

As we mention in chapter 2, we developed a power and timing model using BB control. It is based on real-chip measurements of leakage current, switching current, and maximum operating frequency. We measured the target chip at 25℃. We again assume that the

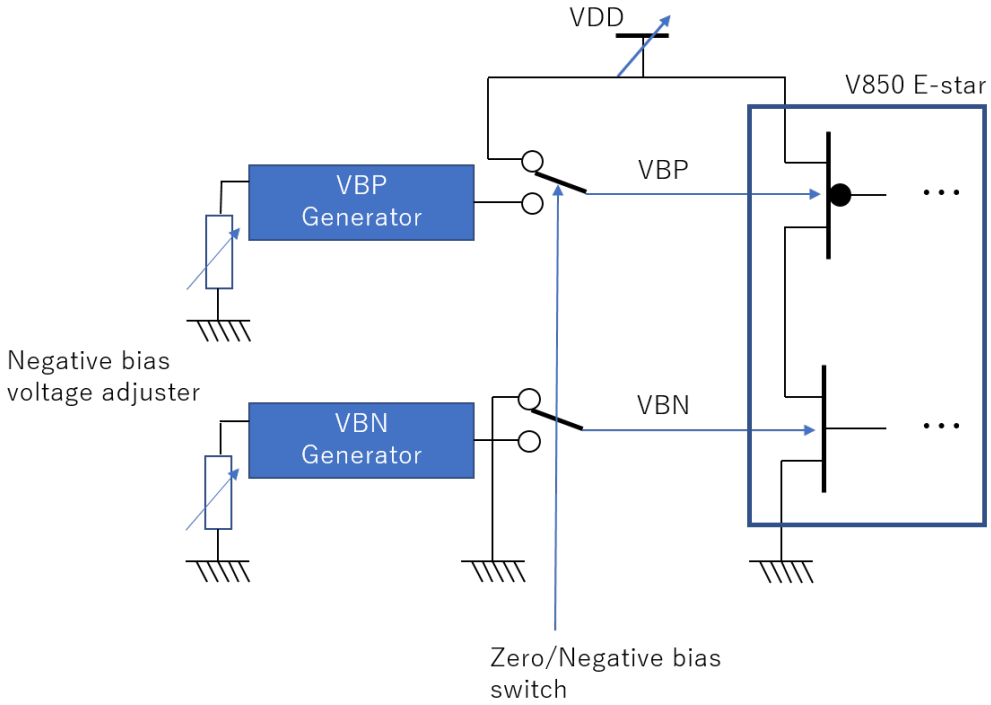


Fig. 5.1 Device-circuit level connection for the target system: V850 E-star. VDD is adjusted to the optimal value for each target application and deadline, and not changed during the execution. VBN/VBP are also adjusted to the optimal BB value, and switched dynamically to/from zero bias.

execution time of the target task is fixed and can be estimated, as described in a previous paper [**?**]. Here, execution time is represented as T_{exe}. The total energy (E_T) is the sum of the static energy (E_s), dynamic energy (E_d), energy overhead of the sleep-down transition (E_{ovs}), and idle energy (E_{id}):

$$E_T = E_s + E_d + E_{ovs} + E_{id}. \tag{5.1}$$

This equation can be represented as

$$\begin{aligned} E_T = \quad & I \quad \cdot 10^{A \cdot VDD} \cdot VDD \cdot T_{exe} \\ & + \quad \alpha \cdot C \cdot VDD^2 \cdot N \cdot CPI \\ & + \quad E_{ovs} \\ & + \quad I \cdot 10^{A \cdot VDD + B \cdot VBN} \cdot VDD \cdot T_{id}, \end{aligned} \tag{5.2}$$

where I is the leakage current, A and B are the coefficients of the exponential terms for

VDD and VBN, respectively, and $\alpha \cdot C$ is the coefficient of dynamic energy corresponding to the switching activity factor at capacitance C. CPI is the clock cycles per instruction and represents the number of cycles that an instruction needs to be executed; the target system is a V850 E-Star microcontroller (described in Section 5). E_{ovs} is the sum of the VBN and VBP energies when we apply BB (Fig. 2.3). Although this paper focuses on VBN for simplicity, the evaluated value includes both energies. Only the sleep-down energy is considered here since it represents current charging while the wake-up voltage represents current discharging. The Fig. 5.1 illustrates the device-circuit level connection. We must adjust VDD to the optimal value for each target application and deadline. Additionally, we must adjust VBN/VBP to the optimal BB value and switched dynamically to/from zero bias. The energy consumed by a VBN/VBP generator itself is not included in E_{ovs}. Various types (analog or digital that could work at near-threshold region) of charge pump circuits /DACs with various tradeoffs have been proposed for VBN/VBP generators [**? ? ? ?**], and considering the total system including them is beyond the scope of this thesis.

As stated above, the execution time for a given task is defined as T_{exe}; the task is executed with N instructions. The idle time is defined as T_{id}.

Additionally, T_{exe} should satisfy:

$$T_{exe} + T_{ovsT} + T_{id} = D, \tag{5.3}$$

where D is the deadline at which the critical task must be completed, and T_{ovsT} is the time needed to establish the necessary VDD and VBN when switching to and from active and idle states. T_{ovsT} can be defined as the sum of the wake-up and sleep-down times, t_w and t_s:

$$T_{ovsT} = t_w + t_s. \tag{5.4}$$

Some of these parameters are obtained from real-chip evaluation, and the others come from the characteristics of the SOTB device. The details of obtaining the parameters were described in our previous paper [**?**]. However, there was no way to mathematically represent E_{ovs} in Eq. (5.2). That is why we could not apply an optimization method to the above expressions. Although there are several models for representing transition behavior [**?**], they are mostly for controlling the PG supply voltage. Hence, we propose using a double exponential expression, a conventional method in power electronics.

Before going into detail, we show the overall workflow of this study in Fig. 5.2.

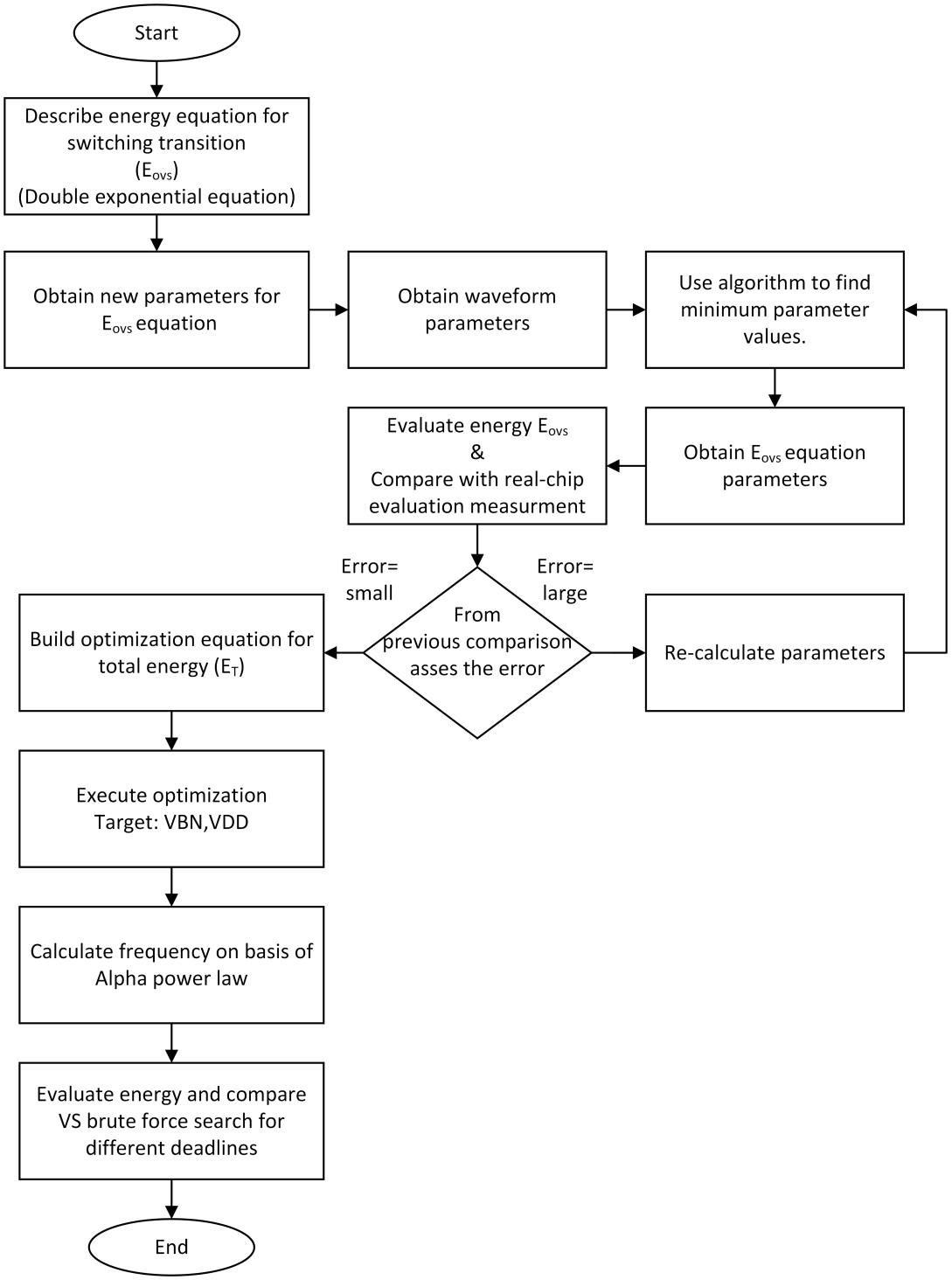


Fig. 5.2 Overall workflow of BB control optimization investigation.

5.1.2 Double Exponential Waveform Expression

We consider an electrical transient to be a temporary disturbance in a power system caused by voltage switching, so a minimal level of transient energy is expected regardless of the circuit. For the transient analysis to model E_{ovs}, we use a widely and well-proven method in power electronics, double exponential equation [**?** **?**]. The electrical transients have the shape of a standard full SI waveform or a double exponential waveform, in our case, this transient occurs, after we finish the task execution and we apply BB as shown in Fig. 2.3. We analyze this transient period from the real chip current measurements. Additionally, to the double exponential expression, we applied Nelder-Mead Simplex algorithm for coefficients calculation [**?**].

SI waveforms are characterized by three parameters: the rise time (t_{rise}), which is the time it takes to reach the maximum current amplitude, the current amplitude (I_{ovs},) and the tail time (t_{tail}), that is the time it takes to settle.

We use these parameters to model E_{ovs} to fit into the double exponential waveform [**?**]. The SI waveform is expressed as

$$I(t) = I_{ovs} \cdot \kappa \cdot (e^{-\gamma t} - e^{-\delta t}), \tag{5.5}$$

where gamma (γ) and delta (δ) are related to the t_{rise} and t_{tail} times, respectively, and kappa (κ) is the amplitude modifying factor used to compensate for interaction between the two exponential terms. The κ factor is related to γ, and δ and can be calculated using Eq. (5.6) [**?**]:

$$\kappa(\gamma, \delta) = (e^{-\gamma \cdot \frac{\ln(\delta)-\ln(\gamma)}{(\delta-\gamma)}} - e^{-\delta \cdot \frac{\ln(\delta)-\ln(\gamma)}{(\delta-\gamma)}})^{-1}. \tag{5.6}$$

Finally, to get the energy overhead, we need to integrate Eq. (5.5) from time 0 to t_s so that E_{ovs} can be expressed as

$$E_{ovs} = VBN \cdot I_{ovs} \cdot \kappa \cdot (\frac{e^{-\delta t_s}}{\delta} - \frac{e^{-\gamma t_s}}{\gamma}). \tag{5.7}$$

Furthermore, we assume that VBN changes instantly between constant values. If the voltage source has an inner resistor, the voltage drop must be taken into consideration. Since some charge pump circuits used in VBN controllers have a large inner resistor, it may need to be considered. Here we assume an ideal battery and a constant VBN/VBP in order to separate the analysis from battery issues.

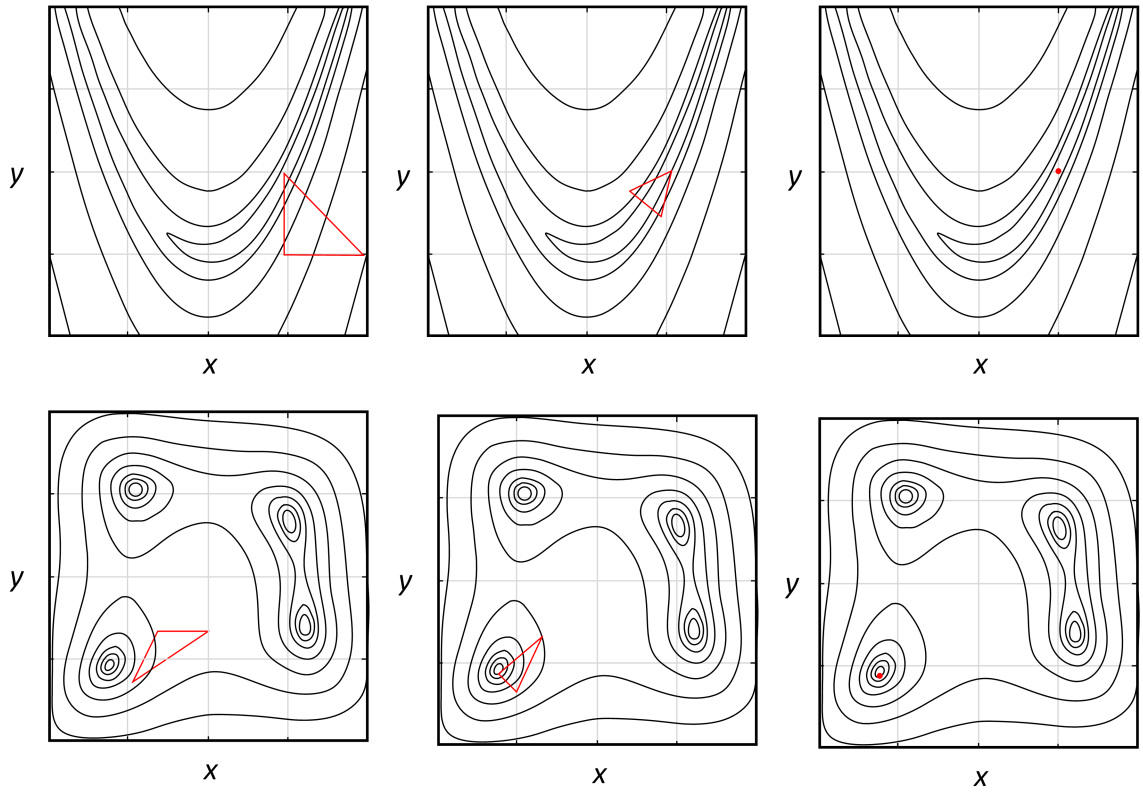


Fig. 5.3 Nelder-Mead Simplex Algorithm. Searching over iterative process examples.

5.1.3 Nelder-Mead Simplex Algorithm

Nelder-Mead Simplex Algorithm is a numerical method used to find the minimum or maximum of an objective function of n variables in a multidimensional space. It is a direct search method that depends on the comparison of function values at the (n+1) vertices of a general Simplex, followed by the replacement of the vertex with the highest value by another point. The Simplex adapts itself to the local landscape, and contracts on to the final minimum. It estimates the Hessian matrix in the neighborhood of the minimum (maximum), needed in statistical estimation problems. This method has been proved very computationally efficient. It is often applied to nonlinear optimization problems for which derivatives may not be known [**?** **?**]. The Fig.5.3 illustrates the iterative progress to search over the minimum of an objective function. It is commonly used with double exponential waveform problems [**?**].

5.1.4 Switching Impulse Waveform Model Coefficients

To find appropriate coefficients for the target chip, we use real-chip measurement results with several predefined values of VBN.

The proposed method for modeling E_{ovs} is based on known physical parameters, t_s and I_{ovs}, and the VBN voltage variation (-200 mV to -700 mV) that we established from our previous evaluation of an SOTB device [**?**]. First, we use the Nelder-Mead algorithm to calculate the analytical function parameters of the SI waveform (γ, δ, and κ) from the known measurements (t_{rise} and t_{tail}) of t_s [**?**]. The following approximations are used to initiate the algorithm:

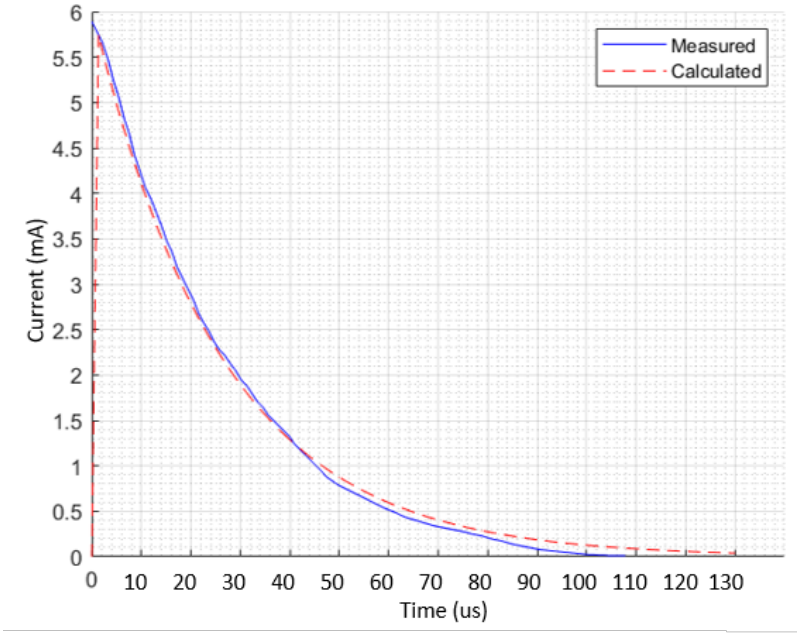
$$\gamma = \frac{1}{t_{tail}}, \tag{5.8}$$

$$\delta = \frac{1}{t_{rise}}. \tag{5.9}$$

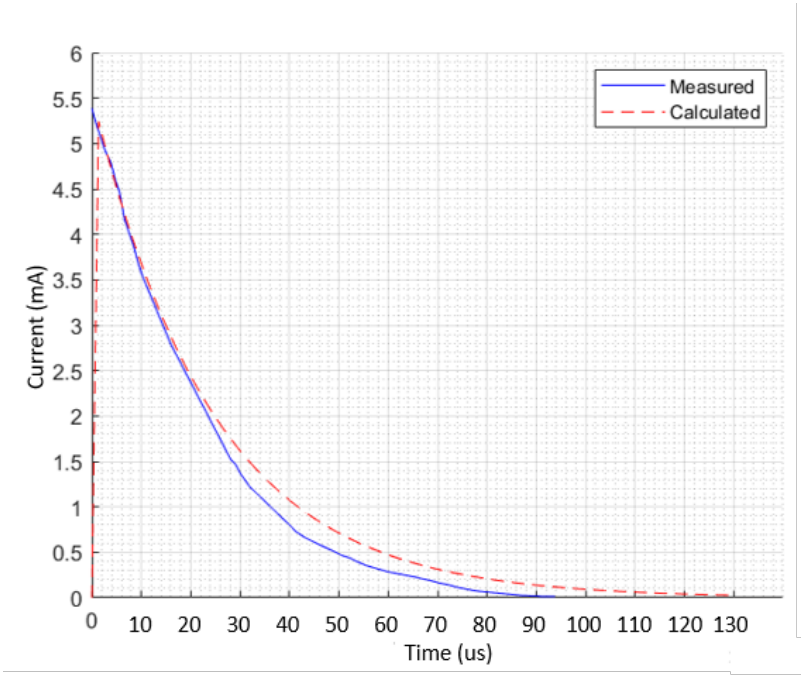
Next, we calculate κ with Eq. (5.6) and then Eq. (5.7) using the computed coefficients and evaluate the results by using the mean absolute percentage deviation. Finally, we adjust t_{rise}, t_{tail}, γ, and δ as required to optimize the fitting and recalculate κ. We used this fitting process in each VBN step of our evaluation.

This fitting process is repeated until coefficients are obtained with a minimal error in accordance with the extracted real-chip measurements made in our previous work; for those measurements, we used an SG-4322 function generator to provide BB. Both VBP and VBN were changed simultaneously. The energy and timing overheads were measured using a Keysight MSOX 4104A oscilloscope and N2820A current probe [**?**]. This process is done to enable a comparison of the calculated results with the measured ones and is used as a reference to fine-tune the analytical parameters. That is, real-chip measurement is required only once.

To check the validity of our proposed method, we compare the measured and calculated results in Fig. 5.4 for the worst cases (the largest and smallest of VBN) evaluation. This fitting process yielded time parameters $t_{rise} = \frac{9.34}{100} t_s$ and $t_{tail} = \frac{24}{100} t_s$ with an average error of 10.5%. We present the results of using the E_{ovs} model in Table 5.1. Although the maximum error was about 14%, the impact on the total energy required was about 1.6%, as explained below. This seems to be a reasonable error. The γ and δ analytical function parameters were fully evaluated through optimization (Section 4) and through the scenario, as shown in Fig. 2.3 (Section 5). The mean error for these settings is discussed in a later section.



(a) VBN = -700 mV



(b) VBN = -200 mV

Fig. 5.4 Comparison of calculated and previously measured current–time profiles for largest and smallest values of VBN: (a) -700 mV; (b) -200 mV.

Table 5.1 Energy overhead: real-chip measurement vs. model analysis results.

VBN (mV)	Real chip (μJ)	Model anal. (μJ)	Error (%)
-700	0.578	0.626	7.66
-600	0.489	0.558	12.46
-500	0.424	0.483	12.10
-400	0.373	0.414	9.82
-300	0.293	0.341	14.18
-200	0.250	0.264	5.31

In short, we update Eq. (5.2) with the energy overhead equation Eq. (5.7), giving us the updated equation for total energy:

$$\begin{aligned} E_T = {} & I \cdot 10^{A \cdot VDD} \cdot VDD \cdot T_{exe} \\ & + \alpha \cdot C \cdot VDD^2 \cdot N \cdot CPI \\ & + VBN \cdot I_{ovs} \cdot \kappa \cdot \left(\frac{e^{-\delta t_s}}{\delta} - \frac{e^{-\gamma t_s}}{\gamma}\right) \\ & + I \cdot 10^{A \cdot VDD + B \cdot VBN} \cdot VDD \cdot T_{id}. \end{aligned} \tag{5.10}$$

5.2 Optimization

5.2.1 Problem Definition

There is a tradeoff between power savings and switching overhead. While a high RBB saves a significant amount of static power in the idle state, the switching overhead is larger. Several variables are involved in this tradeoff. Moreover, there is a considerable number of tradeoff possibilities. Let us consider the BB characteristics tradeoff mentioned in Section 2.1. We control the RBB characteristics by using several electrical parameters concurrently.

More advanced analyses are required to weigh the tradeoffs among all the variables involved. Therefore, we aim at optimizing the selection of the RBB and supply voltage while simultaneously considering the given task deadline, with minimal energy switching penalties and energy waste.

Consistent with the tradeoff information mentioned above, we can describe the

Table 5.2 Group variables involved in the optimization problem.

Group	Variables
Optimization target variables	VBN, VDD
Application coefficient variables (given by the application)	D, N, CPI, T_{exe}, T_{id}
System coefficient variables (given by the system)	I, A, B, α, C, γ, δ, κ
Measured variables	I_{ovs}, t_s

problem as a single-objective optimization problem: given an application, optimize the energy consumption and performance of the given task when there are concurrent options for RBB and the supply voltage.

We use Eq. (5.10) to model this optimization problem. We catalog the variables and coefficients involved in four groups, as summarized in Table 5.2. The system coefficient variables (I, A, B, α, and C) are acquired in accordance with the method described in [**?**].

Here, the problem is to optimize energy consumption by finding the optimal VBN and VDD voltages, constrained by the switching overhead penalties and energy waste. This is thus a problem of finding the minimum constrained nonlinear multi-variable equation.

5.2.2 Interior Point Method - Nonlinear Programming Model

The Newton-Raphson method is commonly applied to engineering problems due to its swift and robust convergence characteristics. Nonetheless, if a given problem has saddles, multiple roots, or the initial condition is not a $valid$ starting point (since from a geometrical point of view, selection of the starting point is arbitrary), the algorithm might get caught in a suboptimal solution or may not even converge. It is thus essential that the convergence condition is ensured; therefore, we use a more robust method, the *interior point* method (IPM). By using the IPM, we can reach and guarantee convergence to the optimum solution by traversing the interior region described by the function rather than around its surface, as done by the Newton-Raphson method. IPM has been proven to

achieve an optimal solution efficiently for these types of optimization problems [**? ? ? ?**]. Its convergence advantage and computationally efficiency make IPM an excellent problem-solving method for NLP [**?**]. The Fig.5.5 illustrates an example on how IPM works.

Therefore, the objective function is the equation for total energy (Eq. (5.10)) a function of VBN, VDD. Hence, the optimization problem is

minimize

$$\begin{aligned} E_T(VBN,VDD) = I \cdot 10^{A \cdot VDD} \cdot VDD \cdot T_{exe} \\ +\alpha \cdot C \cdot VDD^2 \cdot N \cdot CPI \\ +VBN \cdot I_{ovs} \cdot \kappa \cdot (\frac{e^{-\delta t_s}}{\delta} - \frac{e^{-\gamma t_s}}{\gamma}) \\ +I \cdot 10^{A \cdot VDD + B \cdot VBN} \cdot VDD \cdot T_{id} \end{aligned} \tag{5.11}$$

subject to

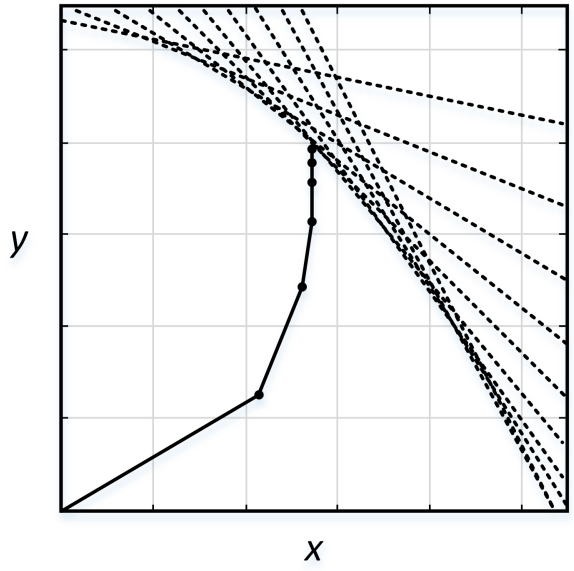
$$-700 \text{ mV} \leq VBN \leq -200 \text{ mV} \tag{5.12}$$

$$304.11 \text{ mV} \leq VDD \leq 470.87 \text{ mV}. \tag{5.13}$$

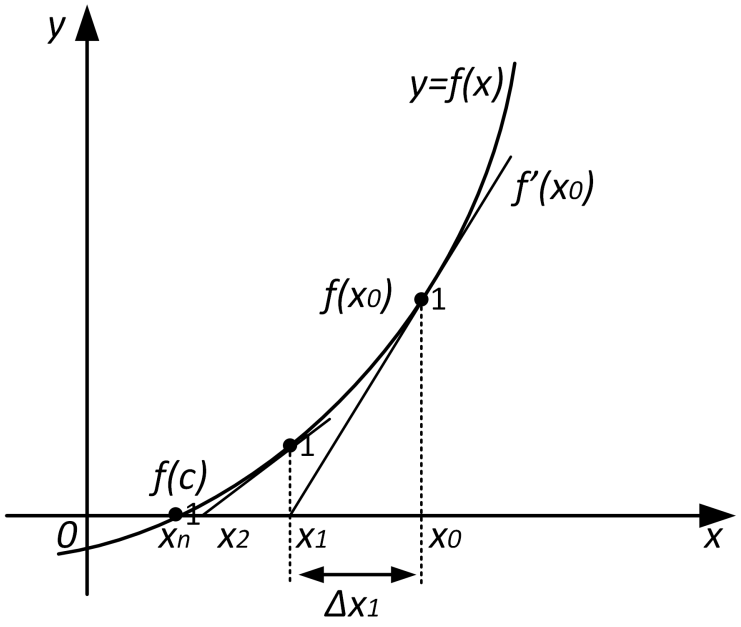
The goal is to minimize E_T. To do so, we minimize VBN and VDD in the objective function (Eq. (5.11)) while satisfying the voltage variation constrained for VBN ((5.12)) and VDD ((5.13)), such constraints are based on our previous analysis of an SOTB device [**?**]. Another crucial constraint is the frequency since VDD is related to frequency by the alpha power law; the system must work at a minimum frequency while attaining the performance required to avoid wasting energy. Furthermore, the frequency must be calculated in accordance with its VDD. Such relationship is described in the next section. We established an evaluation framework [**?**] from 20 MHz to 60 MHz with 10 MHz steps and calculated the VDD from this frequency range.

We must ensure our model complies with the device for its operational time and rising time when applying BB. To demonstrate this, we assume a hypothetical scenario of 3 ms deadline, which is the independent variable.

We develop a program in MATLAB [**?**] to compute the objective function with the IPM–NLP algorithm. The target optimization variables for the algorithm are VBN and VDD. We map the coefficients and the formulas to compute the variables of (Eq. (5.11)). Next, the program calculates the variables and sweeps across the coefficients. Each iteration, the programs evaluate each variable with the possible combinations. In this



(a) Interior Point Method



(b) Newton Raphson

Fig. 5.5 (a) Interior Point Method VS (b) Newton Raphson algorithm. Searching over methodologies.

manner, we keep the relationships among variables. It iterates the objective function of evaluating VBN and VDD until it converges. The results for our scenario were a VBN of -449 mV and a VDD of 397 mV.

In reality, the VBN and operational frequency are discrete values; however, both have various tradeoffs between cost and accuracy, depending on the available BB generators and clock frequency controllers. Since our method can find a continuous optimal value, we can set the most promising discrete values close to the optimal one in consideration of the available BB generators and clock controllers [**? ? ? ?**].

We compiled the optimization model with MATLAB R2019a 9.6.0.1174912 on an HP notebook computer (Windows 10 64-bit, Intel i7-8550U CPU 1.8 GHz, RAM 16 GB). The IPM–NLP computation time was 0.474 s.

To evaluate the efficiency of our methods, we estimated the computation time for a brute-force fine-grain search, whereas we used a brute-force coarse-grain search (real-chip measurements) for the evaluation and results. We used a VBN configuration with a voltage variation of -200 mV to -700 mV with 100 mV steps and a frequency range of 20 MHz to 60 MHz with 10 MHz steps and its associated VDD in accordance with the previously reported method [**?**]. Now we estimate the computation time for the brute-force fine-grain search. We used the same ranges as for the coarse-grain search but with unit step granularity for each case (VBN, VDD, and frequency) and swept through every combination. The computation time for the search was 4.265 s. Our proposed optimization method outperformed in a $\approx$90% the brute-force fine-grain search. Moreover, it guarantees an exact optimal solution.

This optimization process is suitable for compiler or design CAD tools if the execution time of the target program and the deadline are fixed. If they are changed due to a change in requirements, optimization must be done in the run-time system. The execution time of 4.265 s is short enough for optimization to be performed in an edge system. This optimization is needed only when a new task is introduced into the system, which is assumed to happen infrequently to give a severe influence of the energy consumption. Thus, the energy for optimization itself was omitted.

5.2.3 Optimal frequency

Once we find the optimal VDD, the next step is to find the optimal frequency f. The gate delay in MOSFETs is expressed using the alpha power law [**?**]:

$$t_d = \varepsilon \cdot \frac{C \cdot VDD}{(VDD - Vth)^{\alpha}}, \tag{5.14}$$

where ε is the process parameter, α is the velocity saturation coefficient for the MOSFET, and $1 \leq \alpha \leq 2$ (2 in the case of SOTB technology) [**?** **?**]. The frequency is proportional to the reciprocal of t_d. Therefore, we can determine f by using VBN–VDD optimization:

$$f = F \cdot \frac{(VDD - Vth)^{\alpha}}{VDD}, \tag{5.15}$$

where *F* is a coefficient related to frequency, and Vth is the threshold voltage, which varies due to the back gate biasing. It can be linearly approximated using:

$$Vth = V_{th0} - K_{\gamma} VBN, \tag{5.16}$$

where V_{th0} is the threshold voltage with ZBB, and K_{γ} is a constant given by the technology process coefficient (back gate biasing).

Table 3.1 summarizes the power model coefficients obtained from real-chip measurements [**?**]. Using the coefficients in the table in Eq. (5.15), we obtained f_{core} = 38.02 MHz and f_{mem} = 38.72 MHz for the core and memory, respectively. The variation between f_{core} and f_{mem} is very small; therefore, as a rule of thumb, we use the slowest one.

5.3 Results and Discussion

5.3.1 Target System: V850 E-Star

To explore the capabilities of the proposed methodology, we evaluate the break even time (BET) and optimize the energy for several deadlines. BET is important, because, if it exceeds the given deadline, the proposed methodology cannot be used. Instead, the device should remain active to cope with the short deadline.

To evaluate the methodology's efficiency, we used a V850 E-Star microcontroller introduced in chapter 3.

5.3.2 Break Even Time

We evaluate the BET using Eq. (5.7), which is the proposed energy overhead calculation model. It is calculated using

$$BET = \frac{E_{ovs}}{P_s - P_{id}}, \tag{5.17}$$

where P_s is the static power consumed during the active state, and P_{id} is the power consumed during the idle state. In our previous work [**?**], we characterized the efficiency

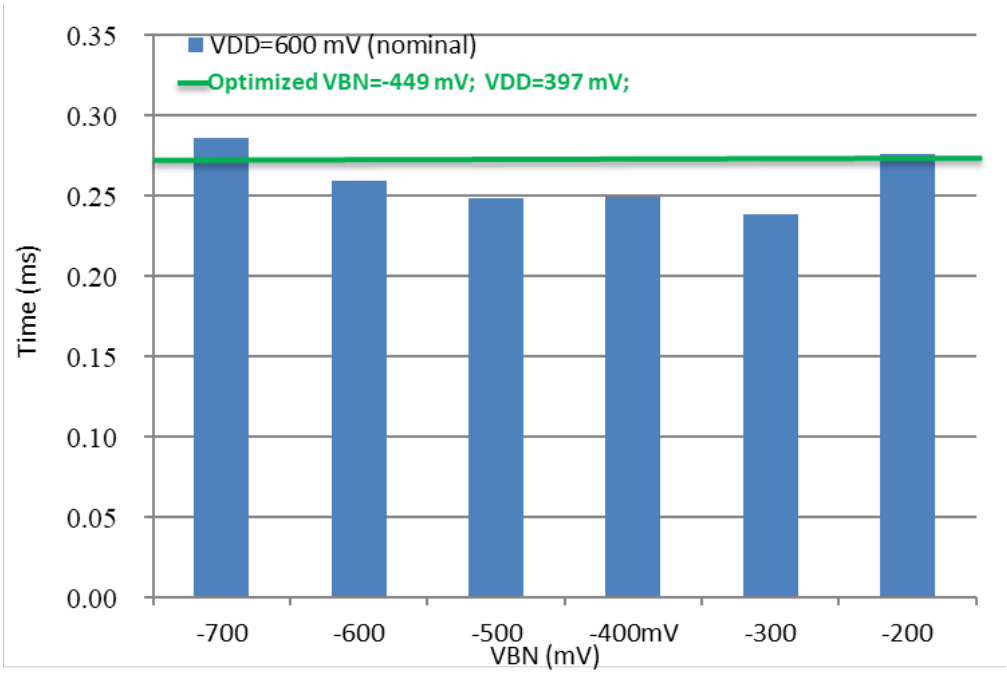


Fig. 5.6 BET comparison: nominal voltage-brute force search vs. optimized VBN–VDD (nominal VDD = 600 m). BET = 0.28 ms for deadline = 3 ms.

of dynamic BB scaling and established the working region of the VBN voltage framework. We set the base of this comparison as the nominal VDD = 600 mV, which is a typical supply voltage for the SOTB used for the V850 E-star. Here we use this working region as a baseline. For the optimization phase, we use a deadline of 3 ms. Moreover, we use the optimized voltage conditions for VDD and VBN computed using Eq. (5.11). As shown in Fig. 5.6, the BET of the optimized VBN is found at the midpoint of the working region. We obtain 0.28 ms. This is consistent with the 0.25 ms for a -500/-400 mV brute-force search, with an ≈10% error.

5.3.3 Optimized VBN–VDD

First, we focus on the active state. We set BB at zero bias, and D and T_{exe} are given. The number N of instructions is determined from the baseline scenario with CPI = 1. As mentioned, the V850 E-Star executes one instruction per clock cycle [**?** **?**]. Since operational frequencies with the settings given above are higher than that of the baseline, the instruction execution of each task finishes prior to the deadline. Furthermore, when a periodic real-time task finishes execution, the system is put into the idle state by the next active state.

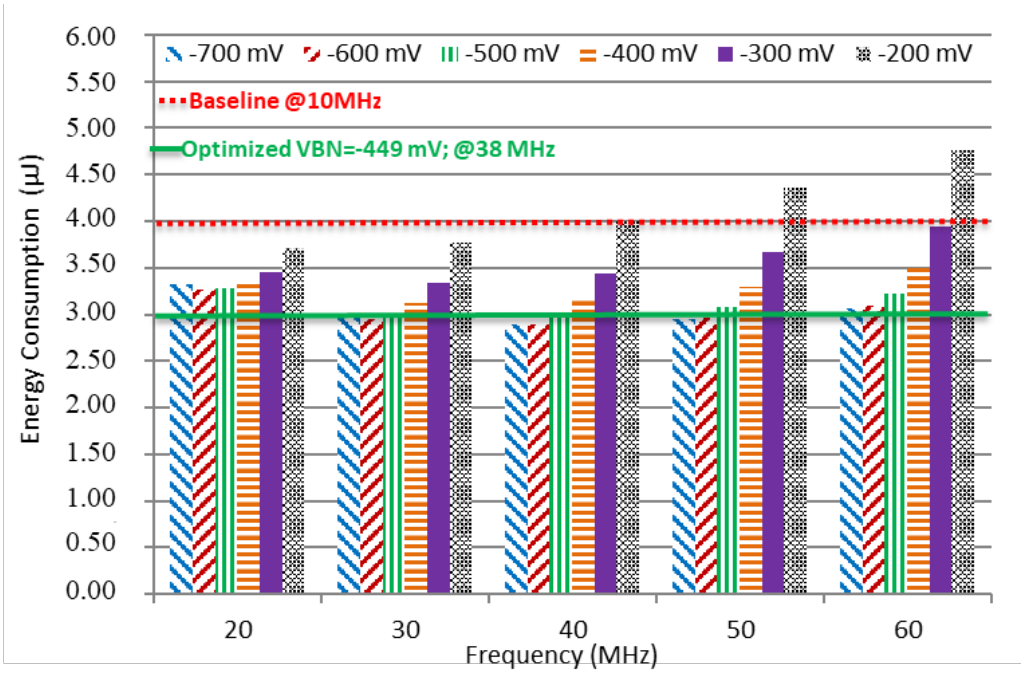


Fig. 5.7 Total energy consumption including energy transition. Brute-force coarse-grain search vs. optimization IPM–NLP: E_T=3.07 μJ, VBN=-449 mV, VDD=397 mV, frequency = 38.06 MHz, deadline = 3 ms.

We use the optimized voltage conditions for VDD and VBN derived from Eq. (5.11). Additionally, we calculate the optimal frequency from Eq. (5.15). We set the optimized VDD and keep it fixed during the active and idle periods. We do not change it dynamically because this increases the cost.

Next, when task execution finishes, we put the system into the idle state by applying RBB with the optimized VBN. This creates an electrical transient, which is calculated using Eq. (5.7).

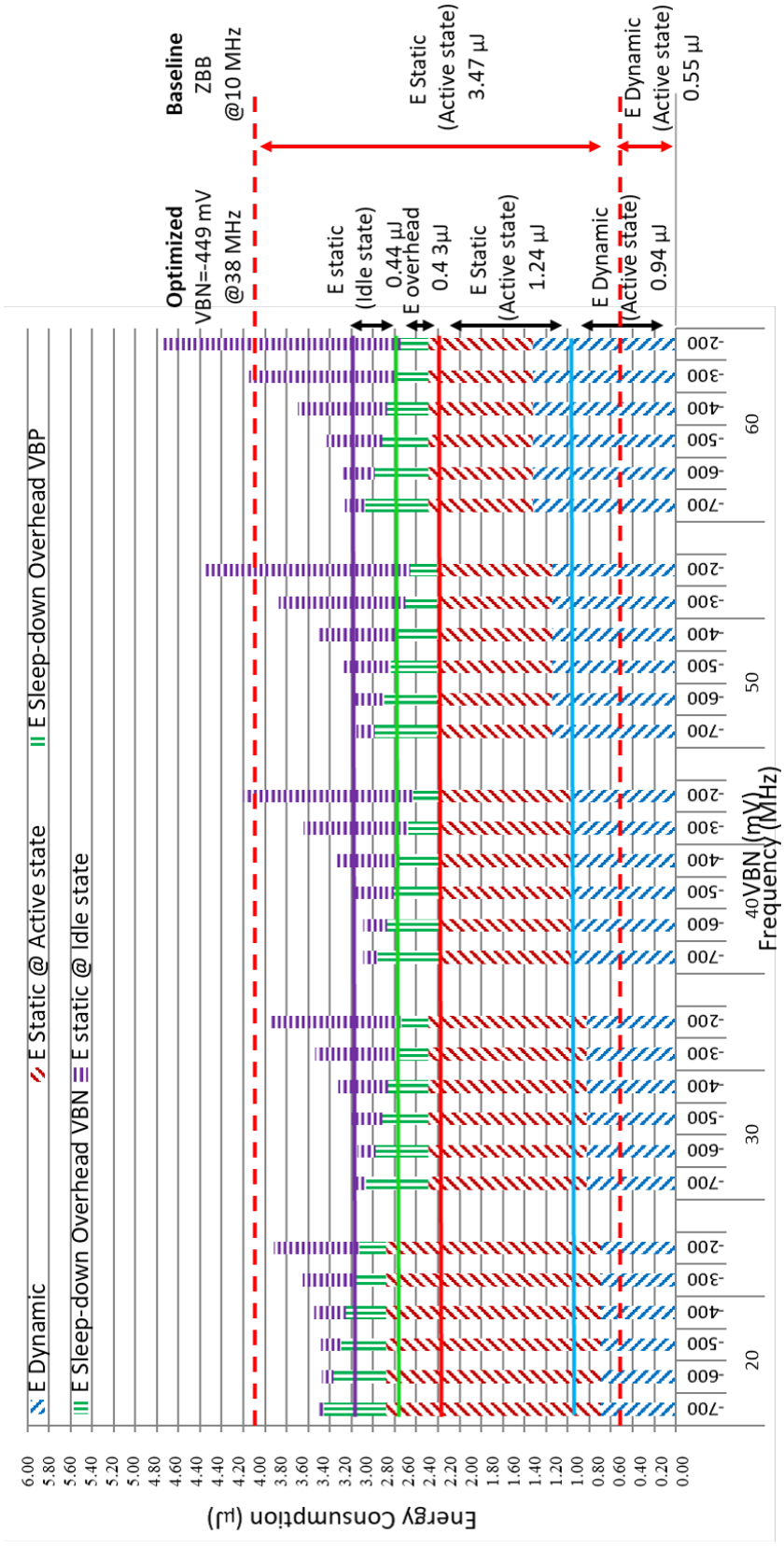


Fig. 5.8 Total energy breakdown by element, explicitly showing amount for each element. The E_{ovs} with optimized value VBN=-449 mV, VDD=397 mV shows a cut down to 14%. Deadline = 3 ms.

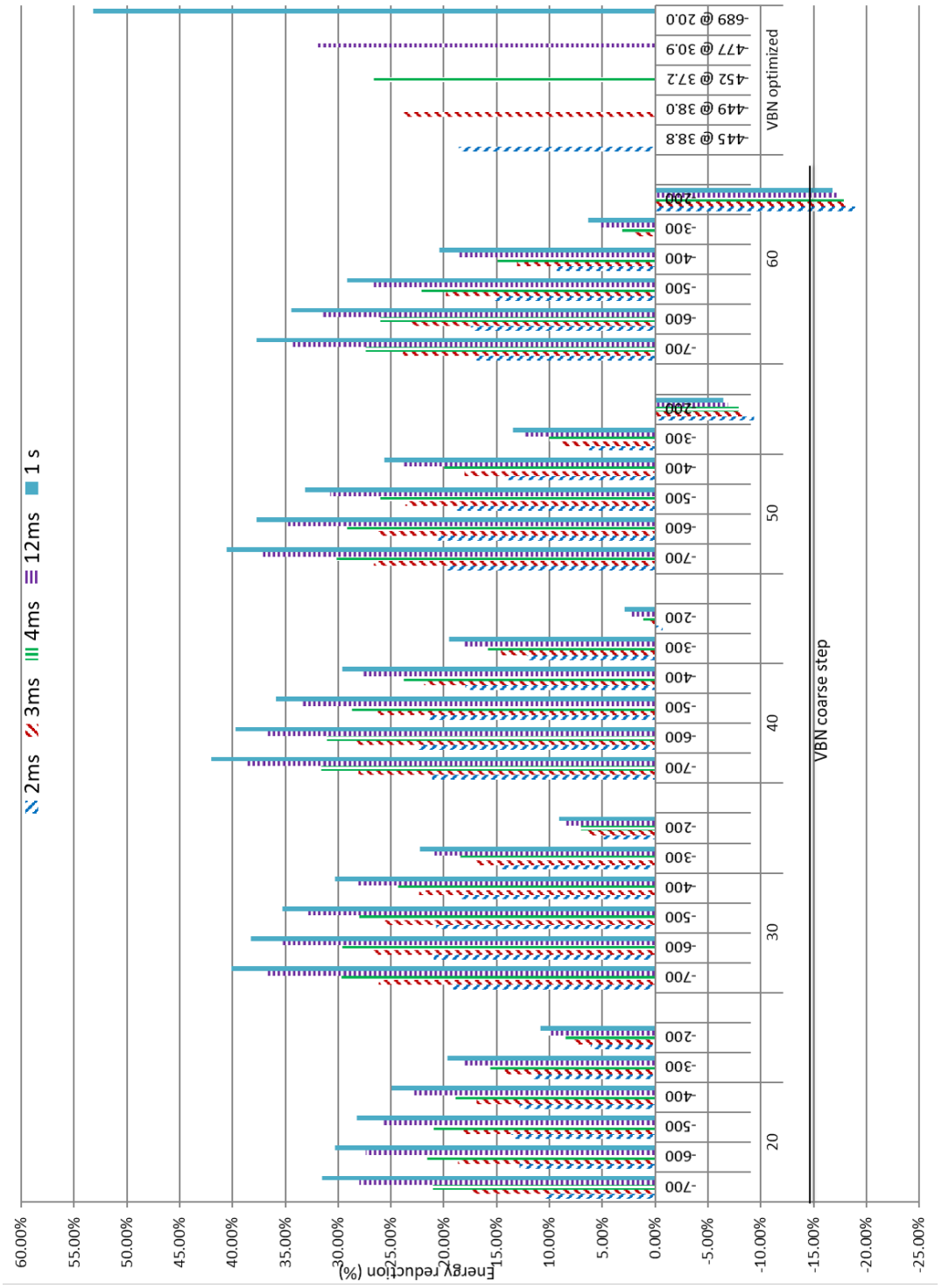


Fig. 5.9 Energy-reduction ratio vs. baseline for deadlines of 2 ms, 3 ms, 4 ms, 12 ms, and 1 s considering leakage current in idle state and optimal frequency. The shorter the deadline, the higher the frequency needed to meet the deadline.

Table 5.3 Supply voltage, RBB voltage and frequency optimized results by deadline, using interior point nonlinear programming.

Deadline (ms)	VDD (mV)	VBN (mV)	Frequency (MHz)
2	399	-445	38.86
3	397	-449	38.06
4	394	-452	37.28
12	375	-477	30.94
1000	341	-689	20.00

As an example of this optimization, we evaluate the test scenario illustrated in Fig. 2.3 using Eq. (5.10) for a deadline of 3 ms. Fig. 5.7 shows the energy consumption at the optimal VBN and the change in energy consumption with different VBN coarse voltages for different frequencies. The baseline scenario is represented by the dotted line (frequency of 10 MHz). The optimal point of energy reduction, represented by the continuous line, is at 38.06 MHz with an energy consumption of 3.07 μJ on average, corresponding to a 76.22% of baseline.

Fig. 5.8 depicts the total energy E_T breakdown by element: E_s, E_d, E_{ovs}, and E_{id}. The graph is grouped by coarse VBN voltages and frequencies. The optimized scenario breakdown energy regions are delimited by horizontal lines. The largest energy reduction is 23.78% for the 38.06-MHz case. These results demonstrate that we can cut the E_{ovs} element (from the VBN coarse voltage step) from 20% (-700 mV worst case) or 6% (-200 mV best case) to an optimal 14% of the total energy.

For further analysis using the same approach used to evaluate the optimized VBN–VDD voltages for a 3 ms deadline (Fig. 5.7), we use this validation approach for deadlines of 2 ms, 3 ms, 4 ms, 12 ms, and 1 s, and use the optimized voltages for VDD and VBN accordingly. As shown in Fig. 5.9, the shorter the deadline, the higher the frequency needed to meet the deadline. The optimized VBN–VDD is on the right side of the graph. The reduction ratios are 18.61%, 23.78%, 26.59%, 32.11%, and 53.19% for 2 ms, 3 ms, 4 ms, 12 ms, and 1 s, respectively. Each reduction ratio is significantly higher than the coarse voltage counterpart. Since the VBN has no significant variation, we can expect that energy reduction is lower for shorter deadlines and higher for longer deadlines. At 1 s, for example, using a lower supply voltage and a lower frequency, it achieves a significant energy reduction. Table 5.3 summarizes the optimized configurations obtained using interior point nonlinear programming.

5.3.4 Model Accuracy

The energy overhead calculation model presented in this paper is based on the power and timing model described in Section 3. In Section 3.1 we presented the model for E_s, E_d, and E_{id}. In 3.2 and 3.3, we introduced the SI double exponential model, Eq. (5.7). This equation introduces double exponential waveform analytical function parameters (γ, δ, and κ). These are the fitting parameters for the SI waveform. We use the Nelder-Mead algorithm for its estimation. This is a well-proven algorithm for estimating these parameters. We consider the following errors.

- *Mean error*. We calculate the error for each VBN coarse voltage as shown in Table 5.1. Despite the difference between the real device and the ideal model, the model depicts a close approximation. We achieve a mean error between the analytical model and the real-chip measurement of 10.5%.

- *Effect over the model*. Although the model uses the time, the error is a function of the VBN voltage. The time duration of E_{ovs} changes slightly however, it does not have a major effect on the waveform. In contrast, the VBN voltage has major changes (every 100 mV), thus it affects the result. The maximum error is about 14%, whereas the effect on total energy is about 1.6%. Even though the model has a mean error of about 10%, the energy reduction is substantially increased. As we can see in Fig. 5.9, the energy reduction ratio increases from 17.97% to 18.61%, from 21.86% to 23.78%, from 23.81% to 26.59%, from 27.71% to 32.11 and from 29.64% to 53.19% for 2 ms, 3 ms, 4 ms, 12 ms, and 1 s, respectively, for decreases in the supply voltage, RBB voltage, and frequency. Thus, the effect of the error over the model is negligible.

Additionally, the coefficients of the target device are dependent on the chip temperature. Nevertheless, as paper [**?**], for the FD-SOI SOTB the coefficients have an accuracy of 93.8% at 25℃ and at 50℃ the accuracy is maintained at 91.6%. However, in the worst case, the highest commercial temperature, 65℃, the accuracy decreases to 79.5%. Thus, the same variations are expected for the proposed model.

Chapter 6

Conclusions and Outlook

6.1 Conclusions

We developed this study in two stages. In the first stage, we presented the first investigation and empirical analysis into BB control with a practical approach to improve the energy efficiency for RTSs. We proposed a mathematical model, for which accurate coefficients were measured from a real chip, as well as overhead parameters used in this model. We also optimized supply voltage (VDD) for a given frequency by using the Brute force coarse-grained method. We analyzed how these overhead conditions affect energy saving with a tradeoff between energy consumption and execution time.

In the second stage, we proposed an analytical approach and methodology for optimizing the reverse body bias and supply voltage using interior point nonlinear programming for real-time systems.

We devised an equation for estimating the overhead energy that includes analytical function coefficients. We computed these coefficients using Nelder-Mead algorithm, thereby transforming the physical parameters of the double exponential waveform into analytical function coefficients. We incorporated this mathematical model to the complete total energy model to improve RTS energy efficiency and accuracy. Then, we used the interior point nonlinear programming model for minimizing the total energy consumption.

The evaluation results demonstrate that the proposed methodology can significantly reduce energy consumption without affecting the system's ability to meet the task deadline. We analyzed how BB optimization affects energy saving in terms of the tradeoff

between energy consumption and execution time. The optimal energy consumption range is from 399 mV to 375 mV for VDD from 2 ms to 12 ms and 341 mV for 1 s. For VBN, it is from -445 mV to -477 mV for the same deadline range and -689 mV for 1 s. This corresponds to frequencies from 38.86 MHz to 30.94 MHz and 20 MHz for 1 s. The results show that the sleep-down transition accounts for 14% of the total energy consumed. We obtained a BET of 0.28 ms, which is consistent with the 0.25 ms of -500 mV/-400 mV brute-force search findings, with $\approx$10% error.

If the execution time of the target program is severely affected by the inputs and difficult to estimate, the proposed methodology cannot be applied. However, a number of real-time scheduling algorithms have been reported for programs for which the execution time can be estimated.

Thus the proposed model can be used as a reference for RTS, automate computation and for CAD (under development and future work) [**? ?**]. Extracting the parameters from real-chip was the first step to model for all the SOTB devices. With this model, we can fix the VBN/VBP, VDD and how much time it should be applied when the target application and the deadline are given. It means that it is useful to design the system, including the chip.

These results demonstrate that the proposed methodology can achieve greater energy reduction, it increases the accuracy and that it can be automated.

Additionally, to select the most suitable BB voltages, we also assess the Break Even Time (BET). We demonstrate SOTB microcontroller BET goes from 0.44ms up to 0.93ms for V850 and 2ms up to 2.25ms for Geyser; these parameters should be included in the algorithms and schedulers for embedded systems. The BET can assure the static energy reduction. Analyzing the BET included in an optimized RBB (interior-point method), confirms our findings in brute force evaluation in the chapter 4.

We analyzed and controlled the VBN outside the chip under the assumption that external control can be equivalent to an internal control since the inputs and outputs have only metal parts; thus, there is no capacitance or diodes. However, on-chip BB generators (under development) [**? ?**] should be used and analyze and its effects (e.g. noise). These results demonstrate that the proposed methodology can achieve greater energy reduction increasing the accuracy and that it can be automated. Moreover, it can be used as a reference for real-time system and computer-aided design.

6.2 Outlook

Although we have found satisfactory optimization results for energy reduction in this study, there is still room of improvement.

In this study, we have developed an analytical model for energy reduction. We analyzed and optimized case studies for a single task for 2ms, 3ms, 4ms, 12ms and 1s. Moreover, the target systems we used to evaluate the energy model and the optimization methods are embedded systems working as stand-alone systems and simple processors and microcontroller.

In this study, I optimized the supply voltage and body bias voltage for a given task and a given deadline, which is a base for implementing in RTSs. Therefore, one point for future work, is that we can do a more comprehensive analysis to include this methodology to real application systems to evaluate its effectiveness. The methodology applied for getting the optimized supply voltage and body bias can be implemented in a decision made by the scheduling algorithm, similar to other proposed studies [**? ?**], in which the DVS scheduling algorithm decides to apply BB. Furthermore, e.g. the algorithm can be implemented in a kernel relying on monitoring system calls.

We also applied computing efficient techniques (Nelder-Mead Simplex algorithm, Interior Point Method) for this aim instead of traditional techniques (e.g. Newton Raphson). Additionally, is that the methodology presented in this study could be evaluated in a cross-platform System On Chip (SoC), Network On Chip (NoC), Graphics Processing Unit (GPU) and Heterogeneous Architectures y future technologies [60]. A second point for future work, is to enhance the methodology to adapt the model to other technologies and analyze its effectiveness.

Another point is to evaluate the model with devices under PVT conditions [**?**], since at the moment of the evaluation only typical (TT) device dies were available. Thus, the future work for this methodology is to validate it across different die devices (fast-FF and slow-SS), temperature and different architecture devices.

References

[1] D. Ikebuchi, N. Seki, Y. Kojima, M. Kamata, L. Zhao, H. Amano, T. Shirai, S. Koyama, T. Hashida, Y. Umahashi, H. Masuda, K. Usami, S. Takeda, H. Nakamura, M. Namiki and M. Kondo, "Geyser-1: A MIPS R3000 CPU core with fine grain runtime power gating," in *Proceedings of IEEE Asian Solid-State Circuits Conference*, Nov 2009, pp. 281–284.

[2] V. Devadas and H. Aydin, "DFR-EDF: A Unified Energy Management Framework for Real-Time Systems," in *IEEE Real-Time and Embedded Technology and Applications Symposium*, 2010, pp. 121–130.

[3] P. Pillai and K. G. Shin, "Real-Time Dynamic Voltage Scaling for Low-Power Embedded Operating Systems," in *ACM Symp. on Operating systems principles*, 2001, pp. 89–102.

[4] L. Yan, J. Luo and N. K. Jha, "Joint dynamic voltage scaling and adaptive body biasing for heterogeneous distributed real-time embedded systems," *IEEE Trans. Comput. Des. Integr. Circuits Syst.*, vol. 24, no. 7, pp. 1030–1041, 2005.

[5] H. Okuhara, K. Kitamori, Y. Fujita, K. Usami and H. Amano, "An optimal power supply and body bias voltage for a ultra low power micro-controller with silicon on thin box MOSFET," in *IEEE/ACM International Symp. On Low Power Electronics and Design*, 2015, pp. 207–212.

[6] C. C. Cortes Torres, H. Okuhara, A. B. Ahmed, N. Yamasaki and H. Amano, "Analysis of Body Bias Control for Real Time Systems," in *Workshop on Synthesis And System Integration of Mixed Information technologies*, Oct. 2016, pp. 48–53.

[7] S.M. Martin, K. Flautner, T. Mudge and D. Blaauw, "Combined dynamic voltage scaling and adaptive body biasing for lower power microprocessors under dynamic workloads." *IEEE/ACM International Conference on Computer Aided Design, 2002. ICCAD*.

[8] T. Ishigaki, R Tsuchiya, Y. Morita, N. Sugii and S. Kimura , *Solid State Circuits Technologies: Ultralow-power LSI Technology with Silicon on Thin Buried Oxide (SOTB) CMOSFET*. InTech, 2010.

[9] B. J. Blalock and P. E. Allen, "Body-driving as low voltage analog design technique for CMOS technology," in *Southwest Symposium on Mixed-Signal Design Iscas*, 2004, pp. 1–6.

[10] K. Zhang, Y. Manzawa and K. Kobayashi, “Impact of body bias on soft error tolerance of bulk and Silicon on Thin BOX structure in 65-nm process,” in *IEEE Int. Reliab. Phys. Symp. Proc.*, 2014, pp. 1–4.

[11] Y. Akgul, D. Puschini, S. Lesecq, E. Beigne, I. Miro-Panades, P. Benoit and L. Torres, “Power management through DVFS and dynamic body biasing in FD-SOI circuits,” in *Proceedings of the 51st Annual Design Automation Conference*, 2014, pp. 1–6.

[12] R. A. McConnell, “Amplitude and energy spectra of transient test waveforms,” in *IEEE EMC International Symposium on Electromagnetic Compatibility*, 2001.

[13] G. Ueta, T. Tsuboi and S. Okabe, “Evaluation of overshoot rate of lightning impulse withstand voltage test waveform based on new base curve fitting methods - study by assuming waveforms generated in an actual test circuit -,” *IEEE Trans. on Dielectrics and Electrical Insulation*, vol. 17, no. 6, pp. 1912–1921, 2010.

[14] J. Y. Zhou and S. A. Boggs, “Low energy single stage high voltage impulse generator,” *IEEE Trans. on Dielectrics and Electrical Insulation*, vol. 11, no. 4, pp. 597–603, 2004.

[15] N. H. E. Weste and D. M. Harris, *Power in CMOS VLSI DESIGN*, 4th ed. Addison-Wesley-Pearson, 2011.

[16] K. Usami and N. Ohkubo, “A Design approach for fine-grained run-time power gating using locally extracted sleep signals,” in *Proceedings of International Conference on Computer Design (ICCD)*, Oct 2006.

[17] Y. J. Chen, C. L. Yang, J. W. Chi and J. J. Chen, “TACLC: Timing-aware cache leakage control for hard real-time systems,” *IEEE Trans. Comput.*, vol. 60, no. 6, pp. 767–782, 2011.

[18] H. Kweon, Y. Do, J. Lee and B. Ahn, “An efficient power-aware scheduling algorithm in real time system,” *IEEE Pacific RIM Conf. Commun. Comput. Signal Process. - Proc.*, pp. 350–353, 2007.

[19] D. Duarte, Y. Tsai, N. Vijaykrishnan and M. J. Irwin, “Evaluating run-time techniques for leakage power reduction,” in *Proc. Asia and South Pacific and the 15th Int. Conf. on VLSI Design*, 2002, pp. 31–38.

[20] R. G. Dreslinski, M. Wieckowski, D. Blaauw, D. Sylvester and T. Mudge, “Near-threshold computing: Reclaiming Moore’s law through energy efficient integrated circuits,” in *Proceedings of the IEEE*, 2010, pp. 253–266.

[21] M.-S. Lee and C.-H. Lee, “Enhanced cycle-conserving dynamic voltage scaling for low-power real-time operating systems,” *IEICE Trans. Inf. Syst.*, vol. E97-D, no. 3, p. 480–487, 2014.

[22] A. Keshavarzi, S. Ma, S. Narendra, B. Bloechel, K. Mistry, T. Ghani, S. Borkar and V. De, “Effectiveness of reverse body bias for leakage control in scaled dual Vt CMOS ICs,” in *Proc. Int. Symp. Low power Electron. Des.*, 2001, pp. 207–212.

[23] T. Chen and S. Naffziger, “Comparison of Adaptive Body Bias (ABB) and Adaptive Supply Voltage (ASV) for Improving Delay and Leakage Under the Presence of Process Variation,” *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 11, no. 5, p. 888–899, 2003.

[24] Y. Akgul, D. Puschini, L. Vincent, M. Altieri and P. Benoit, “Energy-efficient control through power mode placement with discrete DVFS and Body Bias,” *Conf. Proc. - 13th IEEE Int. NEW Circuits Syst. Conf. NEWCAS 2015*, 2015.

[25] I. Lee, J. Leung and S. Son, *Handbook of Real-Time and Embedded Systems*, 1st ed. Chapman and Hall/CRC Taylor and Francis Group, 2008.

[26] Z. Hu, A. Buyuktosunoglu, V. Srinivasan, V. Zyuban, H. Jacobson and P. Bose, “Microarchitectural Techniques for Power Gating of Execution Units,” in *Proceedings - International Symposium on Low Power Electronics and Design*, 2004, p. 32–37.

[27] Y. Shuto, S. Yamamoto and S. Sugahara, “Energy performance of nonvolatile power-gating SRAM using SOTB technology,” in *Proceedings - European Solid-State Circuits Conference*, 2016, p. 87–90.

[28] M. Kondo, H. Kobyashi, R. Sakamoto, M. Wada, J. Tsukamoto, M. Namiki, W. Wang, H. Amano, K. Matsunaga, M. Kudo, K. Usami, T. Komoda and H. Nakamura, “Design and Evaluation of Fine-Grained Power-Gating for Embedded Microprocessors,” in *Proceedings of Design, Automation and Test in Europe Conference and Exhibition (DATE) 2014*, March 2014.

[29] J.M. Kuehn, H. Amano, O. Bringmann and W. Rosenstiel, “Leveraging FDSOI through body bias domain partitioning and bias search,” *Proc. of Design Automation Conference (DAC)*, July 2016.

[30] Y. Tsai, D. E. Duarte, N. Vijaykrishnan and M. J. Irwin, “Characterization and modeling of run-time techniques for leakage power reduction,” *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 12, no. 11, pp. 1221–1232, 2004.

[31] S. Takeda, S. Miwa, K. Usami and H. Nakamura, “Stepwise sleep depth control for run-time leakage power saving,” in *Proc. of the great lakes symposium on VLSI, GLSVLSI’12* , May 2012.

[32] C. C. Cortes Torres, H. Okuhara, N. Yamasaki and H. Amano, “Analysis of Body Bias Control Using Overhead Conditions for Real Time Systems: A Practical Approach,” *IEICE Trans. on Information and Systems*, vol. 101, no. 4, pp. 1116–1125, 2018.

[33] C. Cortes and H. Amano, “Switching Region Analysis for SOTB Technology,” in *International Caribbean Conference on Devices, Circuits and Systems*, June 2017, p. 33–36.

[34] H. A. Carlos Cortes and N. Yamasaki, “Break even time analysis using empirical overhead parameters for embedded systems on sotb technology,” in *Conference on Design of Circuits and Integrated Systems (DCIS)*, Nov 2017.

[35] N. Kamae, A. K. M. Mahfuzul Islam, A. Tsuchiya and H. Onodera, “A Body Bias Generator with Wide Supply-Range down to Threshold Voltage for Within-Die Variability Compensation,” in *Proceedings of the Asian Solid State Circuits Conference*, Nov. 2014, pp. 53–56.

[36] K. Kitamori, H. Su and H. Amano, “Power optimization of a micro-controller with Silicon On Thin Buried Oxide,” in *The 18th Workshop on Synthesis And System Integration of Mixed Information technologies*, 2013, pp. 68–73.

[37] Y. Koizumi, H. Amano, H. Matsutani, N. Miura, T. Kuroda, R. Sakamoto, M. Namiki, K. Usami, M. Kondo, and H. Nakamura, “Dynamic power control with a heterogeneous multi-core system using a 3-d wireless inductive coupling interconnect,” in *2012 International Conference on Field-Programmable Technology*, Dec. 2012, pp. 10–12.

[38] L. Zhao, D. Ikebuchi, Y. Saito, M. Kamata, N. Seki, Y. Kojima, H. Amano, S. Koyama, T. Hashida, Y. Umahashi, D. Masuda, K. Usami, K. Kimura, M. Namiki, S. Takeda, H. Nakamura, and M. Kondo, “Geyser-2: The second prototype CPU with fine-grained run-time power gating,” in *Proc. of the Asia and South Pacific Design Automation Conference, (ASP-DAC’2011)*, 2011, pp. 87–88.

[39] Y. Take, H. Matsutani, D. Sasaki, M. Koibuchi, T. Kuroda and H. Amano, “3-D NoC with Inductive-Coupling Links for Building-Block SiPs,” *IEEE Transactions on Computers (TC)*, vol. 63, no. 3, pp. 748–763, Mar. 2014.

[40] T. Katagiri and H. Amano, “A high speed design and implementation of dynamically reconfigurable processor using 28nm SOI technology,” *Proc. of Field Programable Logic and Applications (FPL)*, Sept. 2014.

[41] H. Amano, *ARC 200th Commemorative Panel*. The IPSJ Special Interest Group on System Architecture, (IPSJ-SIGARC’2014), 2014, [Japanese].

[42] J. M. Kühn, A. B. Ahmed, H. Okuhara, H. Amano, O. Bringmann, and W. Rosenstiel, “Muccra4-bb: A fine-grained body biasing capable drp,” *2016 IEEE Symposium in Low-Power and High-Speed Chips (COOL CHIPS XIX)*, pp. 1–3, 2016.

[43] J. M. Kühn, “FDSOI Design using Automated Standard-Cell-Grained Body Biasing,” PhD Thesis, Eberhard-Karls-University Tübingen, 2016.

[44] NEC Corporation, *V800 Series multimedia RISC microcomputers pave the way for System On a Chip.*, 12th ed. NEC Corporation, 2000.

[45] Renesas Electronics Corporation, *V850/SC1TM , V850/SC2TM , V850/SC3TM. 32-Bit Single-Chip Microcontrollers. User’s Manual. RENESAS*, 3rd ed. Renesas Electronics Corporation, 2010.

[46] H. Okuhara, Y. Fujita, K. Usami and H. Amano, “Power Optimization Methodology for Ultralow Power Microcontroller with Silicon on Thin BOX MOSFET,” *IEEE Trans. Very Large Scale Integr. Syst*, vol. 25, no. 4, pp. 1578–1582, 2017.

[47] S. Clerc, M. Saligane, F. Abouzeid, M. Cochet, J. Daveau, C. Bottoni, D. Bol, J. De-Vos, D. Zamora, B. Coeffic, D. Soussan, D. Croain, M. Naceur, P. Schamberger, P. Roche and D. Sylvester, "8.4 A 0.33V40°C Process/Temperature Closed-Loop Compensation SoC Embedding All-Digital Clock Multipller and DC-DC Converter Exploiting FDSOI 28nm Back-Gate Biasing," in *Proceedings of International Solid-State Circuits Conference (ISSCC)*, Feb. 2015, pp. 150–151.

[48] M. Blagojevic, M. Cochet, B. Keller, P. Flatresse, A. Vladimirescu and B. Nikolic, "A Fast, Flexible, Positive and Negative Adaptive Body-Bias Generator in 28nm FDSOI," in *Proceedings of IEEE symposium on VLSI Circuits*, June 2016, pp. 1–2.

[49] S. Hoppner et al., "Adaptive body bias aware implementation for ultra-low-voltage designs in 22fdx technology," *IEEE Transactions on Circuits and Systems II: Express Briefs*, pp. 1–1, 2019.

[50] H. Okuhara, A.B. Ahmed and H. Amano, "Digitally Assisted On-Chip Body Bias Tuning Scheme for Ultra Low-Power VLSI systems,," in *2016 IEEE Trans. on Circuits and Systems I*, vol. 65, 2018, pp. 3241–3253.

[51] G. Ramarao and K. Chandrasekaran, "Calculation of multistage impulse circuit and its analytical function parameters," *International Journal of Pure and Applied Mathematics*, vol. 114, no. 12, pp. 583–592, 2017.

[52] M. Camp and H. Garbe, "Parameter Estimation of Double Exponential Pulses (EMP, UWB) With Least Squares and Nelder Mead Algorithm," *IEEE Trans. on Electromagnetic Compatibility*, vol. 46, no. 4, pp. 675–678, 2004.

[53] J. Nelder and R. Mead, "A simplex method for funtion minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965.

[54] Wikipedia. The Free Encyclopedia., *Nelder–Mead method*. Wikipedia. The Free Encyclopedia., 2020. [Online]. Available: https://en.wikipedia.org/wiki/Nelder-Mead_Method

[55] X. Yang, *Engineering Optimization Introduction, An Applications, Metaheuristic*. Wiley, 2010.

[56] B. Venkateswara Rao, G. V. Nagesh Kumar, R. V. S. Lakshmi Kumari and N. G. S. Raju, "Optimization of a power system with interior point method," in *International Conference on Power and Energy Systems*, Dec 2011.

[57] Wikipedia. The Free Encyclopedia., *Interior-point method*. Wikipedia. The Free Encyclopedia., 2020. [Online]. Available: https://en.wikipedia.org/wiki/Interior-point_method

[58] I. I. Dikin, "Iterative solution of problems of linear and quadratic programming." *Soviet Mathematics. Doklady, 8:674. ISSN: 0197-6788*.

[59] Math Works, *MATLAB: The language of technical computing from Math Works*. Math Works, 2018. [Online]. Available: https://www.mathworks.com

[60] H. Okuhara et al., “Power optimization methodology for ultralow power microcontroller with silicon on thin box mosfet,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 4, pp. 1578–1582, 2017.

[61] T. A. Krauss, “Planar Electrostatically Doped Reconfigurable Schottky Barrier FDSOI Field-Effect Transistor Structures,” PhD Thesis, Technical University of Darmstadt, 2017.

[62] Institute for information processing technology (ITIV)., *PARFAIT*. Karlsruhe Institute of Technology, 2017. [Online]. Available: https://www.itiv.kit.edu/english/5497.php

[63] H. Cai et al., “Robust ultra-low power non-volatile logic-in-memory circuits in fd-soi technology,” *IEEE Transactions on circuits and systems-I*, vol. 64, no. 4, pp. 847–857, 2017.

Appendix A

MATLAB source code

Interior Point Algorithm - Non-Linear Programming

```
%=========================================================================
=======================================
% Keio University, Information and Information Department <HungaLab> Created by
Carlos Cortes (2018/06/22)
% Last update (2018/06/22) version 1
% Description:
% Program made for the analysis of the bias of the body.
% Performs an optimization of the bias of the inverse body, when finding the minimum
value of VBN.
% Use the embedded MATLAB function "fmincon".
% Sweep through all combinations of VBN values.
% Save the results in an Excel file.
% Input files:
% There are no input files, all the information is embedded in this file.
% Output files:
% - "fileNameXls".xlsx
% How to execute:
% Write in the MATLAB command terminal: fminconSweep20180622.m
% License:
% You can use this code for your research giving me the appropriate mention in your work.
%=========================================================================
=======================================
```

```
clear;

vbnParam = 700;
it = 1;
it2 = 1;
itMax = 5;
freqCnt = 2; itTable = 1;

fileNameXls='VddVbnOptFminconSweep20190518D3ms.xlsx';
sheetRaw='raw';
sheetGraph='Graphs';
sheetTable='Tables';
headerXlsRaw='Iteration','newVal(vdd)','newVal(vbn)','vdd','vbn','aatCm','Am','Bm',
'Im','aatCc','Ac','Bc','Ic','a','b','k','NCPI','Tid','Ttsn','Texe','Freq','Deadline';
xlswrite(fileNameXls,headerXlsRaw,sheetRaw);
%headerXlsGraph='vdd','vbn','Freq','700NV(vdd)','700NV(vbn)','700NVRatio(%)','600N
V(vdd)','600NV(vbn)','600NVRatio(%)','500NV(vdd)','500NV(vbn)','500NVRatio(%)','40
0NV(vdd)','400NV(vbn)','400NVRatio(%)','300NV(vdd)','300NV(vbn)','300NVRatio(%)',
'200NV(vdd)','200NV(vbn)','200NVRatio(%)';
headerXlsGraph='vdd','vbn','Freq','newVal(vdd)','newVal(vbn)','Ratio(%)';
xlswrite(fileNameXls,headerXlsGraph,sheetGraph);

headerXlsTable='Freq','newVal(vdd)','newVal(vdd)','newVal(vdd)','newVal(vdd)','new
Val(vdd)','newVal(vdd)';
xlswrite(fileNameXls,headerXlsTable,sheetTable,'A1');
xlswrite(fileNameXls,20,sheetTable,'A2');
xlswrite(fileNameXls,30,sheetTable,'A3');
xlswrite(fileNameXls,40,sheetTable,'A4');
xlswrite(fileNameXls,50,sheetTable,'A5');
xlswrite(fileNameXls,60,sheetTable,'A6');

headerXlsTable='Freq','newVal(vbn)','newVal(vbn)','newVal(vbn)','newVal(vbn)','new
Val(vbn)','newVal(vbn)';
xlswrite(fileNameXls,headerXlsTable,sheetTable,'A8');
```

```
xlswrite(fileNameXls,20,sheetTable,'A9');
xlswrite(fileNameXls,30,sheetTable,'A10');
xlswrite(fileNameXls,40,sheetTable,'A11');
xlswrite(fileNameXls,50,sheetTable,'A12');
xlswrite(fileNameXls,60,sheetTable,'A13');

while(vbnParam>=200)%

while(freqCnt<=6)%
%syms Et(vdd, vbn);
% syms Eid(vdd, vbn);
% % % syms a b k;
% syms aatCc Ac Bc Ic;
% syms aatCm Am Bm Im;
% syms Texe Tid Tts;
% % syms NCPI;
% syms vth0 vth fmax Freq kg;
% syms Ioc Iom;
% syms EtVal;

%Et = (v)((Ic*10^(Ac*v(1))+Im*10^(Am*v(1)))*v(1)*Texe)+((aatCc+aatCm)*v(1)^2*NCPI)
+((Ic*10^(Ac*v(1)+Bc*v(2))+Im*10^(Am*v(1)+Bm*v(2)))*v(1)*Tid) + (k*v(2)*(exp(-a)-
exp(-b))*(Ioc+Iom)*Tts) - EtVal;

%vddEq=((((vth0-kg*vbn)+fmax/Freq)+(((vth0-kg*vbn)+(fmax/Freq))^2-4*(vth0-kg*vbn)
^2)^(1/2))/2)-vdd; %vddEq

%Previous value b=1.316E+00; a=1.000E+01; k=1.566E+00;

if(vbnParam == 700)column = 'B';
%VDD-V850 VBN(V) Z2R Beta Alpha k
vbn=-0.200; Ttsmp=107.81e-6; Ttscp=110.93e-6; Ttsmn=106.25e-6; Ttscn=129.68e-6;
Ttsn=125.78e-6; Ttsp=102.08e-6; b=10000000; a=38655; k=1.0258; Iocp=5.91E-03;
Iocn = 6.16E-03; Iomp=5.89E-03; Iomn = 6.24E-03;
if (freqCnt==2) vdd=0.47087;Tid=1.26E-03; Texe=1.50E-03;Freq=20; EtVal=1.20E-06;
end if (freqCnt==3) vdd=0.47087;Tid=1.76E-03; Texe=1.00E-03;Freq=30; EtVal=1.21E-
```

06; end if (freqCnt==4) vdd=0.47087;Tid=2.01E-03; Texe=7.50E-04;Freq=40; EtVal=1.30E-06; end if (freqCnt==5) vdd=0.47087;Tid=2.16E-03; Texe=6.00E-04;Freq=50; EtVal=1.44E-06; end if (freqCnt==6) vdd=0.47087;Tid=2.26E-03; Texe=5.00E-04;Freq=60; EtVal=1.60E-06; end

elseif(vbnParam == 600)column = 'C';
vbn=-0.200; Ttsmp=106.25e-6; Ttscp=96.87e-6; Ttsmn=104.68e-6; Ttscn=129.68e-6; Ttsn=125.78e-6; Ttsp=102.08e-6; b=10000000; a=38655; k=1.0258; Iocp=5.91E-03; Iocn = 6.1E-03; Iomp=5.97E-03; Iomn = 6.22E-03;
if (freqCnt==2) vdd=0.47087;Tid=1.27E-03; Texe=1.50E-03;Freq=20; EtVal=1.21E-06; end if (freqCnt==3) vdd=0.47087;Tid=1.77E-03; Texe=1.00E-03;Freq=30; EtVal=1.22E-06; end if (freqCnt==4) vdd=0.47087;Tid=2.02E-03; Texe=7.50E-04;Freq=40; EtVal=1.31E-06; end if (freqCnt==5) vdd=0.47087;Tid=2.17E-03; Texe=6.00E-04;Freq=50; EtVal=1.46E-06; end if (freqCnt==6) vdd=0.47087;Tid=2.27E-03; Texe=5.00E-04;Freq=60; EtVal=1.62E-06; end

elseif(vbnParam == 500) column = 'D';
vbn=-0.200; Ttsmp=103.12e-6; Ttscp=100e-6; Ttsmn=104.68e-6; Ttscn=126.56e-6; Ttsn=125.78e-6; Ttsp=102.08e-6; b=10000000; a=38655; k=1.0258; Iocp=5.91E-03; Iocn = 6.09E-03; Iomp=5.64E-03; Iomn = 6.22E-03;
if (freqCnt==2) vdd=0.47087;Tid=1.29E-03; Texe=1.50E-03;Freq=20; EtVal=1.22E-06; end if (freqCnt==3) vdd=0.47087;Tid=1.79E-03; Texe=1.00E-03;Freq=30; EtVal=1.24E-06; end if (freqCnt==4) vdd=0.47087;Tid=2.04E-03; Texe=7.50E-04;Freq=40; EtVal=1.34E-06; end if (freqCnt==5) vdd=0.47087;Tid=2.19E-03; Texe=6.00E-04;Freq=50; EtVal=1.49E-06; end if (freqCnt==6) vdd=0.47087;Tid=2.29E-03; Texe=5.00E-04;Freq=60; EtVal=1.66E-06; end

elseif(vbnParam == 400) column = 'E';
vbn=-0.200; Ttsmp=101.56e-6; Ttscp=100e-6; Ttsmn=103.12e-6; Ttscn=123.43e-6; Ttsn=125.78e-6; Ttsp=102.08e-6; b=10000000; a=38655; k=1.0258; Iocp=5.91E-03; Iocn = 6.09E-03; Iomp=5.58E-03; Iomn = 6.21E-03;
if (freqCnt==2) vdd=0.47087;Tid=1.29E-03; Texe=1.50E-03;Freq=20; EtVal=1.25E-06; end if (freqCnt==3) vdd=0.47087;Tid=1.79E-03; Texe=1.00E-03;Freq=30; EtVal=1.28E-06; end if (freqCnt==4) vdd=0.47087;Tid=2.04E-03; Texe=7.50E-04;Freq=40; EtVal=1.39E-06; end if (freqCnt==5) vdd=0.47087;Tid=2.19E-03; Texe=6.00E-04;Freq=50; EtVal=1.55E-06; end if (freqCnt==6) vdd=0.47087;Tid=2.29E-03; Texe=5.00E-04;Freq=60; EtVal=1.73E-

06; end

elseif(vbnParam == 300) column = 'F';
vbn=-0.200; Ttsmp=100e-6; Ttscp=100e-6; Ttsmn=98.43e-6; Ttscn=123.43e-6;
Ttsn=125.78e-6; Ttsp=102.08e-6; b=10000000; a=38655; k=1.0258; Iocp=5.83E-03;
Iocn = 6E-03; Iomp=5.39E-03; Iomn = 6.19E-03;
if (freqCnt==2) vdd=0.47087;Tid=1.30E-03; Texe=1.50E-03;Freq=20; EtVal=1.29E-06;
end if (freqCnt==3) vdd=0.47087;Tid=1.80E-03; Texe=1.00E-03;Freq=30; EtVal=1.35E-
06; end if (freqCnt==4) vdd=0.47087;Tid=2.05E-03; Texe=7.50E-04;Freq=40; EtVal=1.47E-
06; end if (freqCnt==5) vdd=0.47087;Tid=2.20E-03; Texe=6.00E-04;Freq=50; EtVal=1.65E-
06; end if (freqCnt==6) vdd=0.47087;Tid=2.30E-03; Texe=5.00E-04;Freq=60; EtVal=1.84E-
06; end

elseif(vbnParam == 200) column = 'G';
vbn=-0.200; Ttsmp=93.75e-6; Ttscp=96.87e-6; Ttsmn=96.87e-6; Ttscn=121.87e-6;
Ttsn=125.78e-6; Ttsp=102.08e-6; b=10000000; a=38655; k=1.0258; Iocp=5.83E-03;
Iocn = 5.5E-03; Iomp=5.2E-03; Iomn = 6.021E-03;
if (freqCnt==2) vdd=0.47087;Tid=1.30E-03; Texe=1.50E-03;Freq=20; EtVal=1.36E-06;
end if (freqCnt==3) vdd=0.47087;Tid=1.80E-03; Texe=1.00E-03;Freq=30; EtVal=1.46E-
06; end if (freqCnt==4) vdd=0.47087;Tid=2.05E-03; Texe=7.50E-04;Freq=40; EtVal=1.62E-
06; end if (freqCnt==5) vdd=0.47087;Tid=2.20E-03; Texe=6.00E-04;Freq=50; EtVal=1.82E-
06; end if (freqCnt==6) vdd=0.47087;Tid=2.30E-03; Texe=5.00E-04;Freq=60; EtVal=2.05E-
06; end end

aatCc=6.274770E-11;aatCm=1.366870E-10;Ac=0.509666667;Am=0.395333333;
Bc=2.380569354;Bm=2.380569354;Ic=0.000186;Im=0.000650667;
%NCPI=1.2E+05; Dead=12E-03; Texe=3E-03; %12ms Deadline
%NCPI=4E+04; Dead=4E-03; Texe=1E-03; %4ms Deadline
NCPI=3E+04; Dead=3E-03; Texe=7.5E-04; %3ms Deadline
%NCPI=2E+04; Dead=2E-03; Texe=5E-04; %2ms Deadline
Tid=Dead-Texe-Ttsn;
vth0=(1.95E-01+2.3E-01); fmax=3E+07; kg=(0.111039695+0.068156727);
%v(1)= vdd
%v(2)=vbn
%Et = @(v)((Ic*10^(Ac*v(1))+Im*10^(Am*v(1)))*v(1)*Texe)+((aatCc+aatCm)*

```
v(1)^2*NCPI)+((Ic*10^(Ac*v(1)+Bc*v(2))+Im*10^(Am*v(1)+Bm*v(2)))*v(1)*Tid) +
(v(2)*k*(Iocp+Iomp)*((exp(-(Tts/b)))/b-(exp(-(Tts/a)))/a)) - EtVal;
%Et = @(v)((Ic*10^(Ac*v(1))+Im*10^(Am*v(1)))*v(1)*Texe)+((aatCc+aatCm)*
v(1)^2*NCPI)+((Ic*10^(Ac*v(1)+Bc*v(2))+Im*10^(Am*v(1)+Bm*v(2)))*v(1)*Tid) +
(((v(2)*(Iocn+Iomn))+((v(2)+0.6)*(Iocp+Iomp)))*k*((exp(-(Tts/b)))/b-(exp(-(Tts/a)))/a)) - Et-
Val;
%Et = @(v)((Ic*10^(Ac*v(1))+Im*10^(Am*v(1)))*v(1)*Texe)+((aatCc+aatCm)*
v(1)^2*NCPI)+((Ic*10^(Ac*v(1)+Bc*v(2))+Im*10^(Am*v(1)+Bm*v(2)))*v(1)*Tid) +
(v(2)*(Iocn+Iomn)*k*((exp(-(Ttsn/b)))/b-(exp(-(Ttsn/a)))/a)) + (((-1*v(2))+0.6)*(Iocp+Iomp)
k*((exp(-(Ttsp/b)))/b-(exp(-(Ttsp/a)))/a)) - EtVal;
Et = @(v)((Ic*10^(Ac*v(1))+Im*10^(Am*v(1)))*v(1)*Texe)+((aatCc+aatCm)*
(v(1)^2)*NCPI)+((Ic*10^(Ac*v(1)+Bc*v(2))+Im*10^(Am*v(1)+Bm*v(2)))*v(1)*Tid) +
(v(2)*(Iocn+Iomn)*k*((exp(-(Ttsn/b)))/b-(exp(-(Ttsn/a)))/a)) + (((-1*v(2))+0.6)*(Iocp+Iomp)
k*((exp(-(Ttsp/b)))/b-(exp(-(Ttsp/a)))/a));
%vddEq=((((vth0-kg*vbn)+fmax/Freq)+(((vth0-kg*vbn)+(fmax/Freq))^2-4*
(vth0-kg*vbn)^2)^(1/2))/2)-vdd; %vddEq

%[cC, ceqC]=nonlcon1(v)((((vth0-kg*vbn)+fmax/Freq)+(((vth0-kg*v(2))+(fmax/Freq))^2-
4*(vth0-kg*v(2))^2)^(1/2))/2)-v(1); %vddEq
%c=[-0.2>=v(2)<=-0.7, 0.47087<=v(1)>=0.34099];

%nonlcon2=@(v)[v(1)^2 + v(2)^2 - 1, ];
cC = [];
ceqC = [];
AeqC=[];
beqC=[];
AC=[];
bC=[];
lbC=[0.34099, -0.7];
ubC=[0.47087,-0.2];
%options = optimoptions('fmincon','Display','iter','Algorithm','interior-point',
'OutputFcn',optimplotfval); %Display ON
options = optimoptions('fmincon','Display','iter','Algorithm','interior-point');
%Display ON
%options = optimoptions('fmincon','Algorithm','interior-point'); %Display Off
x0=[vdd,vbn];
```

```
%val = [vdd;vbn];
vbnOrigV=vbn;
vddOrigV=vdd;
% itCnt=0;

%while(itCnt<itMax)
newVal = fmincon(Et,x0,AC,bC,AeqC,beqC,lbC,ubC,nonlcon1,options);

newRowRaw=strcat('A',num2str(it+1));
outputMatrix = it newVal(1) newVal(2) vdd vbn aatCm Am Bm Im aatCc Ac Bc Ic a b k
NCPI Tid Ttsn Texe Freq Dead;
xlswrite(fileNameXls,outputMatrix,sheetRaw,newRowRaw);

% if(itCnt==0)
% newRowGraph=strcat('A',num2str(it2+1));
%newColGraph=strcat('B',num2str(it2+1));
ratio = (newVal(2)*100)/vbn;
%outputMatrixGraphBase = Freq vddOrigV;
outputMatrixGraph = vddOrigV vbnOrigV Freq newVal(1) newVal(2) ratio;
xlswrite(fileNameXls,outputMatrixGraph,sheetGraph,newRowRaw);
%it2=it2+1;
% end

it=it+1;
% val = newVal;
% vdd = val(1);
% vbn = val(2);
% itCnt=itCnt+1;
%end
newRowTable=strcat(column,num2str(itTable+1));
outputMatrixTable = newVal(1); %vdd
xlswrite(fileNameXls,outputMatrixTable,sheetTable,newRowTable);

newRowTable=strcat(column,num2str(itTable+8));
outputMatrixTable = newVal(2); %vbn
```

```
xlswrite(fileNameXls,outputMatrixTable,sheetTable,newRowTable);
itTable=itTable+1;

freqCnt=freqCnt+1;
%it2=it2+1;
end
freqCnt = 2;
itTable = 1;
%it2 = 0 ;
% it2=it2+1;
vbnParam = vbnParam - 100;
end
```

MISC Functions: Plotting

```
%=========================================================================
=========================================
% Keio University, Information and Information Department <HungaLab> Created by
Carlos Cortes (2018/07/16)
% Last update (2018/07/16) version 1
% Description:
% Plotting 3D function based on an excel file.
% Input files:
% OptValPlot.xlsx, containing matrix.
% Output files:
% MATLAB graph
% How to execute:
% Write in the MATLAB command terminal: PlottingExamples.m
% License:
% You can use this code for your research giving me the appropriate mention in your work.
%=========================================================================
=========================================
DS= xlsread('OptValPlot.xlsx','Sheet1','A1:G7');

%x=DS(:,1);
%y=DS(:,2);

Freq=[10 20 30 40 50 60];
VBNX0=[700 600 500 400 300 200];
IntPointVBN=DS(2:end,2:end);
surf(Freq,VBNX0,IntPointVBN);
title('VBN optimization solution space');
xlabel('Frequency (MHz)');
ylabel('VBN coarse step (mV)');
zlabel('VBN minimal point (mV)');
```

Achievements

Publications

Journal Papers

[1] Carlos C. Cortes Torres, Ryota Yasudo and Hideharu Amano Body Bias Optimization for Real-Time Systems, Journal of Low Power Electronics and Applications, Vol.10, No.1, February 2020, DOI 10.3390/jlpea10010008.

[2] Carlos C. Cortes Torres, H. Okuhara, N. Yamasaki and H. Amano Analysis of Body Bias Control Using Overhead Conditions for Real Time Systems: A Practical Approach, IEICE Transactions on Information and Systems, Vol.101, No.4, pp.1116–1125, April 2018.

International Conference Papers

[3] Carlos C. Cortes Torres, H. Amano and N. Yamasaki, Break even time analysis using empirical overhead parameters for embedded systems on SOTB technology, *Conference on Design of Circuits and Integrated Systems (DCIS'2017)*, pp.1–6, Nov 2017.

[4] Carlos C. Cortes Torres, and H. Amano, Switching Region Analysis for SOTB Technology, *International Caribbean Conference on Devices, Circuits and Systems (ICCDCS'2017)*, pp.33–36, Jun 2017.

[5] Carlos C. Cortes Torres, H. Okuhara, A. Ben Ahmed, N. Yamasaki, and H. Amano, Switching Region Analysis for SOTB Technology, *Workshop on Synthesis And System Integration of Mixed Information technologies (SASIMI'2016)*, pp.48–53, Oct 2016.

Co-Authoring, Collaborations and Awards

Journal Papers

Collaborations

[6] Karlsruhe Institute of Technology, Karlsruhe Germany (08/July/2019 - 23/March/2020).
Doctoral stay.
Institut für Technik der Informationsverarbeitung (ITIV).
(Institute for Information Processing Technology).
Research topic: Dehancement Mode Field Effect Transistor design, Power-aware ambipolar fpga architecture. Developing a new type of Silicon-On-Insulator transistor (SOI). Analysis of electrical characteristics, body bias control, process, voltage and temperature (PVT) conditions. Developing hardware for machine learning.

Awards