



CENICS 2012

The Fifth International Conference on Advances in Circuits, Electronics and Micro-electronics

ISBN: 978-1-61208-213-4

August 19-24, 2012

Rome, Italy

CENICS 2012 Editors

Sergey Yurish, IFSA - Barcelona, Spain

Pascal Lorenz, University of Haute Alsace, France

CENICS 2012

Foreword

The Fifth International Conference on Advances in Circuits, Electronics and Micro-electronics [CENICS 2012], held between August 19-24, 2012 in Rome, Italy, continued a series of events initiated in 2008, capturing the advances on special circuits, electronics, and micro-electronics on both theory and practice, from fabrication to applications using these special circuits and systems. The topics cover fundamentals of design and implementation, techniques for deployment in various applications, and advances in signal processing.

Innovations in special circuits, electronics and micro-electronics are the key support for a large spectrum of applications. The conference is focusing on several complementary aspects and targets the advances in each on it: signal processing and electronics for high speed processing, micro- and nano-electronics, special electronics for implantable and wearable devices, sensor related electronics focusing on low energy consumption, and special applications domains of telemedicine and ehealth, bio-systems, navigation systems, automotive systems, home-oriented electronics, bio-systems, etc. These applications led to special design and implementation techniques, reconfigurable and self-reconfigurable devices, and require particular methodologies to be integrated on already existing Internet-based communications and applications. Special care is required for particular devices intended to work directly with human body (implantable, wearable, eHealth), or in a human-close environment (telemedicine, house-oriented, navigation, automotive). The mini-size required by such devices confronted the scientists with special signal processing requirements.

We take here the opportunity to warmly thank all the members of the CENICS 2012 Technical Program Committee, as well as the numerous reviewers. The creation of such a high quality conference program would not have been possible without their involvement. We also kindly thank all the authors who dedicated much of their time and efforts to contribute to CENICS 2012. We truly believe that, thanks to all these efforts, the final conference program consisted of top quality contributions.

Also, this event could not have been a reality without the support of many individuals, organizations, and sponsors. We are grateful to the members of the CENICS 2012 organizing committee for their help in handling the logistics and for their work to make this professional meeting a success.

We hope that CENICS 2012 was a successful international forum for the exchange of ideas and results between academia and industry and for the promotion of progress in the field of circuits, electronics and micro-electronics.

We are convinced that the participants found the event useful and communications very open. We also hope the attendees enjoyed the historic charm Rome, Italy.

CENICS 2012 Chairs:

Peeter Ellervee, Tallinn University of Technology, Estonia

Martin Horauer, University of Applied Sciences Technikum Wien, Austria

Josu Etxaniz Marañon, University of the Basque Country / Universidad del País Vasco / Euskal Herriko Unibertsitatea - Bilbao, Spain

Adrian Muscat, University of Malta, Malta

Vladimir Privman, Clarkson University - Potsdam, USA

Falk Salewski, Lacroix Electronics, Germany

Ravi M. Yadahalli, PES Institute of Technology & Management - Karnataka, India

Sergey Y. Yurish, Technical University of Catalonia (UPC-Barcelona), Spain

Yulong Zhao, Xi'an Jiaotong University, China

CENICS 2012

Committee

CENICS Advisory Chairs

Vladimir Privman, Clarkson University - Potsdam, USA
Sergey Y. Yurish, Technical University of Catalonia (UPC-Barcelona), Spain
Martin Horauer, University of Applied Sciences Technikum Wien, Austria
Adrian Muscat, University of Malta, Malta

CENICS 2012 Research/Industry Chairs

Ravi M. Yadahalli, PES Institute of Technology & Management - Karnataka, India

CENICS 2012 Industry Liaison Chairs

Falk Salewski, Lacroix Electronics, Germany

CENICS 2012 Special Area Chairs

Formalisms

Peeter Ellersee, Tallinn University of Technology, Estonia

Application-oriented

Josu Etxaniz Marañon, University of the Basque Country / Universidad del País Vasco / Euskal Herriko Unibertsitatea - Bilbao, Spain

Sensors

Yulong Zhao, Xi'an Jiaotong University, China

CENICS 2012 Technical Program Committee

Amir Shah Abdul Aziz, TM Research & Development, Malaysia
Said Al-Sarawi, The University of Adelaide, Australia
Mohammad Amin Amiri, Iran University of Science and Technology, Iran
Lotfi Bendaouia, ETIS-ENSEA, France
Javier Calpe, University of Valencia, Spain
Jose Carlos Meireles Monteiro Metrolho, Polytechnic Institute of Castelo Branco, Portugal
David Cordeau, LAII-IUT Angoulême, France
Marc Daumas, Université de Perpignan, France
Stefano Del Sordo, IASF - INAF (Istituto Nazionale di Astrofisica), Italy
Javier Diaz-Carmona, Technological Institute of Celaya, Mexico
Gordana Jovanovic Dolecek, Institute INAOE - Puebla, Mexico
Peeter Ellersee, Tallinn University of Technology, Estonia

Ykhlef Fayçal, Centre de Développement des Technologies Avancées, Algeria
Sérgio Adriano Fernandes Lopes, Universidade do Minho, Portugal
Francisco V. Fernández, IMSE, CSIC and University of Sevilla, Spain
Luis Gomes, Universidade Nova de Lisboa, Portugal
Petr Hanáček, Brno University of Technology, Czech Republic
Martin Horauer, University of Applied Sciences Technikum Wien, Austria
Emilio Jiménez Macías, University of La Rioja, Spain
Kenneth Blair Kent, University of New Brunswick, Canada
Tomas Krilavicius, Vytautas Magnus University - Kaunas & Baltic Institute of Advanced Technologies - Vilnius, Lithuania
Junghee Lee, Georgia Institute of Technology, USA
Kevin Lee, Murdoch University, Australia
Alie Eldin Mady, University College Cork (UCC) - Cork, Ireland
Cesare Malagu', University of Ferrara and Istituto di acustica e sensoristica Orso Maria Corbino CNR-IDASC, Italy
José Carlos Metrôlho, Instituto Politécnico de Castelo Branco, Portugal
Tarek Mohammad, University of Western Ontario - London, Canada
Bartolomeo Montrucchio, Politecnico di Torino, Italy
Adrian Muscat, University of Malta, Malta
Arnaldo Oliveira, Universidade de Aveiro, Portugal
Adam Pawlak, Silesian University of Technology - Gliwice, Poland
Angkoon Phinyomark, Prince of Songkla University, Thailand
Eduardo Correia Pinheiro, Instituto de Telecomunicações - Lisboa, Portugal
Anton Satria Prabuwono, Universiti Kebangsaan Malaysia, Malaysia
Vladimir Privman, Clarkson University - Potsdam, USA
Càndid Reig, University of Valencia, Spain
Falk Salewski, Lacroix Electronics, Germany
Arvind K. Srivastava, NanoSonix Inc., USA
Ivo Stachiv, Institute of Physics, Academia Sinica - Taipei, Taiwan
Ephraim Suhir, University of California – Santa Cruz, USA
Felix Toran, European Space Agency, Germany
Francisco Torrens, Institut Universitari de Ciència Molecular / Universitat de València, Spain
Manuela Vieira, UNINOVA/ISEL, Portugal
Chin-Long Wey, National Central University, Taiwan
Jianwu Xu, University of Chicago, USA
Ravi M. Yadahalli, PES Institute of Technology & Management - Karnataka, India
Jianhua (Joshua) Yang, Hewlett Packard Laboratories - Palo Alto, USA
Sergey Y. Yurish, IFSA, Spain
David Zammit-Mangion, University of Malta – Msida, Malta

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission to reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

A 2.45 GHz CMOS Voltage Controlled Ring Oscillator for Active Transponder <i>Jubayer Jalil, Mamun Bin Ibne Reaz, Labonnah Farzana Rahman, Mohammad Marufuzzaman, and Mohammad Syedul Amin</i>	1
Novel High-Speed and Ultra-Low-Voltage CMOS NAND and NOR Domino Gates <i>Yngvar Berg and Omid Mirmotahari</i>	5
A Novel High Speed Differential Ultra Low-Voltage CMOS Flip-Flop for High Speed Applications <i>Yngvar Berg</i>	11
Integration of Design Space Exploration into System-Level Specification Exemplified in the Domain of Embedded System Design <i>Falko Guderian and Gerhard Fettweis</i>	17
Unsupervised Image Segmentation Circuit Based on Fuzzy C-Means Clustering <i>Wen-Jyi Hwang, Zhe-Cheng Fan, and Tsung-Mao Shen</i>	23
Various Discussions and Improvements of Voltage Equalizer for EDLCs Including Secondary Batteries <i>Keiju Matsui, Kouhei Yamakita, and Masaru Hasegawa</i>	31
ASIP for Multi-Standard Video Decoding <i>Jae-Jin Lee, KyungJin Byun, and NakWoong Eum</i>	37
New Design Approach of an FIR Filters Based FPGA-Implementation for a Bio-Inspired Medical Hearing Aid <i>Lotfi Bendaouia, Si Mahmoud Karabernou, Lounis Kessal, Hassen Salhi, and Faycal Ykhlef</i>	43
Reliable CMOS VLSI Design Considering Gate Oxide Breakdown <i>Kyung Ki Kim</i>	50

A 2.45 GHz CMOS Voltage Controlled Ring Oscillator for Active Transponder

Jubayer Jalil, Manum Bin Ibne Reaz, Labonnah Farzana Rahman,

Mohammad Marufuzzaman and Mohammad Syedul Amin

Department of Electrical, Electronic and Systems Engineering,
Universiti Kebangsaan Malaysia

43600 Bangi, Selangor, Malaysia

*jubayer.jalil@gmail.com, mamun.reaz@gmail.com, labonnah.deep@gmail.com,
marufsust@gmail.com, syedul8585@yahoo.com*

Abstract—An improperly designed voltage controlled oscillator (VCO) for radio frequency (RF) phase locked loop (PLL) simply degrades performance of wireless communication. This paper proposes a low power ring oscillator based VCO developed for 2.45 GHz operated active readerless RFID transponder compatible with IEEE 802.11b protocol. In favor of easy integration and implementation of the module in small die size, a 3-stage differential delay cell has been adopted to fabricate the proposed voltage controlled ring oscillator (VCRO). 0.18 μm CMOS process is used for designing the proposed VCRO with 1.8 V power supply. Simulated results show that the proposed VCRO will work in the tuning range of 2.32 - 2.85 GHz and dissipate only 11.25 mW of power at 2.45 GHz. Thus, the proposed VCRO will be a vital module for active readerless RFID transponder.

Keywords-VCRO; RFID; Transponder; CMOS; Differential.

I. INTRODUCTION

Radio-frequency identification (RFID) is a smart identification system, relying on storing and remotely retrieving data using devices called tags or transponders. The typical RFID system comprises one or several readers which communicate with many tags simultaneously. Nowadays, implementations of RFID systems are extensively introduced in the supply chain, public transportation and biomedical applications. The operating frequency ranges of current RFID systems established for international standards extend from 135 KHz to 2.45 GHz [1]. In the RFID systems, tags can be categorized generally into two types: Passive and Active based on the power source. Passive tags use the magnetic field of readers as a source of energy and thus communicate with the readers. Active tags are battery-powered devices that have an active transmitter onboard. Unlike passive tags, active tags generate RF energy by themselves and this autonomy from the reader means that they can communicate at long distances dissipating more power than their counterparts.

At present, RFID deployment in numerous applications is a key challenge for technologist due to multiple standardization issues and expensive vendor specific readers. Moreover, RFID tags operating in several bands—high-frequency (HF) (13.56 MHz), ultra-high-frequency (UHF) (860–915 MHz), and microwave band

(2.4 GHz), have limited operational range, less than 2m to maximum 9m [2]. To overcome these concerns, a concept of reader-less RFID system based on IEEE 802.11b or Wi-Fi technology has been proposed [3]. In that system, RFID transponder will be battery-powered active device and its operating frequency will be 2.45 GHz (unlicensed ISM band). Moreover, conventional reader will be replaced by wireless network interface card (WNIC) utilizing desktop computer or laptop to make the system generic. However, effective use of active transponder's power is undoubtedly a crucial issue to implement this reader-less RFID system successfully.

During its activation, a RF transceiver of operating in gigahertz range usually dissipates substantial amount of power. That is why a direct conversion RF transceiver was proposed to implement the readerless transponder [4]. In this analog transceiver, one of the major blocks is the frequency synthesizer or local oscillator, which is done typically by using phase lock loop (PLL). This PLL is composed of phase detector (PD), low pass filter (LPF), voltage controlled oscillator (VCO) and frequency divider shown in Fig. 1. In this type of PLL-based frequency synthesizer, the most power hungry module is VCO, which generates frequency and changes the oscillating frequency varying control voltage.

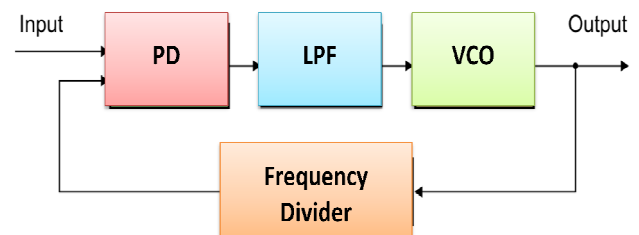


Figure 1. Block diagram of PLL based frequency synthesizer

Until now, LC-type and RC-type of CMOS VCOs have been used in wireless communication systems [5]. These VCOs performances are usually analyzed by low phase noise, low power dissipation, low voltage operation, high speed oscillation, multi-phase application, supply sensitivity reduction, simplified integration method, small layout area and wide tuning range. So far, LC based VCO has low level of phase noise among all CMOS VCOs [6]. However, it has narrow tuning range, greater power

dissipation and large die area [7]. In addition, it is very difficult to integrate inductor in digital CMOS technology [8]. These shortcomings of LC-VCO are overcome by ring based VCO or better known VCRO. Recently, VCRO are widely accepted not only in wireless communication but also in optical communication and many more applications of the emerging ultra-wide band (UWB) and wireless sensor networks (WSNs).

VCRO can be implemented by single-ended or differential architecture of delay cell. Usually a number of delay cell blocks are connected in a positive or regenerative feedback loop for building a ring oscillator (RO). In VCRO, single ended ring topology comprises of inverters and each inverter is made up of an NMOS and PMOS transistors. On the other hand, differential topology is made up of a load (active or passive) with a NMOS differential pair. Currently, differential circuit topology is getting popularity among designers as it has common-mode rejection of supply and substrate noise [9]. Moreover, it could be formed by odd or even number of stages and is possible to achieve both in-phase and quadrature outputs in DROs [10].

In this paper, a unique differential delay cell has been proposed in 0.18 μm CMOS process designing in DA-IC of Mentor Graphics environment. The novel delay cell will be used for the proposed VCRO of readerless RFID transceiver. While designing the module for 2.45 GHz operating frequency, power consumption should be reduced to improve the performance of the transponder. In this research work, it is focused on widening the tuning range and reduction of power of the VCRO.

This paper will be organized as follows: Section II discusses the details of oscillator design; Section III describes construction of delay cell and its operation; Section IV presents simulation results and comparisons with other works; a conclusion is drawn in Section V.

II. VCRO ARCHITECTURE

For incorporation of this ring oscillator, only three of differential amplifiers or inverter stages are connected in a single delay path formation as shown in Fig. 2. Several novel delay cells have been demonstrated to compose the two-stage ring VCO, but extra power is inevitably needed to provide an excess phase shift for oscillation satisfying Barkhausen criterion. On the other hand, implementation of 4-stage of RO consumes considerable amount of power. Though three-stage ring oscillator cannot produce quadrature outputs like 2-stage or 4-stage RO, nevertheless it is faster than its four-stage counterpart. Moreover, in three-stage RO, fulfillment of proper start-up conditions can easily be attained unlike even number ROs, where latch-up frequently occur. Thus, the use of 3-stage is chosen to increase the oscillation and reduce power consumption at the same time.

Principle operation of this oscillator is that if one of the nodes is excited, the pulse will propagate through all the stages and will reverse the polarity of the initially excited node. For start-up and oscillation criteria, the

transfer function for this ring oscillator with the number of stages set to 3 and can be represented as,

$$H(S) = \frac{-A_0^3}{\left(1 + \frac{S}{\omega_0}\right)^3} \quad (1)$$

where A_0 denotes voltage gain of each delay cell and ω_0 denotes 3dB bandwidth at each stage.

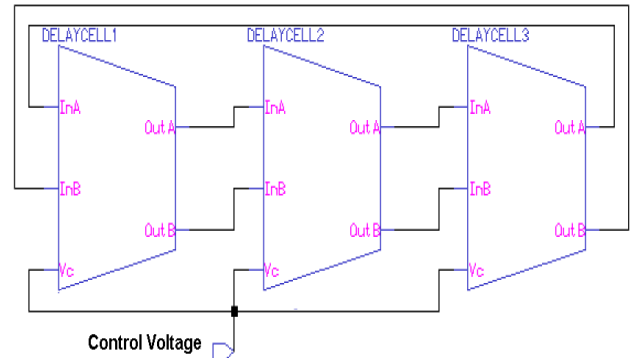


Figure 2. Architecture of the three-stage ring oscillator

As one of the criteria for oscillation is a phase shift of 180° that is each stage contributes with 60° of phase shift, the frequency at which it occurs given as,

$$\omega_{osc} = \omega_0 \cdot \tan\left(\frac{180^\circ}{N}\right) \quad (2)$$

The other criterion for oscillation is a loop gain greater than 1 at ω_{osc} . Thus, it has been calculated the minimum voltage gain per delay cell by inserting the oscillation frequency expression of (2) into the gain equation found from (1). By solving this calculation, yields the minimum voltage gain of 2 (two) for each delay cell.

For every signal cycle, there is a downward as well as an upward transition. Since the high-to-low (t_{pHL}) and low-to-high (t_{pLH}) propagation delays associated with these transitions are not usually equal, the average propagation delay is given by

$$T = \frac{(t_{pHL} + t_{pLH})}{2} \quad (3)$$

The oscillation frequency for an N -stage ring is derived from the average propagation delay (T) of the inverter. A propagating signal will have to pass twice through the chain of delay cells, for a total delay of $2NT$, to complete one period. Thus, the frequency of the oscillation (f) is expressed as,

$$f = \frac{1}{2NT} \quad (4)$$

III. PROPOSED DELAY CELL ARCHITECTURE

In this research, novel delay cell architecture for the VCRO has been proposed as shown in Fig. 3. The

proposed combination of the delay cell circuit is preferred as it alleviates necessity of tail current transistor caused flicker noise [11]. Additionally, it will improve output voltage stability without redundant bias circuit, which occupies a large space in chip.

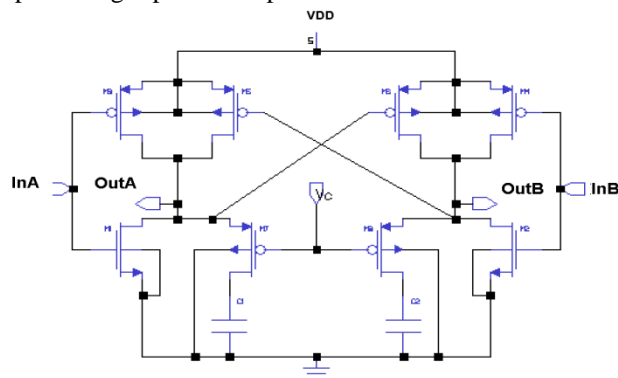


Figure 3. Schematic diagram of the proposed delay cell

A pair of CMOS differential push-pull inverter will be used as inputs in the new delay cell architecture, which is also shown in Fig.3. The push-pull inverter will consist of two different sizing of PMOS and NMOS. Additionally, two cross-coupled PMOS transistors will be connected in parallel with inverters PMOS transistors. These cross-coupled PMOS transistors will be introduced for fast switching speed. Sizes of all four PMOS in the cell will be chosen equally for smooth oscillation. In addition, a serially connected PMOS with a load capacitor of 0.1 pF will be employed in parallel with each NMOS input for frequency tuning.

The operation of the delay cell can be described considering half-cell circuit. While the input, *InA* will be high (near VDD), the input, *InB* will be low (equal to zero volt). This will turn on NMOS of the node, *InA*. On the other hand, PMOS of the input node, *InA* and cross-coupled PMOS connected in parallel with this input PMOS will remain off. Then voltage of the output node, *OutA* will be grounded. During that period, charge from the capacitor (*C_l*) will be discharged, or in other words, a path will be formed, which sinks current from *OutA* to bring its potential to 0 V. Similarly, if the input, *InA* will turn into 0 V, then the input, the input, *InB* will be high (near VDD). Zero potential of the input, *InA* will turn on PMOS and turn off NMOS simultaneously. Cross-coupled PMOS connected in parallel with the input PMOS of the node, *InA* will also remain switched on at this time. Thus, the discharged capacitor will be recharged again through these PMOS transistors. However, in both operations, a PMOS tuning transistor will control the overall charging and discharging of the load capacitor.

IV. SIMULATION RESULTS AND COMPARISONS

The proposed delay cell circuit has been verified by using the ELDO RF simulator (Mentor Graphics) of the CEDEC process. To determine the center frequency of the proposed delay cell circuit, the simulated output of the

VCRO is shown in Fig. 4. If the control voltage is set to 0.22 V, frequency of 2.45 GHz is achieved as shown in Fig. 4. The supply voltage is set to 1.8 V and the 0.1 pF load capacitor is selected in the circuit for reducing die area.

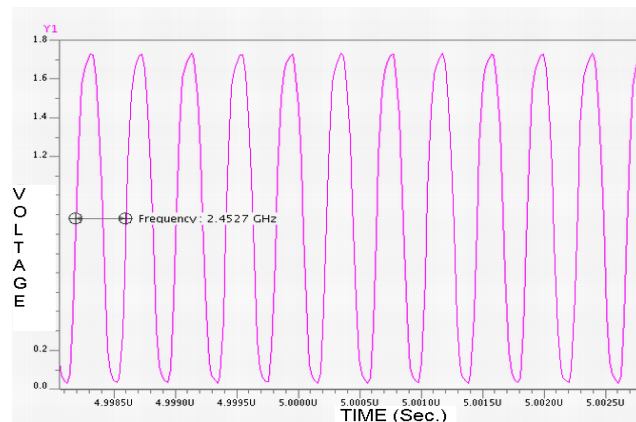


Figure 4. Simulated output of the proposed VCRO

In order to validate the proposed circuit in wide frequency range, the simulation is done at different control voltage. The output of different control voltages are shown in Fig. 5. In Fig. 5, it is being shown that if the control voltage is set to 0 V the proposed circuit is able to work in 2.32 GHz frequency. While VCRO's control voltage is increased to 1.1 V, the circuit oscillates in 2.85 GHz frequency. It is observed that by increasing the control voltage made the circuit working in higher frequency without changing the oscillation output voltage, i.e., the amplitude remains constant with increasing frequency. The voltage gain of VCO (*K_{vco}*) is given by

$$K_{vco} = \frac{f_{max} - f_{min}}{V_{max} - V_{min}} \quad (5)$$

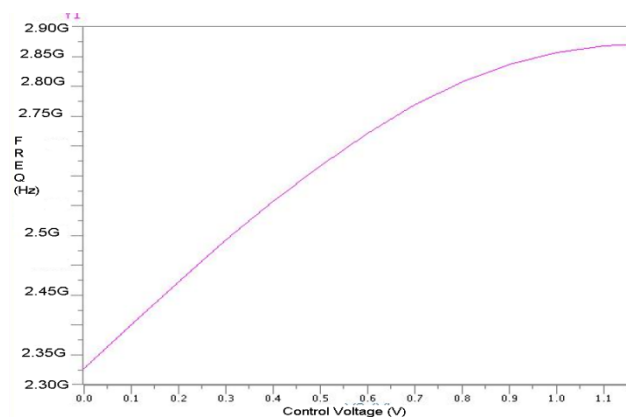


Figure 5. Simulated tuning range

A frequency-tuning ratio of 18.60% is attained from 2.32 GHz to 2.85 GHz. The gain of VCRO is achieved 480 MHz/V from (5). Since IEEE 802.11b protocol required 2.4 GHz to 2.5 GHz frequency, the proposed delay circuit will make the VCRO working on that frequency range

which will be certainly a key component of readerless RFID transponder. It exhibits a single side-band phase noise of -112 dBc/Hz at 10 MHz offset frequency from a center frequency of 2.45 GHz shown in Fig. 6.

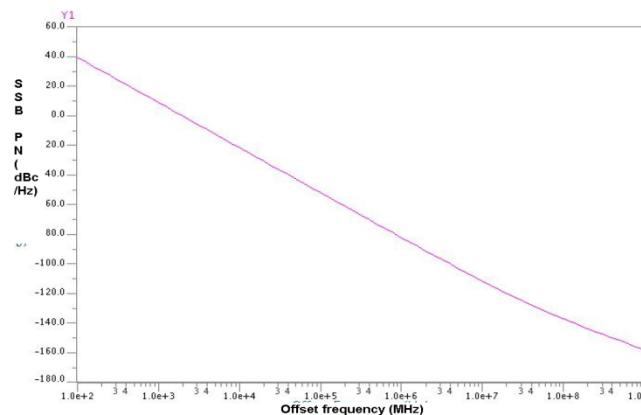


Figure 6. Simulated single side-band phase noise

TABLE I: PERFORMANCE COMPARISONS OF CMOS VCRO

Architecture	Center Frequency (GHz)	Tuning Range (GHz)	Supply Voltage (V)	Power (mW)	CMOS Process (μm)
4 stage, Dual delay loop [11]	0.9	0.75-1.2	3	-	0.6
2 stage, Single delay loop [12]	0.9	0.66-1.27	2.5	15.5	0.5
2 stage, Single delay loop [13]	0.9	0.73-1.43	1.8	65.5	0.18
4-stage, Dual delay loop [14]	-	1.77-1.92	1.8	13	0.18
3 stage, Single delay loop [This work]	2.45	2.2-2.85	1.8	11.25	0.18

Finally, the performance comparisons of CMOS VCRO of various technologies are shown in Table 1. Compared to other research works, it is shown that the proposed VCRO dissipates lowest power, which is around 11.25 mW and can operate in very high frequencies than others.

V. CONCLUSION AND FUTURE WORK

Despite the continuous improvement in the-state-of-the-art of CMOS VCOs, these devices still remain the most key blocks of RF PLLs. In this paper, a ring VCO has been proposed developed for active readerless RFID transponder. The simulated results showed that its operating frequency is 2.45 GHz, which will be compatible with the readerless RFID transponder and able to work with IEEE 802.11b protocol. In future, the research will be concentrated in the area of improving phase noise to increase the signal-to-noise ratio as well as improving figure of merit (FOM) of VCRO.

ACKNOWLEDGMENT

The authors would like to express sincere gratitude to the Ministry of Science Technology and Innovation (MOSTI) for supporting this research project through its MOSTI/BGM/R&D/20 (Brain Gain Malaysia) and UKM-AP-ICT-20-2010 (Arus Perdana) program.

REFERENCES

- [1] Y. S. Hwang and H. C. Lin, "A new CMOS analog front end for RFID tags," *IEEE Trans. Ind. Electron.*, vol. 56, no. 7, pp. 2299-2307, July 2009.
- [2] V. Pillai, H. Heinrich, D. Dieska, P. V. Nikitin, R. Martinez, and K. V. S. Rao, "An ultra-low-power long range battery/passive RFID tag for UHF and microwave bands with a current consumption of 700 nA at 1.5 V," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 7, pp. 1500-1512, July 2007.
- [3] F. R. Labonnah, M. B. I. Reaz, M. A. M. Ali, Mohd. Marufuzzaman, and M. R. Alam, "Beyond the WiFi: Introducing RFID system using IPv6", in *Proc. of the Kaleidoscope: Beyond the Internet? - Innovations for Future Networks and Services, ITU-T*, pp.1-4, 13-15 December 2010, Pune, India.
- [4] J. Jalil, M. B. I. Reaz, M. S. Amin, F. R. Labonnah, and Mohd. Marufuzzaman, "Development of 2.45 GHz analog front-end for readerless active RFID transponder", in *Proc. of Regional Engineering Postgraduate Conference (EPC)*, 6 pages, 4-5 October 2011, UKM, Bangi, Malaysia.
- [5] M. Moghavvemi and A. Attaran, "Performance review of high-quality-factor, low-noise, and wideband radio-frequency LC-VCO for wireless communication," *IEEE Microw. Mag.*, vol. 12, no. 4, pp. 130-146, June 2011.
- [6] O. Casha, I. Grech, and J. Micallef, "Comparative study of gigahertz CMOS LC quadrature voltage-controlled oscillators with relevance to phase noise," *Analog Integr. Circ. Sig. Process.*, vol. 52, no. 1-2, pp. 1-14, 2007.
- [7] M. Moghavvemi and A. Attaran, "Recent advances in delay cell VCOs," *IEEE Microw. Mag.*, vol. 12, no. 5, pp. 110-118, August 2011.
- [8] B. Leung, "A novel model on phase noise in ring oscillator based on last passage time," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 3, pp. 471-482, March 2004.
- [9] A. Hajimiri, S. Limotyrakis, and T. H. Lee, "Jitter and phase noise in ring oscillators," *IEEE J. of Solid-State Circuits*, vol. 34, no. 6, pp. 790-804, June 1999.
- [10] Y. Toh and J. A. McNeill, "Single-ended to differential converter for multiple-stage single-ended ring oscillators," *IEEE J. of Solid-State Circuits*, vol. 38, no. 1, pp. 141-145, January 2003.
- [11] C. H. Park and B. Kim, "A low-noise, 900-MHz VCO in 0.6-μm CMOS," *IEEE J. of Solid-State Circuits*, vol. 34, no. 5, pp. 586-591, May 1999.
- [12] W. S. T. Yan and H. C. Luong, "A 900-MHz CMOS low-phase-noise voltage-controlled ring oscillator," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 48, no. 2, pp. 216-221, February 2001.
- [13] Z. Q. Lu, J. G. Ma, and F. C. Lai, "A low-phase-noise 900-MHz CMOS ring oscillator with quadrature output," *Analog Integr. Circ. Sig. Process.*, vol. 49, no. 1, pp. 27-30, 2006.
- [14] Z. Z. Chen and T. C. Lee, "The design and analysis of dual-delay-path ring oscillators," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 3, pp. 470-478, March 2011.

Novel High-Speed and Ultra-Low-Voltage CMOS NAND and NOR Domino Gates

Yngvar Berg
Institute of Microsystems Technology
Vestfold University College
Horten, Norway
Email: Yngvar.Berg@hive.no

Omid Mirmotahari
Department of Informatics
University of Oslo
Oslo, Norway
Email: omidmi@ifi.uio.no

Abstract—In this paper we present novel ultra-low-voltage and high-speed CMOS NAND and NOR gates. For supply voltages below 500mV the delay for an ultra-low-voltage NAND2 gate is approximately 10% of a complementary CMOS inverter. Furthermore, the delay variations due to mismatch are much lesser than for conventional CMOS. Differential domino gates for AND2/NAND2 and OR2/NOR2 operation are presented. Ultra-low-voltage pass transistors are presented which can be used as latching gates. The ultra-low-voltage gates presented are going to be used for implementation of low-voltage and high speed adders.

Keywords-Low-Voltage, High-Speed, NAND2, NOR2, CMOS, Floating-Gate

I. INTRODUCTION

The aggressive scaling of device dimensions to achieve greater transistor density and circuit speed results in substantial subthreshold and gate oxide tunneling leakage currents. Energy efficiency is one of the most required features for modern electronic systems designed for high-performance and/or portable applications. In recent years, the power problem has emerged as one of the fundamental limits facing the future of CMOS integrated circuit design. On one hand, the ever increasing market segment of portable electronic devices demands the availability of low-power building blocks that enable the implementation of long-lasting battery-operated systems. On the other hand, the general trend of increasing operating frequencies and circuit complexity, in order to cope with the throughput needed in modern high-performance processing applications, requires the design of very-high-speed circuits.

Depending upon the application, there are numerous methods that can be used to reduce the power consumption of VLSI circuits [1], [2], these can range from low-level measures based upon fundamental physics, such as using a lower power supply voltage or using high-threshold voltage transistors; to high-level measures such as clock-gating or power-down modes. The power consumption in digital circuits, which mostly use complementary metal-oxide semiconductor (CMOS) devices, is proportional to the square of the power supply voltage; therefore, voltage scaling is one of the important methods used to reduce power consumption. In order to achieve a high transistor drive current and thereby improve the circuit performance,

the transistor threshold voltage V_t must be scaled down in proportion to the supply voltage. However, a decrease in the transistor threshold voltage V_t results in significant increase in the subthreshold leakage current.

Floating-Gate (FG) gates have been proposed for Ultra-Low-Voltage (ULV) and Low-Power (LP) logic [3]. However, in modern CMOS technologies there are significant gate leakages which undermine non-volatile FG circuits. FG gates implemented in a modern CMOS process require frequent initialization to avoid significant leakage. By using floating capacitances, either poly-poly, MOS or metal-metal, to the transistor gate terminals the semi-floating-gate (SFG) nodes can have a different DC level than provided by the supply voltage headroom [3]. There are several approaches to FG CMOS logic [4], [5]. The gates proposed in this paper are influenced by ULV non-volatile FG circuits[5].

In this paper we focus on implementation of low-voltage and high-speed Boolean gates. In section II an extended description of the ULV inverter [6] is given. In section III ULV NOR and NAND gates are presented and ULV latching pass transistors are described in section IV. Alternative implementations for Boolean gates are presented in section V and a conclusion is given in section VI.

II. ULTRA-LOW-VOLTAGE SEMI-FLOATING-GATE LOGIC

The ULV logic styles presented in this paper are related to the ULV domino logic style presented in [6]. The main purpose of the ULV logic style is to increase the current level for low supply voltages without increasing the transistor widths. We may increase the current level compared to complementary CMOS using different initialization voltages to the gates and applying capacitive inputs. The extra loads represented by the floating capacitors are lesser than extra load given by increased transistor widths. The capacitive inputs lower the delay through increased transconductance while increased transistor widths only reduce parasitic delay.

The simple dynamic edge and level ULV inverters [6] are shown in Figure 1. In order to retain a logic 1 a) when the input remain at logic 0 the width of the pMOS precharge transistor E_p is 4 times the minimum width while the nMOS evaluate transistor has minimum width. The width of the pMOS evaluate transistor in b) is 2 times minimum and the

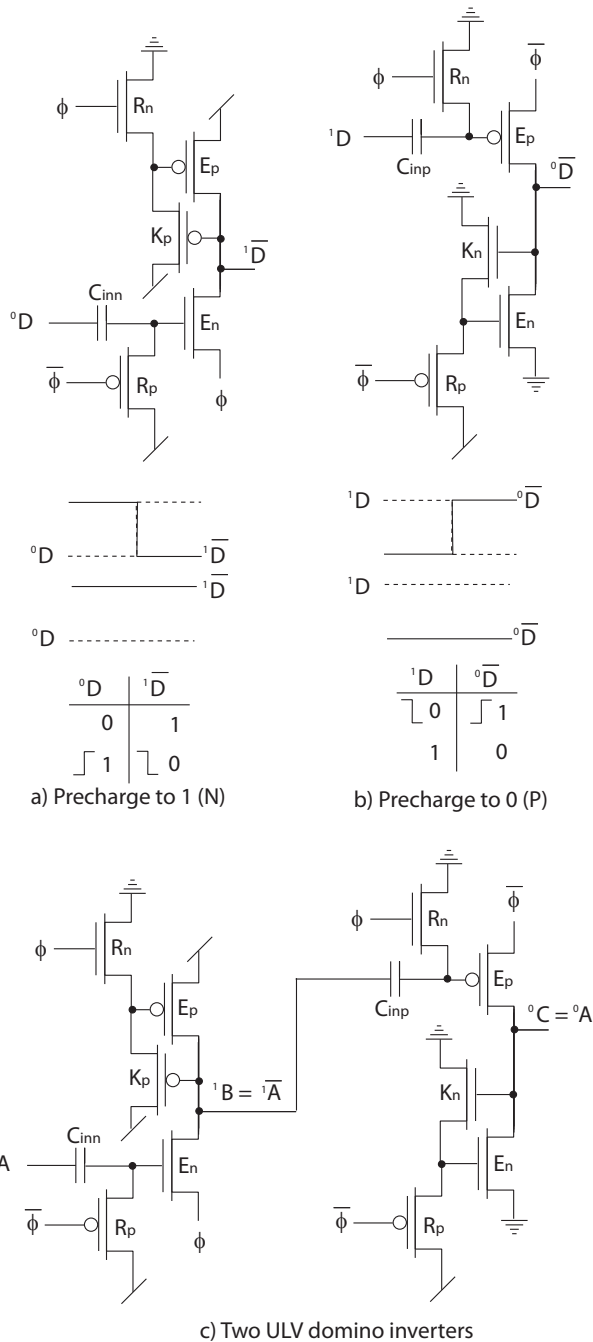


Figure 1. ULV domino inverters.

precharge nMOS transistor E_n is also 2 times minimum. The ULV domino gates in this report are ratioed logic and the size of the precharge transistors may be increased to secure required robustness or noise margin. The time constant of a false falling or rising voltage is however always significantly larger than the time constant for an active output edge, i.e. the problem will only be evident in very long domino chains. The recharge and evaluation mode of the ULV logic are

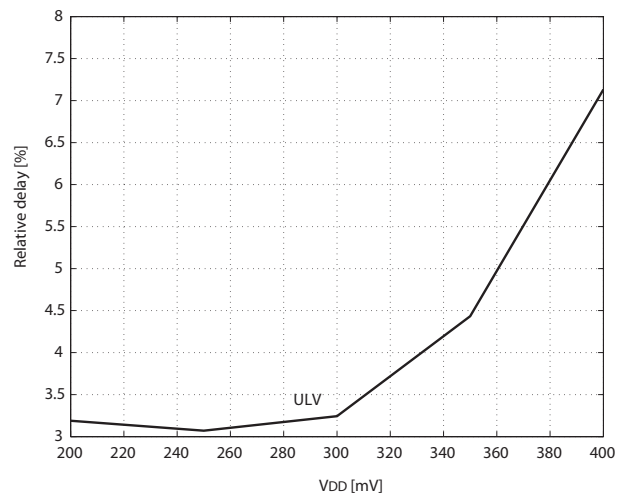


Figure 2. Delay for ULV logic styles relative to a CMOS inverter.

characterized by:

- **Recharge.** The precharge and recharge phase starts when ϕ switches from 0 to 1. The recharge transistors, labeled R, are turned ON and will recharge the gate of the evaluating transistors labeled E. More specifically, the gate of the nMOS evaluating transistors will be forced to V_{DD} and the gate of the pMOS evaluating transistors will be recharged to gnd .
- **Precharge.** $\phi = 1$. The output of the inverter in Figure 1 will be driven to V_{DD} or 1 and the inverter in b) will be precharged to 0.
- **Evaluate.** In the evaluation phase, determined by $\phi = 0$, the recharge transistors are turned OFF and the gate of the evaluating transistors are temporarily floating allowing an input transition to affect the current running through the transistors.

The ULV logic styles may be used in critical subcircuits where high-speed and low supply voltage is required. The ULV logic styles may be used together with more conventional CMOS logic. A ULV high speed serial carry chain [7] has been presented using a simple dynamic ULV logic [8]. In this paper we exploit an NP domino ULV static differential logic style.

We define a signal D precharged to 0 as 0D and a signal precharged to 1 as 1D . We Apply a clock signal to power the inverter, i.e. either ϕ to E_n and V_{DD} to E_p , or $\bar{\phi}$ to E_p and GND to E_n and precharge to 1 or 0 respectively. The gate resembles NP, i.e. precharge to 0 and precharge to 1, domino logic. In order to hold the precharged value until an input transition arrives the E transistor connected to a supply voltage is made stronger than the other E transistor. The function of the inverter can be described as ${}^0D \rightarrow {}^1\bar{D}$ and ${}^1D \rightarrow {}^0\bar{D}$.

Relative delays for ULV inverters compared to Standard

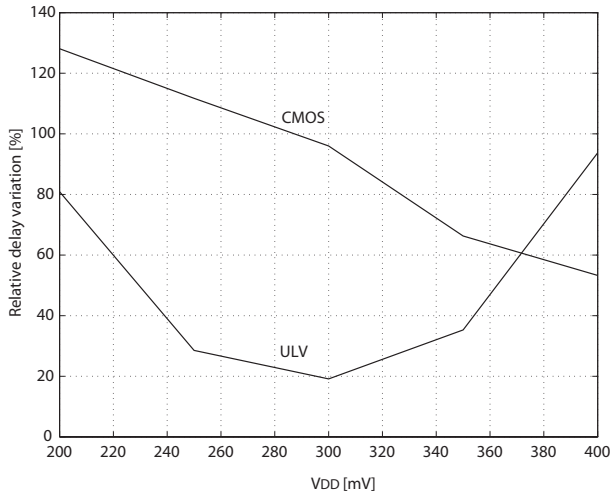


Figure 3. Delay variation due to process mismatch.

CMOS inverters are shown in Figure 2. For supply voltages in the region from 200 to 400 the delays of the different ULV logic styles presented are less than 8% of standard CMOS delay. The main target for the logic style presented is 300mV which will yield 96% delay reduction compared to standard CMOS. A typical application for the ULV logic styles are low voltage serial adders. For a supply voltage equal to 300mV we may apply a 32-bit carry chain using the ULV logic with the same delay as a one-bit standard CMOS carry gate. Delay for different ULV inverters relative to complementary CMOS inverter are shown in Figure 2. The delay improvement is more significant for the proposed ULV inverters than for the original ULV inverters for supply voltages below 320mV due to reduced capacitive load.

The ULV logic style is defined by the applied terminal inputs as shown in TABLE I. The ON and OFF currents of a complementary CMOS inverter is given by the effective gate source voltages V_{DD} and 0V respectively. Assuming $\frac{C_{in}}{C_T} = 0.5$ where C_T is the total capacitance seen by a floating gate, we may estimate the delay, dynamic and static power and noise margins of the different ULV logic styles relative to a complementary CMOS inverter.

Monte Carlo simulation is performed including process mismatch and the results in terms of delay variations are shown in Figure 3. For the ULV logic style the mismatch of the clock drivers (standard CMOS inverters) are included. The delay variations of the clock drivers will be equal to the standard CMOS inverters which is significantly larger than the ULV inverters. Hence, the delay variations of the clock drivers will not affect the delay variations significantly.

III. ULV NOR AND NAND GATES

The ULV domino NOR2 gate is shown in Figure 4. The function is defined by ${}^1O = {}^0A + {}^0B$ and reveals a Boolean NOR2 function. The function can be defined in terms of

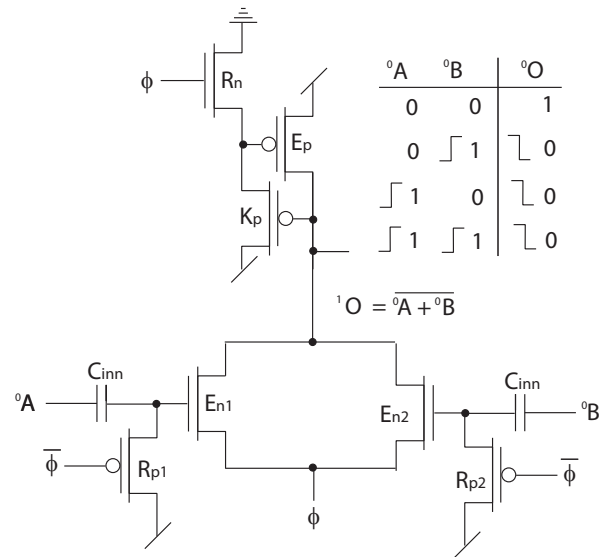


Figure 4. ULV domino NOR2 gate.

edges and in this context the function is OR2, i.e. for any input edges the output will provide an edge. In order to retain a logic 1 when both inputs remain at logic 0 the width of the pMOS precharge transistor E_p is 8 times the minimum width. The increased width of the precharge transistor and the added parallel evaluate transistor will increase the delay by close to a factor 2 compared to an ULV domino inverter. The worst case scenario is when one and only one of the inputs 0A or 0B switches to 1 and the other remains at 0.

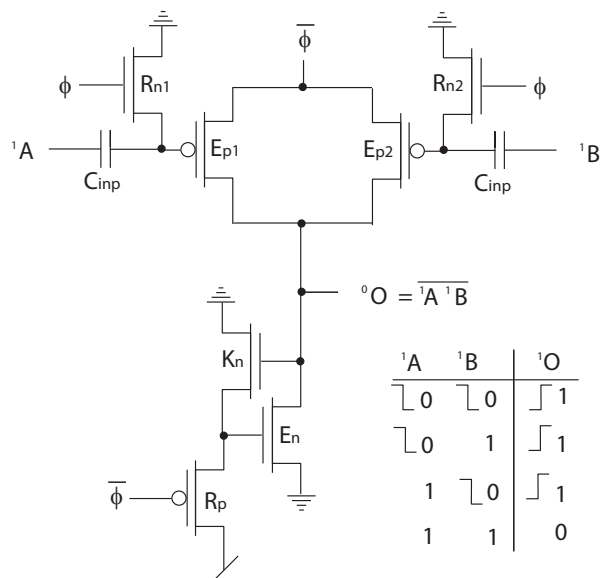


Figure 5. ULV domino NAND2 gate.

The ULV domino NAND2 gate is shown in Figure 5. The function is defined by ${}^0O = \overline{{}^1A \cdot {}^1B}$ and reveals a Boolean NAND2 function. The function can be defined in terms of

ΔV	E_p	E_n	$V_{gs} I_{ON}$	$V_{gs} I_{OFF}$	NM'	Relative delay	Comment
V_{DD}	$\bar{\phi}$	GND	$\frac{3V_{DD}}{2}$	$\frac{V_{DD}}{2}$	V_{DD}	$\approx 5\%$	Precharge to 0
$-V_{DD}$	V_{DD}	ϕ	$\frac{3V_{DD}}{2}$	$\frac{V_{DD}}{2}$	V_{DD}	$\approx 5\%$	Precharge to 1

Table I

ULV logic styles. ΔV is the output voltage swing. The simple model for the noise margin NM' is given by the ratio of the ON current and the OFF current given by the effective gate to source voltage. The capacitive division factor, $\frac{C_{inn}}{C_T}$ where C_T is the total capacitance seen by a floating gate is assumed to be 0.5. The delay is relative to a standard complementary CMOS inverter.

edges and in this context the function is OR2, i.e. for any input edges the output will provide an edge.

IV. ULV PASS TRANSISTORS

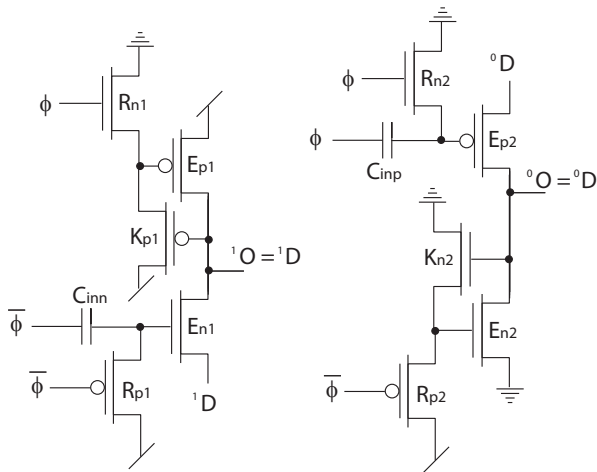


Figure 6. ULV latching pass transistors.

Precharge pass transistors are shown in Figure 6. The circuits can be used in ULV latches and Flip-Flops. The evaluate transistors E_{n1} and E_{p2} are powered by the input signals 1D and 0D , and the inputs signals are pushed to the output by the clock (edge) $\bar{\phi}$ and ϕ . The delay from the input to the output of the pass transistor gate is less than for an inverter. By using a combination of the ULV domino inverter and the ULV pass transistor we can implement different Boolean functions.

V. ALTERNATIVE BOOLEAN CIRCUITS

In this section we employ both ULV domino inverters and ULV pass transistors. The Boolean functions are implemented using two or more stages. Furthermore, the Boolean function of the gates can be defined in terms of standard Boolean logic levels or in terms of signal edges.

By using the evaluate transistor both as an inverting device and a pass transistor as shown in Figure 7 the ULV gates can be used as AND and OR gates. The Boolean function of the OR gate on the left is $^1O = ^1A + ^1B$. We assume that $^0\bar{A}$ is generated by an ULV domino inverter as shown in Figure 1 b). The OR gate provide a Boolean OR function.

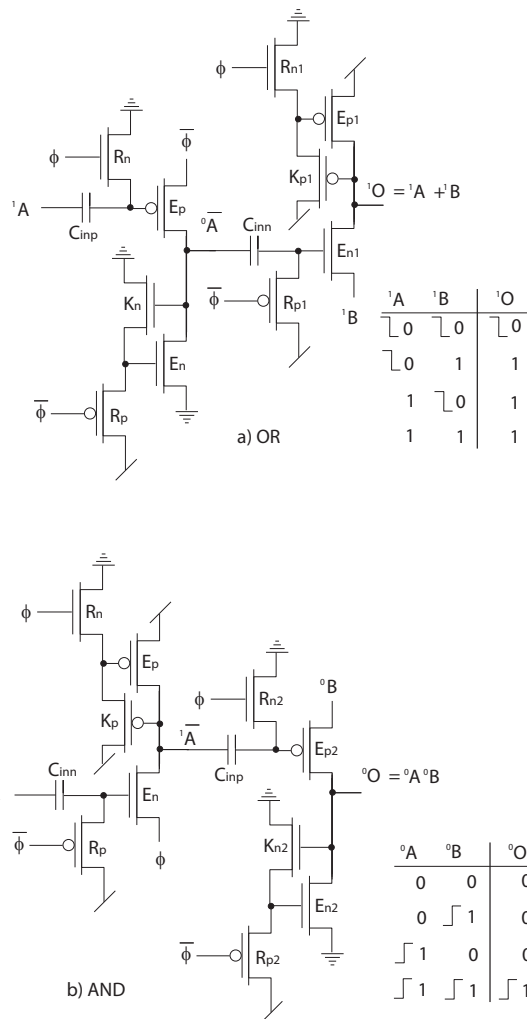


Figure 7. ULV domino and pass transistor AND2 and OR2 gates.

The function can also be defined in terms of edges. In this context the function is AND, i.e. an output transition will occur if and only if both input provide edges. For the AND gate on the left the function is given by $^0O = ^0A^0B$. In the edge context the function is still AND.

An alternative NOR2 gate is shown in Figure 8 and an alternative NAND2 gate is shown in Figure 9. These gates are slightly different than the previous gates. Both inputs are connected to the gate by floating capacitors which will

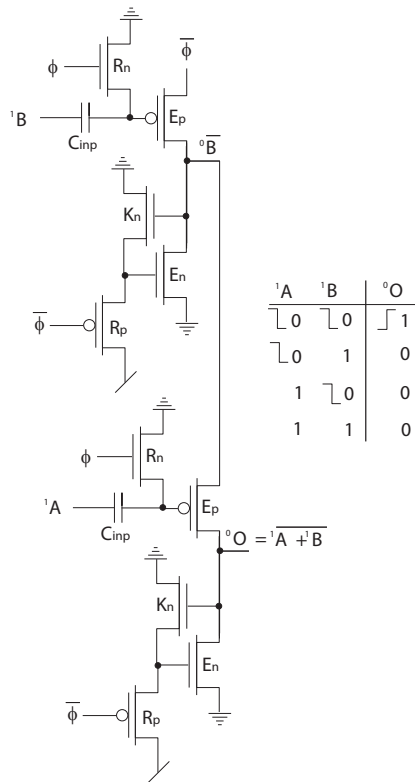


Figure 8. Alternative ULV domino NOR2 gate.

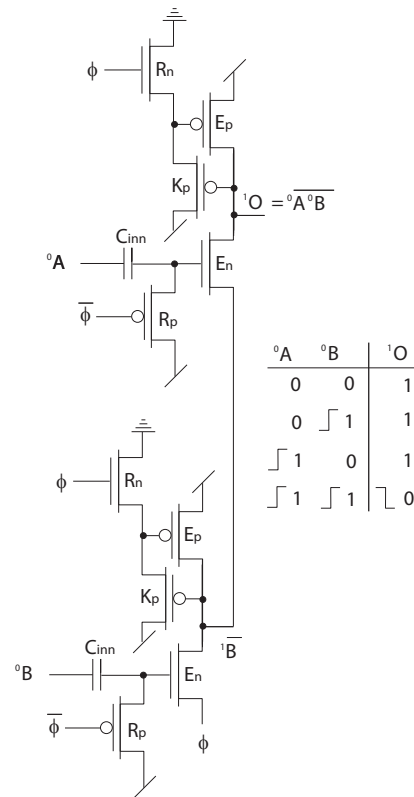


Figure 9. Alternative ULV domino NAND2 gate.

Logic style	C_{load}	delay	Comment
Figure 1	8C	≈ 4%	Inverter
Figure 4	13C	≈ 7%	NOR2
Figure 5	13C	≈ 7%	NAND2
Figure 6	7C	≈ 3%	Pass
Figure 7	10C + 9C	≈ 10%	AND2
Figure 8	7C + 8C	≈ 7%	NAND2

Table II

Capacitive load and worst case relative delay for a supply voltage equal to 200mV (compared to a CMOS inverter). C is equivalent to the gate or parasitic diffusion capacitance of a minimum-sized transistor.

prevent draining current from the gates providing the input signals. These gates will be more symmetrical in terms of delay from each input to the output. The delay from the inputs to the output of the gates shown in 7 are different, i.e. the delay from the inputs ⁰B and ¹B are significantly less than from the ⁰A and ¹A. This asymmetrical property is helpful when the delay for the inputs is different due to different signal paths. If the gate is used in a carry chain the carry signal should be provided through a pass transistor as shown in 7.

Capacitive load and relative delay compared to a standard CMOS inverter for the different gates proposed are presented in Table II. The delay of the ULV domino inverter is lesser than 4% compared a CMOS inverter for supply voltages less

than 330mV as shown in Figure 2. The delay of the ULV pass transistor is less than for the ULV domino inverter. The different implementations of the Boolean gates are equal in terms of delay and close to two times the delay of the ULV domino inverter.

VI. CONCLUSION

Different ultra-low-voltage domino NAND and NOR gates have been presented. The ULV two-input domino Boolean gates are high-speed, i.e. the delay compared to a CMOS inverter is less than 10%. The delay variation of the ULV gates due to process mismatches is much less than for a CMOS inverter operating at the same supply voltage. The ultra-low-voltage gates presented are going to be used to implement low-voltage and high-speed adders. Preliminary results show that the delay for the ULV NAND2 and NOR2 gates are less than 10% of the delay for a complementary CMOS inverter for ultra low supply voltages.

REFERENCES

- [1] Chandrakasan A.P. Sheng S. Brodersen R.W.: "Low-power CMOS digital design" , *IEEE Journal of Solid-State Circuits*, Volume 27, Issue 4, April 1992 Page(s):473 - 484
- [2] Verma N. Kwong J. Chandrakasan A.P.: "Nanometer MOSFET Variation in Minimum Energy Subthreshold Circuits" , *IEEE Transactions on Electron Devices*, Vol. 55, NO. 1, January 2008 Page(s):163 - 174

- [3] Y. Berg, D. T. Wisland and T. S. Lande: "Ultra Low-Voltage/Low-Power Digital Floating-Gate Circuits", *IEEE Transactions on Circuits and Systems*, vol. 46, No. 7, pp. 930–936, July 1999.
- [4] K. Kotani, T. Shibata, M. Imai and T. Ohmi. "Clocked-Neuron-MOS Logic Circuits Employing Auto-Threshold-Adjustment", *In IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 320-321, 388, 1995.
- [5] T. Shibata and T. Ohmi. "A Functional MOS Transistor Featuring Gate-Level Weighted Sum and Threshold Operations", *In IEEE Transactions on Electron Devices*, vol 39, 1992.
- [6] Y. Berg and O. Mirmotahari: "Ultra Low-Voltage and High Speed Dynamic and Static Precharge logic", *In Proc. of the 11th Edition of IEEE Faible Tension Faible Consommation*. June 6-8, 2012, Paris, France.
- [7] Y. Berg "Ultra Low Voltage Static Carry Generate Circuit", *In Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, Paris, May 2010.
- [8] Y. Berg: "Static Ultra Low Voltage CMOS Logic", *In Proc. IEEE NORCHIP Conference*, Trondheim, NORWAY, November 2009.

A Novel High Speed Differential Ultra Low-Voltage CMOS Flip-Flop for High Speed Applications

Yngvar Berg

Institute of Microsystems Technology

Vestfold University College

Horten, Norway

Email: Yngvar.Berg@hive.no

Abstract—In this paper we present a simple ultra low-voltage and high speed D flip-flop. The Flip-Flop may be used in any standard digital low-voltage CMOS applications. Furthermore, the ultra low-voltage Flip-Flop offers reduced data to output delay compared to conventional CMOS Flip-Flops. Different master latch configurations are presented and a differential symmetric ultra low-voltage Flip-Flop is presented. Simulated data using *HSpice* and process parameters for 90nm CMOS are provided. Preliminary results show that the proposed Flip-Flop has a delay less than 20% compared to a conventional CMOS Flip-Flop.

Keywords—CMOS, low-voltage, Flip-Flop, high-speed, Floating-Gate.

I. INTRODUCTION

The ever increasing problem associated with modern CMOS processes is the demand for digital CMOS gates operating at low supply voltages. The available supply voltage and threshold voltage is lowered as a consequence of the reduction in transistor length. When the supply voltage is decreased the speed of the logic circuits may be reduced due to reduced effective input voltage to the transistors. When the threshold voltage is reduced the off current running through transistors which are switched off will increase and thereby increase static power consumption and reduce noise margins. Voltage scaling reduces the active energy and unfortunately speed as well. Low voltage applications are often dominated by low speed and low energy requirements, typical battery-powered electronics. The optimal supply voltage for CMOS logic in terms of energy delay product (EDP) is close to the threshold voltage of the nMOS transistor V_{tn} for the actual process, assuming that the threshold voltage of the pMOS transistor V_{tp} is approximately equal to $-V_{tn}$ [1]. Several approaches to high speed and low voltage digital CMOS circuits have been presented [2][3][4].

Floating-gate (FG) CMOS gates have been proposed for ultra low-voltage (ULV) and low power (LP) logic [5]. However, in modern CMOS technologies there are significant gate leakages which undermine non-volatile FG circuits. FG gates implemented in a modern CMOS process require frequent initialization to avoid significant leakage. By using floating capacitances to the transistor gate terminals the

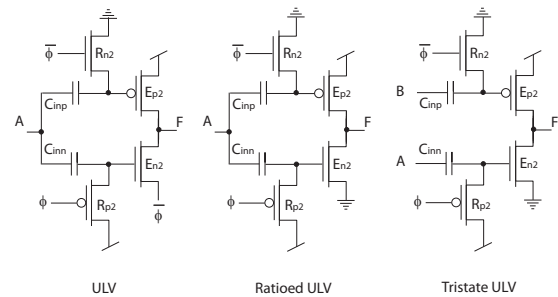


Figure 1. The ULV logic.

semi-floating-gate (SFG) nodes can have a different DC level than provided by the supply voltage headroom [5].

The ULV logic [6], [7] gates can be operated at a clock frequency more than 10 times than the maximum clock frequency of a similar complementary CMOS gate operating at the same supply voltage. For high clock frequencies, the switching energy consumed by the ULV gate will be reduced compared to a complementary gate.

In this paper we present an ultra low-voltage flip-flop (UFF) using ULV CMOS logic. The UFF offers a significant speed improvement compared to conventional sense amplifier FF (SAFF's) [8] hereafter called FF1. In section II a short introduction to ultra low-voltage logic is given. The simple UFF is described in section III [9]. Four new master configurations are presented in section IV including a low-power version. Symmetric ultra low-voltage differential FF's are presented in section V with simulated results using *HSpice* simulator and 90nm TSMC process.

II. ULTRA LOW VOLTAGE LOGIC

The original ULV inverter and ULV NP domino inverters are shown in Figure 1. The recharge phase starts when the clock signal ϕ switches from 0 to 1. Assuming a NPULV P ϕ inverter there are two different situations dependent on the state of the gate. First, assume that the output is 1 or close to 1, the nMOS floating gate is close to $V_{offset+}$ and the pMOS floating-gate is close to $V_{offset-}$ due to a static input in the previous evaluation phase. In this case the

only work to be done is a marginal refresh of the floating-gates and the output. Secondly, assume that the output is 0, the nMOS floating gate is close to $V_{offset+} + k_{in}V_{DD}$ and the pMOS floating-gate is close to $V_{offset-} + k_{in}V_{DD}$ due to a positive input transition in the previous evaluation phase. In this case the output needs to be pulled to 1 and this is done by the pMOS and nMOS transistors in parallel. Notice that the nMOS E_{n2} is positive biased at the time of the clock edge and will contribute significantly to pull the output from 0 to 1. When the output is getting close to 1 the recharged pMOS evaluate transistor E_{p2} will pull the output to 1. The nMOS floating-gate will initially have a potential of $V_{offset+} + k_{in}V_{DD} \approx 1.5 \times V_{DD}$ and a positive current will flow to $V_{offset+}$ or V_{DD} while the nMOS floating-gate will be recharged through a negative current drawn from $V_{offset-}$ or gnd . Simulation shows that the time required to precharge the NPULV logic is two to three times the raise and fall times for different supply voltages.

The evaluation phase starts when the clock signal ϕ switches from 1 to 0. In the evaluation phase there are two different situations depending on the input. If the input is stable, i.e. no transition, during the evaluation phase the output will remain close to 1. The circuit will in this situation consume significant static current. The static current is dependent on the applied offset voltages $V_{offset-}$ and/or $V_{offset+}$ as well. Assuming a positive input transition, the floating-gates will be moved by $k_{in}V_{DD}$ and the output will be pulled down to 0 in a similar manner as a complementary inverter. The active current will be larger due to the boost of the evaluate transistors.

The ULV inverters shown in Figure 1 recharge simultaneously when $\phi = 1$. The precharge level is different, the P_ϕ precharges to 1 and while the output of the N_ϕ precharges to 0. The P_ϕ gate is susceptible to a positive input transition when the evaluate phase starts, i.e. $\phi = 0$.

III. SIMPLE ULTRA LOW VOLTAGE FLIP-FLOP

The transistor counts for the Flip-Flops presented in this paper are less than for conventional Flip-Flops. The layout area is dependent on the implementation of the floating capacitors. The capacitance values for the floating capacitors are typically less than $1fF$, and hence can be implemented using MOS transistor parasitic capacitances and metal-metal. The accuracy of the floating capacitors is not critical and high level metal can be used.

The simple ULV Flip-Flop is shown in Figure 2 [9]. The input D is loaded onto QMP and QMN when $\phi = 0$. When ϕ switches from 0 to 1, one of the evaluate transistors E_{n1} or E_{p1} will be activated due to a boosted voltage level of QMN or QMP . If $D = 0$ then QMP will be pulled down to $\approx -V_{DD}/2$ and a large current provided by the E_{p1} transistor will be used to set the output of a slave latch and Flip-Flop to $Q = \bar{D}$. If $D = 1$ then QMN will be pulled up to $\approx 3V_{DD}/2$ and a large current provided by the E_{n1}

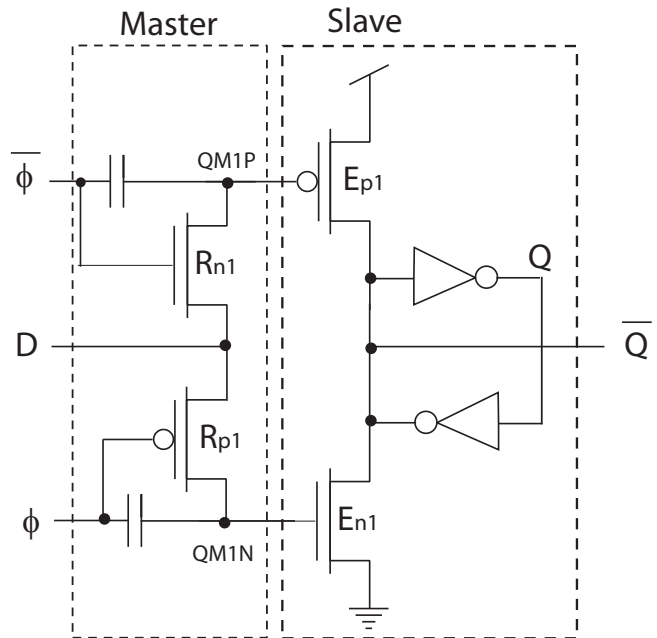


Figure 2. Basic inverting ULV Flip-Flop.

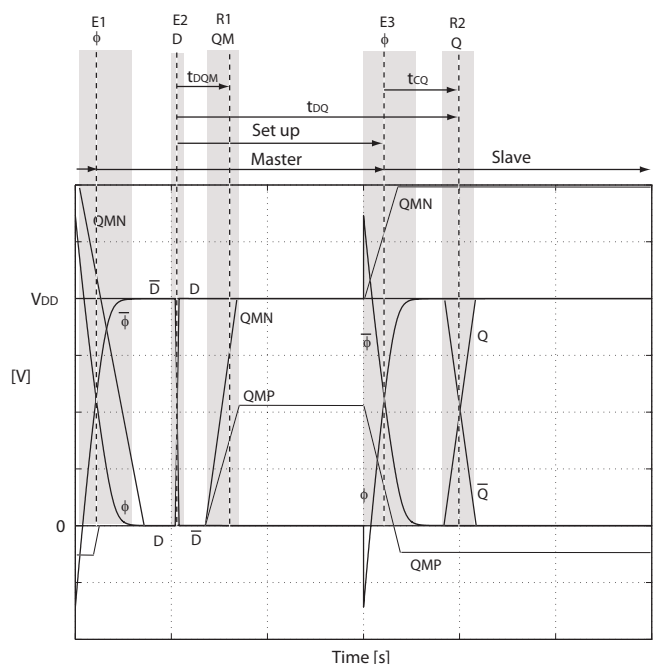


Figure 3. Basic ULV Flip-Flop timing.

transistor will be used to set the output of the Flip-Flop to $Q = \bar{D}$. The timing of the ULV Flip-Flop is shown in Figure 3. There are some critical events and timing restrictions that are important for performance of the ULV Flip-Flop:

- 1) E_1 . Clock signal ϕ switches from 1 to 0. Any change in D will be loaded to QMP and QMN . If $D = 0$

then $QMP = D$ and $QMN \approx D$, and if $D = 1$ then $QMN = D$ and $QMP \approx D$. The slave latch will not be influenced by any changes in QMP or QMN due to the inverters at the output. The inverters must be strong enough to hold a stable output value when the master latch is transparent.

- 2) E_2 . Input D is stable. The critical timing restriction is the setup time and thus only dependent on the delay through a pass transistor.
- 3) R_1 . QMN and QMP will be set to D . More specifically, $V_{QMN} = V_{DD}$ if $D = 1$ and $V_{QMN} \approx V_{DD}/2$ if $D = 0$, and $V_{QMP} = 0V$ if $D = 0$ and $V_{QMP} \approx V_{DD}/2$ if $D = 1$.
- 4) $E_2 \rightarrow R_1$. Data to QM delay t_{DQM} .
- 5) E_3 . Clock signal ϕ switches from 0 to 1. One of the evaluate transistors are activated through a boosted QM voltage. The activated evaluate transistor will drive the output Q to \overline{D} because the current level provided by the evaluate transistor is significantly larger than the current level of the inverters. This is the only event that may trigger the slave latch and determine the output of the Flip-Flop. The master latch becomes non active.
- 6) R_2 . Slave latch will respond to QMN or QMP and set the slave latch output $Q = D$ and $\overline{Q} = \overline{DB}$.
- 7) $E_1 \rightarrow E_3$. The master latch is active and output Q is stable due to the cross coupled inverters at the output.
- 8) $E_2 \rightarrow E_3$. Setup time for the input. This is the only significant delay of the ULV Flip-Flop.
- 9) $E_2 \rightarrow R_2$. Data to output time.
- 10) $E_3 \rightarrow E_1$. The slave latch is active and master latch is non active.

The simple ULV master latch is shown in Figure 2. D is loaded onto $QM1P$ and $QM1N$ when $\phi = 0$. More specifically if $D = 1$ then $QM1N = D = 1$ and $QM1P < D$ due to the body effect, and if $D = 0$ then $QM1P = D = 0$ and $QM1N > D$. When the clock signal ϕ switches from 0 to 1 the recharge transistors R_{n1} and R_{p1} closes and the capacitive input will increase the voltage level of $QM1N$ and decrease the voltage level at $QM1P$. In the case of D being equal to 1 the resultant voltage at $QM1N$ is $\gg V_{DD}$ hence $\gg D$ and $QM1P$ is ≈ 0 hence $\approx \overline{D}$. In this case the nMOS transistor E_{n1} is more enhanced than the E_{p1} transistor and the output \overline{Q} will be pulled to $0 = \overline{D}$ through a large current provided by E_{n1} . The QM nodes will be floating until the next event determined by the clock switching from 1 to 0. Hence, the E_{n1} and E_{p1} transistors will be ON until this event and contribute to power consumption. One or both of these transistors should be turned OFF to save power while the slave is active.

The most critical timing issue in the ULV Flip-Flop is the setup time of the master latch. The elevated current level of the evaluation transistors in a slave latch, i.e. slave latch in Figure 2, will pull the output quickly to the right value. Typically, the clock to output delay is negative due to the extremely low rise and fall time of the output \overline{Q} .

IV. MASTER LATCHES

In this section we present new master latch configurations aimed for ultra low-voltage applications including a novel low power version. The master latches presented are different from previously published ULV Flip-Flops [9] in several aspects. The inputs to the Flip-Flops are used to control the recharge transistors and are used to reduce the static power consumption. By adding transistors to increase the control of the semi floating-gates, i.e. QM voltages, we can turn off the non active evaluation transistors. Compared to the ULV Flip-Flop in [9] all the presented master latches and Flip-Flops described in this paper are low power.

Different implementations of ultra low voltage master latches are shown in Figure 4. The basic master latch is shown in Figure 2 where the input D is applied through pass transistors. In Figure 4 2) additional recharge transistors, labeled K_{n2} and K_{p2} , are applied to the QM nodes. The effect of these transistors is provide a way to turn the evaluate transistors off and thereby reducing the power consumption when $\phi = 1$. The $QM2N$ and $QM2P$ will be affected while the clock signal switches and the full effect of the signal through the floating capacitor may be reduced. In Figure 4 3) a differential input master latch is shown where we use the \overline{D} input to turn off the most active evaluate transistor. This configuration will not be as robust as the master latch in 2). The master latch shown in Figure 4 4) resembles the circuit in 2). The effect of the keeper transistors K_{n4} and K_{p4} will be delayed slightly compared to K_{n2} and K_{p2} and the effect of the signal applied to the floating capacitors are more evident. In Figure 4 5) the additional transistors are controlled by the output of the slave latch \overline{Q} . If the output $\overline{Q} = 1$ the K_{n5} transistor will be turned on and reduce the current running through transistor E_{n5} and hence reduce the power consumption and increase the noise margin of a slave latch.

Simulated responses for different master latches are shown in Figure 5. The supply voltage is $200mV$ and timing details for event E_1 and E_2 are shown. The master latches become active when ϕ switches from 1 to 0 and D is passed onto the QM nodes through the recharge transistors. At event E_2 the input changes from 0 to 1 and the QM are affected. $QM2N$, $QM2P$ and $QM4N$ will be pulled to 0 after right after the output of the slave is pulled to 0. Master latches 1), 3) and 5) require less set-up time than 2) and 4).

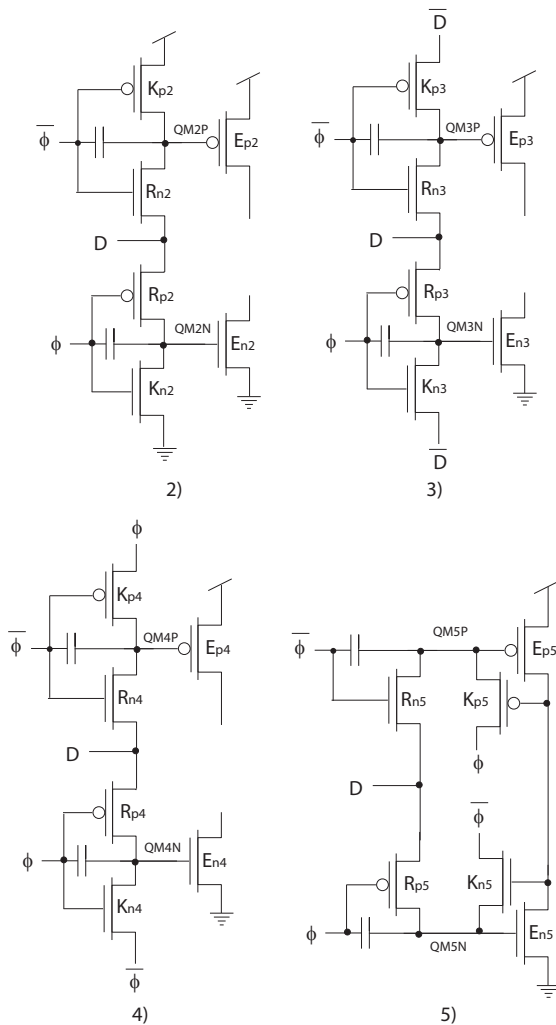


Figure 4. Different master latch configurations. The input D can be used to provide a reference to E_{p1} and E_{p2} , and \bar{D} can be used to provide a reference to E_{n1} and E_{n2} .

V. SYMMETRIC DIFFERENTIAL ULTRA LOW-VOLTAGE FLIP-FLOPS

A symmetric and differential ULV Flip-Flop is shown in Figure 6. The master latches are similar to that of Figure 4 2) and the slave latches resemble the basic slave latch shown in Figure 2. The presented Flip-Flop is different from the Flip-Flop presented in [9] by using the input D , and \bar{D} , to power the evaluation transistors E_{n1} , E_{n2} , E_{p1} and E_{p2} directly. This reduces the signal path from input to output of the ULV Flip-Flops described. The most critical timing issue of the master latches presented is the set-up time. By using the D and \bar{D} inputs directly to the evaluate transistors as if they were pass transistors the Flip-Flop will react more quickly because all evaluate transistors will the pull the outputs in the same direction.

In order to reduce the input load the evaluate transistors

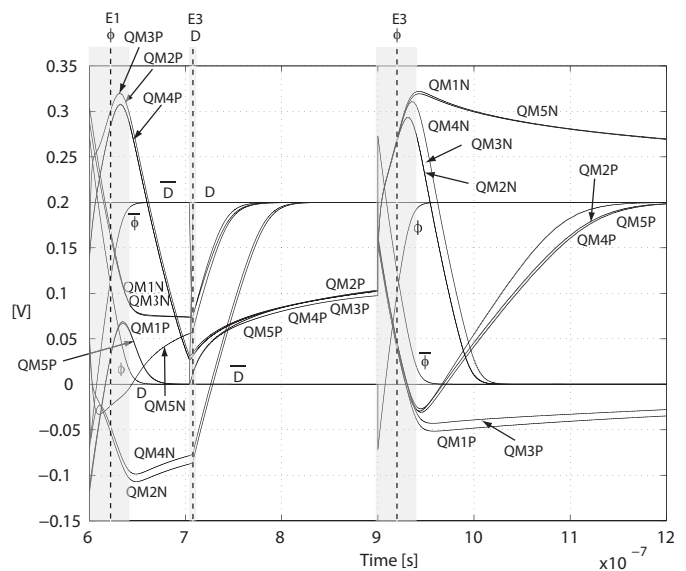


Figure 5. Simulated response for different master latches.

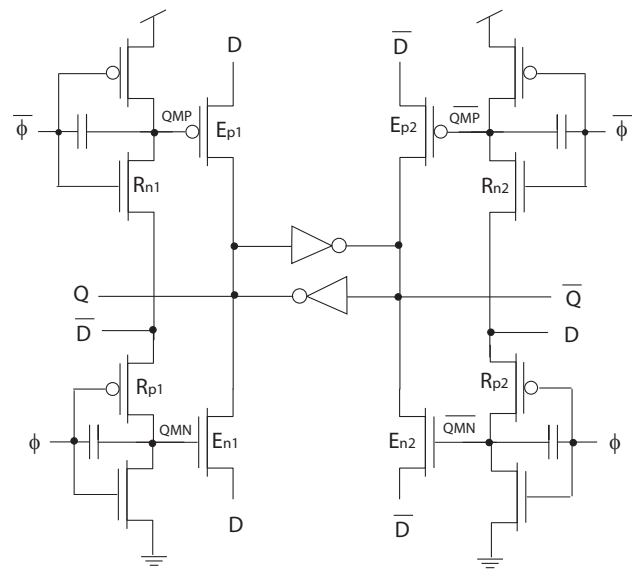


Figure 6. The symmetric high-speed ULV Flip-Flop.

E_{p1} and E_{p2} can be connected directly to V_{DD} , and E_{n1} and E_{n2} can be connected directly to gnd . This will only affect the response of the Flip-Flop slightly.

An alternative Flip-Flop with reduced input load and increased output- and clock load is shown in Figure 7.

A. Set-up details

In Figure 8 the set-up details for input $D = 1$ and $Q = 0$ are shown. The recharge transistor R_{n1} will pass the $\bar{D} = 0$ onto the gate of evaluate transistor E_{p1} . This node is labeled QMP . In the set-up phase QMP becomes 0 and QMN will be close to 0. Transistor E_{p1} will be activated when $\bar{\phi}$ switches from 1 to 0. At the same time the recharge transistor

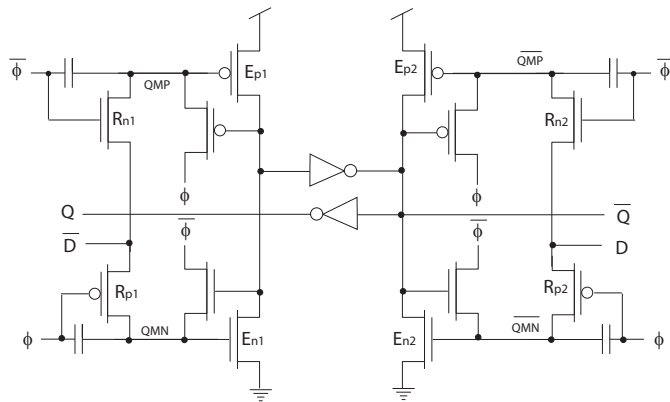


Figure 7. Alternativ symmetric and low power high-speed ULV Flip-Flop.

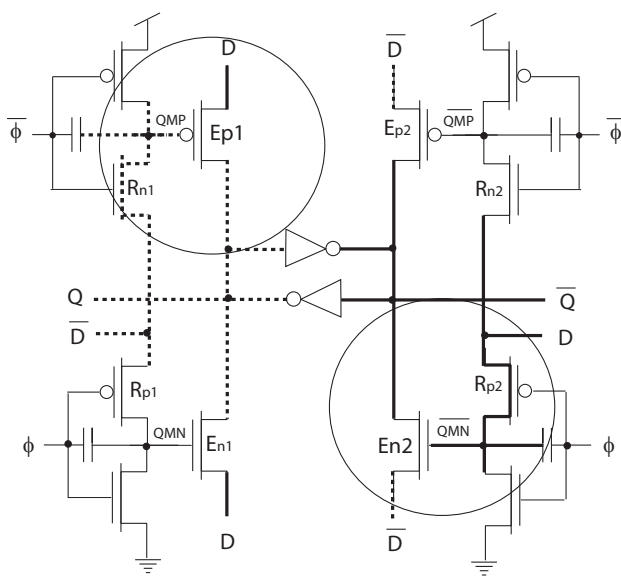


Figure 8. The symmetric high-speed ULV Flip-Flop set-up details.

R_{p2} will pass $D = 1$ onto the gate of transistor E_{n2} labeled QMN . Transistor E_{n2} will be activated when ϕ switches from 0 to 1. The transistors E_{p1} and E_{n2} force the output Q and \bar{Q} quickly to 1 and 0 respectively due to elevated current levels.

B. Simulated delay

V_{DD}	Conv.	Nik.	ULV
	t_{dq}	t_{dq}	$t_{setup} + t_{cq} = t_{dq}$
300mV	22.6ns	8.55ns	3.57ns
350mV	10.0ns	3.65ns	1.69ns

Table I
Simulated delay for conventional CMOS FF[10], Nikolic sense amplifier FF[8] and symmetric differential ULV FF.

Simulated delay, using Hspice and parameters for a

90nm CMOS (RSMC) process, for the symmetric differential ultra low-voltage Flip-Flop in Figure 6 for supply voltages 300mV and 350mV are given in Table I. The data to output delay, i.e. setup time and clock to output delay, is compared to data to output delay of a conventional CMOS Flip-Flop[10] and the Nikolic sense amplifier Flip-Flop[8].

VI. CONCLUSION

In this paper we have presented high-speed low-voltage static Flip-Flop and different master latch configurations. Different low power master latch configurations are presented. The data to output delay for the ultra low-voltage Flip-Flop is significantly reduced compared to conventional CMOS Flip-Flop and sense amplifier Flip-Flop. The Flip-Flops is designed for ultra low-voltage digital systems, i.e. supply voltages below 0.5V. Compared to conventional Flip-Flops the delay of the proposed Flip-Flop is reduced to less than 50%.

REFERENCES

- [1] Chandrakasan A.P. Sheng S. Brodersen R.W.: "Low-power CMOS digital design" , *IEEE Journal of Solid-State Circuits*, Volume 27, Issue 4, April 1992 Page(s):473 - 484
- [2] Verma N. Kwong J. Chandrakasan A.P.: "Nanometer MOSFET Variation in Minimum Energy Subthreshold Circuits" , *IEEE Transactions on Electron Devices*, Vol. 55, NO. 1, January 2008 Page(s):163 - 174
- [3] K. Usami and M. Horowitz: "Clustered voltage scaling technique for low-power design", *International Symposium on Low Power Electronics and Design (ISLPED)*, 1995, Pages: 3 - 8
- [4] Mutoh S., Douseki T., Matsuya Y., Aoki T., Shigematsu S., Yamada J.: "1-V power supply high-speed digital circuit technology with multithreshold-voltage CMOS" *IEEE Journal of Solid-State Circuits*, Volume 30, Issue 8, Aug. 1995 Page(s):847 - 854
- [5] Y. Berg, D. T. Wisland and T. S. Lande: "Ultra Low-Voltage/Low-Power Digital Floating-Gate Circuits", *IEEE Transactions on Circuits and Systems*, vol. 46, No. 7, pp. 930–936, July 1999.
- [6] Y. Berg, O. Mirmotahari, J. G. Lomsdalen and S. Aunet: "High speed ultra low voltage CMOS inverter", *In Proc. IEEE Computer society annual symposium on VLSI*, Montpellier France, April 2008.
- [7] Y. Berg: "Novel High Speed and Ultra Low Voltage CMOS Flip-Flops", *In Proc. IEEE International Conference on Electronics, Circuits and Systems ICECS*. 2010 ISBN 978-1-4244-8156-9. s. 298-301, Athens, Greece.
- [8] B. Nikolic, V.G Oklobdzija, V. Stojanovic, W. Jia, J. K.-S. Chiu and M. T.-T. Leung: "Improved Sense-Amplifier-Based Flip-Flop: Design and Measurements", *IEEE J. Solid-State Circuits*, vol. 35, pp.867-877, June 2006.

- [9] Y.Berg: "Differential static ultra low-voltage CMOS flip-flop for high speed applications", *In Recent researches in circuits, systems, mechanics and transportation systems : Proceedings of the 10th WSEAS International Conference on Circuits, Systems, Electronics, Control and Signal Processing (CSECS '11)*. Montreux, Switzerland, December 29-31, 2011. World Scientific and Engineering Academy and Society 2011 ISBN 978-1-61804-062-6. s. 134-137.
- [10] N.H.E Weste and D.M. Harris: *Integrated circuit design*", Fourth edition 2011, ISBN 10:0-321-69694-8, *Pearson*.

Integration of Design Space Exploration into System-Level Specification exemplified in the Domain of Embedded System Design

Falko Guderian and Gerhard Fettweis
 Vodafone Chair Mobile Communications Systems
 Technische Universität Dresden, 01062 Dresden, Germany
 Email: {falko.guderian, fettweis}@ifn.et.tu-dresden.de

Abstract—The specification of system functionality and design space exploration (DSE) are becoming very challenging in embedded systems due to an increasing number of design parameters and system specifications during the design cycle. An executable system-level specification (SLS), proposed in this paper, reduces design complexity. The SLS represents an executable DSE methodology and encapsulates system specifications. The aim is to formalize and automate design flows in order to scale to larger and more complex embedded systems. SLSs should not be limited to certain embedded system types. Hence, SLSs need to be standardized across tools, designers, and domains. A meta-methodology, as well as a meta-model are proposed to define a domain-independent SLS. Moreover, an electronic design automation environment is presented allowing to graphically create, automatically execute and validate embedded domain-specific SLSs. Finally, a design flow case study demonstrates multiple SLSs for the heterogeneous multicluster architecture.

Keywords—*embedded system design; system-level design; executable specification; design space exploration*

I. INTRODUCTION

Over the past decades, embedded design kept up with an increasing technology scaling through a continuous improvement and integration of computer aided design (CAD) tools. CAD tools evolved from the layout level to the logic level and later to the behavioral synthesis. Consequently, the next step was to develop system-level design tools, including the specification and exploration of complete systems. These advancements in CAD are closely coupled with the development of electronic design automation (EDA) flows. Early EDA flows were dominated by capturing and simulating incomplete specifications. Later, logic and register-transfer synthesis allowed to describe a design only from its behavior. But, a system gap between software (SW) and hardware (HW) designs exists since SW designers still provide HW designers with incomplete specifications [1].

An executable specification, such as a SystemC model [2], closes the system gap by describing the system functionality and enabling design space exploration (DSE) of various design alternatives. Design reuse and documentation are improved through executable specifications [1]. Nevertheless, the design complexity of future embedded systems with thousands of cores increases the number of available design parameters and system specifications during

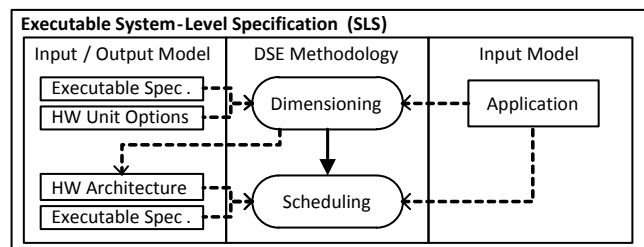


Fig. 1: Example of an executable system-level specification.

the design cycle [3]. In this work, system specifications include input / output models consumed / produced in the design steps, such as executable specifications, descriptions of application, architecture, application mapping, validation result, tool configuration, etc. So far, the challenging tasks of systems specification and defining a DSE methodology are decoupled. But the combined specification and the reuse of DSE methodologies promise for a reduced design complexity and design time, respectively. Hence, we believe that a system-level specification (SLS) needs to consider both the specification of systems and DSE, as exemplary seen in Figure 1. The specified DSE methodology includes two design steps. First, dimensioning creates a HW architecture from an executable specification, the HW unit options and application description. Then, DSE results are obtained from scheduling the application on the HW architecture.

This paper introduces an executable SLS which represents an executable DSE methodology and encapsulates system specifications. In other words, our work relates to a higher abstraction level of executable specifications. An SLS realizes a formalization and automation of design flows allowing to scale to larger and more complex embedded systems. In order to be not limited to certain embedded system types, SLSs will be standardized across tools, designers, and domains. Therefore, a meta-methodology, as well as a meta-model are proposed to define a domain-independent SLS.

In the remainder of the paper, Section II gives an overview about specification languages, related DSE environments, and meta-modeling activities. Section III introduces a conceptual framework generalizing SLS at a meta-level and a domain-level. At meta-level, a domain-independent SLS is proposed enabling interoperability across tools, designers, and domains. This SLS is described using a methodology about design methodologies and a model about design models. At domain-level, domain-specific SLSs are created following the proposed meta-methodology and meta-model. This allows to model various design flows applicable for em-

bedded systems with different characteristics, such as real-time, safety-critical, secure, fault-tolerant, robust, etc. In the section, a domain-specific SLS is illustrated via the λ -chart model [4]. In Section IV, an EDA environment is introduced realizing CAD support to build embedded domain-specific SLSs based on the λ -chart. DSE is automatically executed and validated as defined in the SLS. Then, Section V presents a design flow case study for the heterogeneous multicenter architecture built up from SLSs. Finally, the conclusions and open topics are discussed in Section VI.

II. RELATED WORK

The related work focuses on specification and DSE in embedded system design. First, selected specification languages and representative DSE environments are presented. Then, related studies on meta-modeling are discussed.

Specification Languages and DSE Environments

There is a variety of graphical and textual specification languages and frameworks. They can be used to realize DSE methodologies. Nevertheless, this is done in a less formal and less generic manner compared to our SLS approach. Hence, the reuse and interoperability across tools, designers, and domains are limited. An example is the specification and description language (SDL) [5] allowing for formal and graphical system specification and their implementation. In [6], HW/SW co-design of embedded systems is presented using SDL-based application descriptions and HW-emulating virtual prototypes. Moreover, SystemC [2] and SpecC [7] are system-level design languages (SLDL) which model executable specifications of HW/SW systems at multiple levels of abstraction. These simulation models support SW development. For example, SystemCoDesigner [8] enables an automatic DSE and rapid prototyping of behavioral SystemC models. In [9], a comprehensive design framework for heterogeneous MPSoC is presented. Based on the SpecC language and methodology, it supports an automatic model generation, estimation, and verification enabling rapid DSE. Another example is the specification in a synchronous language, e.g., via Matlab/Simulink. Instead, Ptolemy [10] supports various models of computation to realize executable specifications including synchronous concurrency models.

In addition, the MultiCube project [11] and the NASA framework [12] address the need of a generic infrastructure for system-level DSE mainly enabled by modularization. Nevertheless, the works present neither a domain-independent SLS nor a domain-specific SLS.

Meta-modeling

Our paper differs to existing work since it is the first using meta-modeling in order to describe a domain-independent SLS. In the embedded domain, meta-modeling has been studied to transform from the unified markup language (UML) to SystemC at the meta-model level [13]. This guarantees reuse of models and unifies a definition of the transformation rules. In [14], meta-modeling enables heterogeneous models of computations during modeling. In [15], meta-modeling is used to improve the model semantics and to enable type-

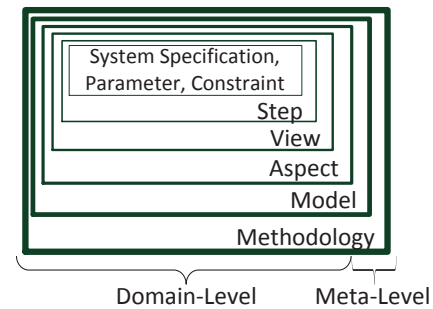


Fig. 2: System-level specification hierarchy.

checking and inference-based facilities.

III. CONCEPTUAL FRAMEWORK

As mentioned before, SLSs aim at reducing the design complexity. The SLS hierarchy, illustrated in Figure 2, gives a hierarchical understanding of SLSs. The abstraction is used as starting point of a formal description. This is realized by separating into a meta-level and a domain-level. The meta-methodology and meta-model allow for developing and testing a methodology and model for a specific design purpose. At the domain-level, specific design aspects, views, steps, system specifications, parameters, and constraints are chosen depending on the domain. That means, certain design tasks are realized in a design aspect using a domain-specific design methodology and design model. Each design aspect includes one or multiple design views modeling orthogonal design functionalities, such as communication, computation and administration infrastructure. Moreover, each design view follows a design process with several steps. Various system specifications, design parameters, and constraints are considered in the steps in order to realize the DSE methodology. Focusing on embedded system design, both levels will be explained in more detail.

A. Meta-Level

At the meta-level, a domain-independent SLS is described to be able to develop and evaluate domain-specific SLSs. Hence, the transfer of design skills gets independent on a design domain and can reach a larger audience. In addition, design concepts and formalisms will be reusable across different tools, designers, and domains. Figure 3-4 illustrate the proposed meta-methodology and meta-model.

In Figure 3, the meta-methodology represents a guiding procedure in order to transform the domain-independent SLS into a domain-specific SLS. It starts to create a separation of the design space into design aspects and a separation of the design aspects into steps. Design aspects divide the design space at a higher abstraction level, as seen in Figure 2. In contrast, a step, system specification, parameter and constraint represent a lower abstraction level. As mentioned before, the specification of design views allows to model orthogonal design functionalities. Referring to Figure 3, an executable DSE methodology is built through an algorithmic ordering of the design aspects and steps. That means, dependencies, loops, branches, etc. realize

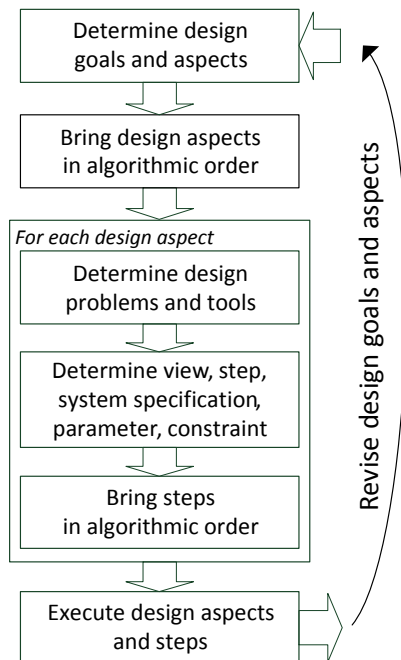


Fig. 3: Meta-methodology for the proposed SLS.

an execution order of aspects and steps in an algorithmic manner. Moreover, design tools are determined in all steps solving the relevant design problems. A design parameter represents a possible description of the structure, behavior, and physical realization of a system. Aiming at improved tool results, suitable design tool parameters are also considered. In each step, the design tools are parameterized and executed using the system specifications. From the DSE results, design goals and aspects can be revised. Finally, the design space is explored by varying the design parameters based on the algorithmic order and DSE strategy, such as exhaustive or heuristic search.

In Figure 4, the proposed meta-model, described via the UML class diagram, represents a model to build domain-specific SLSs. The meta-model forms a fundament or kernel of an EDA environment presented in Section IV. Hence, it includes the definition of the modeling language described via meta classes. Referring to Figure 4, an Element contains Properties and Transitions from/to Elements. A Transition between two Elements is used to model a unidirectional dependency and a Property represents a system specification, design parameter, design constraint, or additional information added to an Element. Moreover, an Aspect and Node inherit from Element. An Aspect includes one or several Nodes. Aspects can be nested to be resolved recursively. This allows to reduce model complexity and to improve the reuse of already modeled aspects. Finally, a Node represents an executable Element, such as a step, loop and branch node, which are necessary to build an algorithmic order of aspects and steps.

B. Domain-Level

In the following, an instantiation of a domain-specific SLS is illustrated with the help of the λ -chart [4] model, as

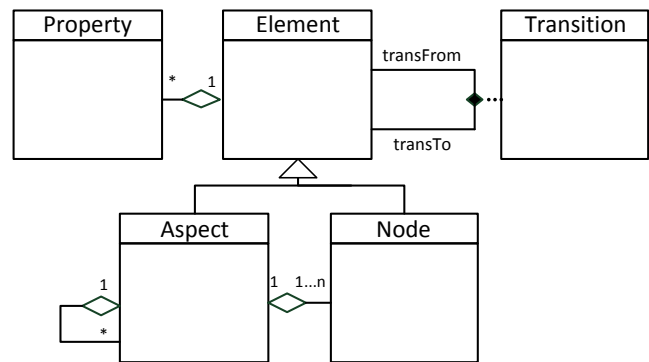


Fig. 4: Meta-model for the proposed SLS.

depicted in Figure 5 (left). The λ -chart models system-level design and exploration in the embedded system domain. As mentioned before, the proposed meta-methodology is used to select appropriate design aspects, views, steps, etc. In addition, an algorithmic order of the aspects and steps must be defined. The λ -chart model is an instance of the proposed meta-model. Referring to Figure 5 (left), a design aspect is represented by a λ -chart instance allowing to define steps in three design views. The administration view considers tasks for planning, monitoring, and control. Computation relates to code execution. Communication includes the design of data storage and data exchange between components. Furthermore, concentric bands underline the five steps of a unified design process. We refer to [4] for a detailed explanation of the λ -chart.

Referring to Figure 5 (left), the exemplary SLS starts with modeling and partitioning the design limited to the communication view. After scheduling and allocation, the DSE results are validated. The allocation and validation steps are iteratively traversed aiming at improved DSE. Similar to [16], the derived network-on-chip (NoC) aspect focuses on finding suitable NoC topology parameters, such as number of rows, columns, and modules per router. Furthermore, meta-model instantiation examples of the domain-specific SLS are illustrated in Figure 5 (right). The allocation step and loop node correspond to a node element in the meta-model. In allocation, exemplary properties are a “Rows” parameter and the communication view. Moreover, a transition from loop to allocation implies an algorithmic order realizing a part of the DSE methodology. The instantiation of an aspect is also shown.

C. Integration in Specification Languages

Specification languages, such as SDL and SystemC, do not currently support the proposed SLS. By doing so, an advantage would be to keep the system designers more aware of the design space in early design stages. The organization into design aspects helps the designer to cope with a complex system-level DSE. In addition, system designers need to structure and arrange their designs into design views. This brings greater attention to orthogonal system functionality, such as computation, communication, and administration. Given the design goals and constraints, it will be more evident that a systematic variation of design parameters is

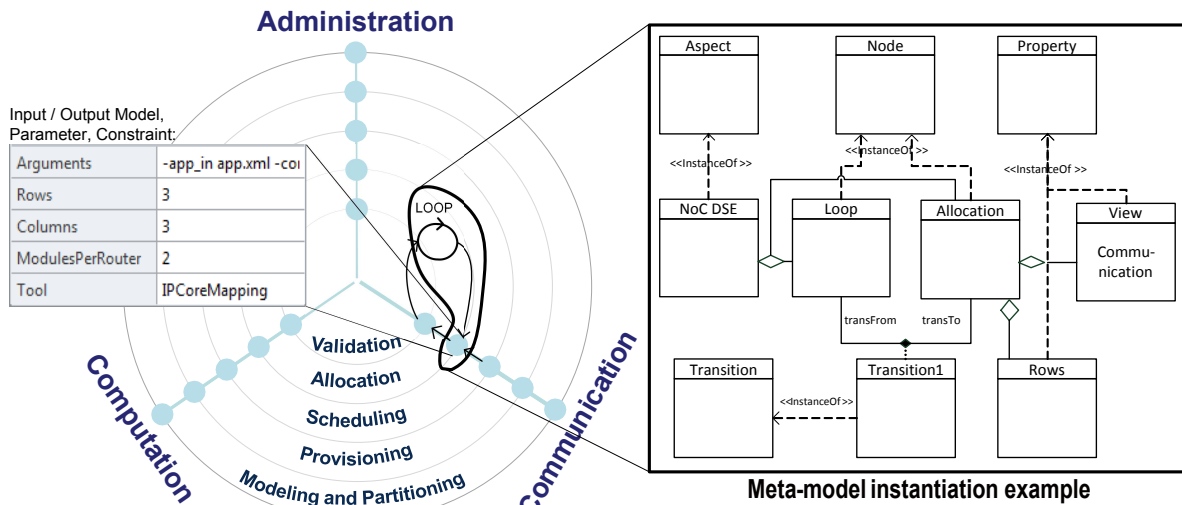


Fig. 5: Example of a domain-specific SLS modeled via the λ-chart [4] (left). Meta-model instantiation examples of the domain-specific SLS (right).

necessary to reach optimal parameter combinations. Hence, the problem of selecting effective search strategies is getting into the focus. Furthermore, an SLS realizes a comprehensive view on available design parameters. Hence, it becomes easier to improve design time and quality by detecting insignificant and interfering parameters. This helps system designers to focus on relevant design and tool parameters.

The integration aims at a coexistence or merger of the proposed SLS and existing specification languages. For example, SDL and SystemC contain module concepts that help to embed system specifications into an executable node of the proposed meta-model. In SDL, systems include a hierarchy of agents called processes and blocks. In SystemC, the Main is the starting point of a SystemC specification. A Main contains several modules and signals to model communications between modules. In the proposed SLS, a node encapsulates an execution of design tools solving design problems, such as scheduling or allocation tasks. These tools produce new or modified system specifications further used as tool input. Depending on the design tool, different system specifications, such as of applications, architectures, application mappings, validation results, tool configurations, etc. are produced and consumed. Hence, a node element can enclose multiple system specifications.

IV. ELECTRONIC DESIGN AUTOMATION ENVIRONMENT

In the following, an EDA environment is briefly introduced allowing to model and execute domain-specific SLSs based on the λ-chart and presented in Section III. In Figure 6, the tripartite structure consisting of front-, middle- and back-end is depicted. In the EDA environment, an SLS is graphically defined and automatically processed by running the executable node elements. The nodes communicate via parameters and XML-based input/output formats representing system specifications. The specifications address very early system-level design by using coarse-grained representations, such as fixed execution time and high-level task graphs. In the graphical front-end, designers are able to create SLS via CAD. Dynamic search and exploration strategies are

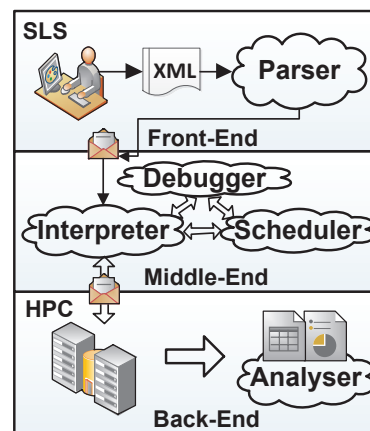


Fig. 6: EDA environment.

considered by including node elements, such as loop, branch, abort, etc. Referring to Figure 6, a parser is responsible for a syntactic analysis of an XML file representing the SLS. In the middle-end, the SLS is interpreted and an execution order is determined via as soon as possible scheduling. A correct execution of each node can be checked through the debugger. In the current implementation, each node executes a command line tool solving specific design problems. Hence, parameters, constraints, and system specifications are assigned via command line arguments. In the back-end, the nodes are executed in a distributed computing environment (High Performance Cluster, HPC). Depending on the purpose, different output formats are created during node execution. DSE results are automatically analyzed and validated as defined in the SLS. Further details of the EDA environment are out of the scope of the paper.

V. DESIGN FLOW CASE STUDY

The following case study demonstrates the usage of executable SLSs in order to realize a design flow for the heterogeneous multicluster architecture [17]. The SLSs are executable on the EDA environment presented in Section IV.

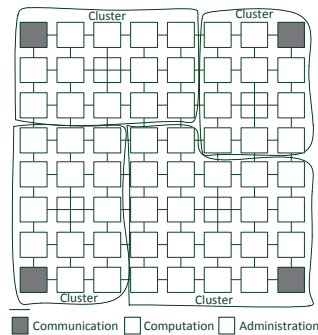


Fig. 7: The heterogeneous multicluster architecture model.

In the paper, the detailed explanations will be limited to one design aspect and the variation of a single design parameter.

A. Application and Architecture Model

The models consider functionalities of the communication, computation and administration view defined in the λ -chart [4]. The application model includes multiple, concurrently running applications and threads, respectively. A thread is represented by a high-level task graph and it sequentially executes tasks. Threads are only synchronized before or after execution. Then, a task is an atomic kernel exclusively executing on an intellectual property (IP) core, e.g., processor core, memory interface, controller interface, etc. Tasks produce and consume chunks of data accessed via shared memory. Side effects are excluded by preventing access to external data during computation.

As shown in Figure 7, the architecture model is a heterogeneous set of multiprocessor system-on-chips (MPSoCs) and clusters, respectively. The administrative unit (AU) represents an application processor and includes a load balancer aiming at equally distributing thread loads amongst the clusters. Moreover, an MPSoC contains heterogeneous types and numbers of IP cores. In the model, each MPSoC contains a NoC connecting the IP cores. Moreover, each cluster includes a control processor (CP) responsible to dynamically schedule arriving tasks to the available IP cores. The CPs are directly connected to the AU.

B. The Design Flow

The design flow, shown in Figure 8, lists several design aspects and executable SLSs, respectively, to create a heterogeneous multicluster architecture. According to Section III, the five domain-specific SLSs are modeled via the λ -chart. They are arranged by the explored parameter types, more specifically with the help of parameters of the design tool, structural, behavioral and physical design. Each SLS realizes a part of the design flow organized as follows:

- 1) The genetic algorithm (GA) sensitivity aims at finding the best tool parameters for two GAs each solving the multicluster dimensioning and IP core mapping problems;
- 2) The design aspect “Multicluster Dimensioning” creates a heterogeneous multicluster architecture by distributing estimated application mappings among

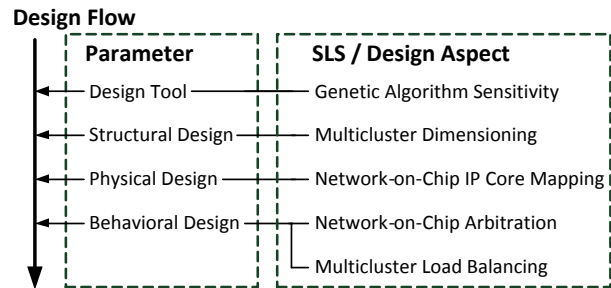


Fig. 8: Design flow case study for the heterogeneous multicluster architecture.

clusters and solving the optimization problem via a GA [18];

- 3) The design aspect “IP Core Mapping” places IP cores in an 1-ary n-mesh NoC constrained by the number of modules at each router. The optimization problem is solved via a GA [16];
- 4) Both design aspects aim at finding suitable behavioral schemes from a selection via simulation. “NoC arbitration” compares a locally fair with a globally fair arbitration scheme. In addition, flit-based and packet-based switching are considered [19]. “Multicluster load balancing” compares different estimators of cluster load, such as response time and queue size, used in the load balancing scheme of the AU.

C. Exemplary SLS and Test Results

In the following, multicluster dimensioning has been selected as exemplary SLS. Detailed explanation of the approach and benchmark results can be found in [18]. Referring to Figure 9, the example focuses on the design aspect of creating a suitable computation infrastructure for the heterogeneous multicluster architecture. Hence, DSE is limited to the computation view of the λ -chart. The same benchmark and simulation setup given in [18] has been chosen. The “modeling and partitioning” step serves as a starting point without any further purpose. In the “provisioning” step, the specifications of the target application and the optional IP cores are used as input for the parallelism analysis [20]. The estimated parallelism values represent the numbers and types of IP cores necessary to execute the applications. Since, multiple applications are concurrently running on the target architecture, the parallelism values are used to dimension the processing elements (PEs) in the clusters and MPSoCs, respectively. All clusters are constrained to a maximum number of PEs per cluster (maxPEs) used as a design parameter in the SLS. Further parameters are out of the scope of the paper. As mentioned before, the optimization problem is solved via a GA. The subsequent “scheduling” step performs application mapping via simulation. The scheduling results are analyzed in the “validation” step. Referring to Figure 9, a loop node is inserted to increment the design parameter from two to six. The selected parameter interval depends on the available numbers and types of PEs defined in the chosen benchmark setup. Figure 10 shows the validation results in terms of number of clusters / PEs and the thread response time. The response time is defined as time from the request

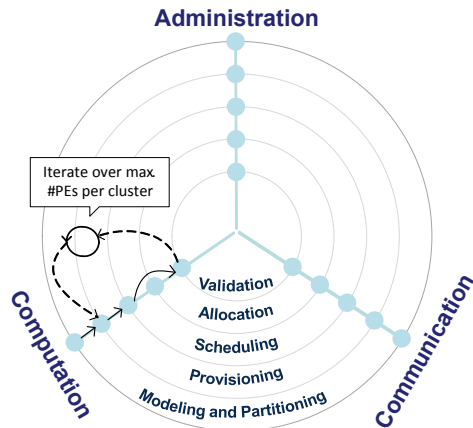


Fig. 9: Multicluster dimensioning SLS and design aspect, respectively.

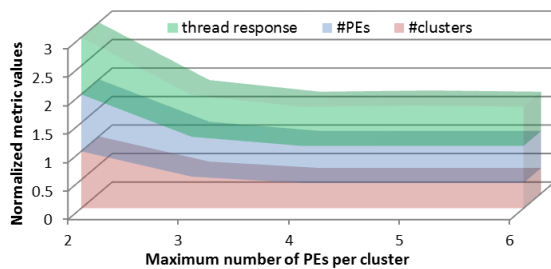


Fig. 10: Test results via parameter exploration for the “Multicluster Dimensioning” SLS.

of a thread until its end. All values have been normalized. In the setup, the best tradeoff is reached for $maxPEs = 4$ resulting in four clusters and 13 PEs. The slightly varying cluster configurations for $maxPEs = \{4, 5, 6\}$, not shown in Figure 10, are due to the heuristic nature of a GA.

VI. CONCLUSIONS AND OPEN TOPICS

In the paper, an executable SLS is proposed in order to cope with an increasing number of design parameters and system specifications during the design cycle. An SLS represents an executable DSE methodology and encapsulates system specifications. The aim is to formalize and automate design flows in order to scale to larger and more complex embedded systems. SLSs should not be limited to certain embedded system types. Hence, SLSs need to be standardized across tools, designers, and domains. Therefore, a meta-methodology, as well as a meta-model are developed defining a domain-independent SLS. Hence, an EDA environment is presented allowing to graphically create and automatically execute embedded domain-specific SLSs. Finally, a case study shows the feasibility of the proposed SLS. Therein, several SLSs demonstrate a realization of a design flow for the heterogeneous multicluster architecture.

In the rest of the paper, a discussion of open topics outlines the future work. So far, an executable SLS can be graphically defined in the introduced EDA environment. Nevertheless, many designers prefer textual programming languages, but according tool support, such as a parser, interpreter, and debugger, is a challenging task. This is left out for future work. Another open topic is to embed industry-relevant

specification languages, such as SDL and SystemC, into the executable SLS. Furthermore, an integrated development environment (IDE) should be provided implementing the proposed meta-modeling and integrating the tool chain. The purpose is to create and execute domain-specific SLSs with IDE support. Additional research is necessary to introduce executable SLSs in other domains, such as in communications and sensor networks.

REFERENCES

- [1] D. D. Gajski, J. Peng, A. Gerstlauer, H. Yu, and D. Shin, “System design methodology and tools.” *CECS, UC Irvine, Technical Report CECS-TR-03-02*, January 2003.
- [2] Systemc, osci. [Online]. Available: <http://www.systemc.org/>
- [3] S. Borkar, “Thousand core chips: a technology perspective,” pp. 746–749, 2007.
- [4] F. Guderian and G. Fettweis, “The lambda chart: A model of design abstraction and exploration at system-level,” in *Proc. of SIMUL*, 2011, pp. 7–12.
- [5] ITU-T, *Recommendation Z.100 (08/02) Specification and Description Language (SDL)*, International Telecommunication Union (2002).
- [6] S. Traboulsi, F. Bruns, A. Showk, D. Szczesny, S. Hessel, E. Gonzalez, and A. Bilgic, “Sdl/virtual prototype co-design for rapid architectural exploration of a mobile phone platform,” in *Proc. of SDL*, 2009, pp. 239–255.
- [7] D. D. Gajski, R. Zhu, J. Dmer, A. Gerstlauer, and S. Zhao, *SpecC Specification Language and Methodology*. Kluwer Academic Publishers, 2000.
- [8] C. Haubelt, T. Schlichter, J. Keinert, and M. Meredith, “System-codesigner: automatic design space exploration and rapid prototyping from behavioral models,” in *Proceedings of the 45th annual Design Automation Conference*, ser. Proc. of DAC, 2008, pp. 580–585.
- [9] R. Dömer, A. Gerstlauer, J. Peng, D. Shin, L. Cai, H. Yu, S. Abdi, and D. D. Gajski, “System-on-chip environment: a specc-based framework for heterogeneous mpsoe design,” *EURASIP J. Embedded Syst.*, vol. 2008, pp. 5:1–5:13, Jan. 2008.
- [10] J. Eker, J. Janneck, E. Lee, J. Liu, X. Liu, J. Ludvig, S. Neuendorfer, S. Sachs, and Y. Xiong, “Taming heterogeneity - the ptolemy approach,” *Proceedings of the IEEE*, vol. 91, no. 1, pp. 127–144, jan 2003.
- [11] C. Silvano *et al.*, “Multicube: Multi-objective design space exploration of multi-core architectures,” in *Proc. of ISVLSI*, July 2010, pp. 488–493.
- [12] Z. J. Jia, A. Pimentel, M. Thompson, T. Bautista, and A. Nunez, “Nasa: A generic infrastructure for system-level mp-soc design space exploration,” in *Proc. of ESTIMedia*, Oct 2010, pp. 41–50.
- [13] L. Bonde, C. Dumoulin, and J.-L. Dekeyser, “Metamodels and mda transformations for embedded systems,” in *FDL*, 2004, pp. 240–252.
- [14] D. Mathaikutty, H. Patel, S. Shukla, and A. Jantsch, “Ewd: A meta-modeling driven customizable multi-moc system modeling framework,” *ACM Trans. Des. Autom. Electron. Syst.*, vol. 12, no. 3, pp. 33:1–33:43, May 2008.
- [15] D. Mathaikutty and S. Shukla, “Mcf: A metamodeling-based component composition framework—composing systemc ips for executable system models,” *IEEE Transactions on VLSI Systems*, vol. 16, no. 7, pp. 792–805, July 2008.
- [16] F. Guderian, R. Schaffer, and G. Fettweis, “Administration- and communication-aware ip core mapping in scalable multiprocessor system-on-chips via evolutionary computing,” in *Proc. of CEC*, 2012, accepted for publication.
- [17] K. I. Farkas, P. Chow, N. P. Jouppi, and Z. Vranesic, “The multicluster architecture: reducing cycle time through partitioning,” in *Proc. of Micro*, 1997, pp. 149–159.
- [18] F. Guderian and R. Schaffer and G. Fettweis, “Dimensioning the heterogeneous multicluster architecture via parallelism analysis and evolutionary computing,” in *Proc. of CEC*, 2012, accepted for publication.
- [19] F. Guderian, E. Fischer, M. Winter, and G. Fettweis, “Fair rate packet arbitration in network-on-chip,” in *Proc. of SOCC*, sept. 2011, pp. 278–283.
- [20] B. Ristau, T. Limberg, O. Arnold, and G. Fettweis, “Dimensioning heterogeneous MPSoCs via parallelism analysis,” in *Proc. of DATE*, 2009, pp. 554–557.

Unsupervised Image Segmentation Circuit Based on Fuzzy C-Means Clustering

Wen-Jyi Hwang, Zhe-Cheng Fan, Tsung-Mao Shen
 Department of Computer Science and Information Engineering,
 National Taiwan Normal University
 Taipei, Taiwan

e-mails: whwang@ntnu.edu.tw; 699470137@csie.ntnu.edu.tw; 698470594@csie.ntnu.edu.tw

Abstract—This paper presents a novel VLSI architecture for unsupervised image segmentation. The circuit is a hardware implementation of fuzzy *c*-means algorithm for the unsupervised clustering. The number of segments is determined by Xie-Beni index. An efficient pipeline circuit is proposed for the computation of the index. The circuit is used as a hardware accelerator of a softcore processor in a system-on-programmable chip for physical performance measurement. Experimental results reveal that the proposed architecture is an effective alternative for realtime segmentation with low error rate and area costs.

Keywords-FPGA; Image Segmentation; Unsupervised Clustering; System-on-Chip.

I. INTRODUCTION

The goal of image segmentation is to cluster image pixels into multiple segments. The segmentation results can be used to identify regions of interest and objects in the scene for the subsequent image analysis or annotation. The fuzzy *c*-means algorithm (FCM) [1] is one of the most used techniques for image segmentation [2][3]. The effectiveness of FCM is due to the employment of fuzziness for the clustering of each image pixel.

Nevertheless, there are some drawbacks to employ the FCM algorithm. The first is its high computational complexity for membership coefficients computation and centroid updating. In addition, the size of membership matrix grows as the product of data set size and number of segments. As a result, the corresponding memory requirement may prevent the algorithm from being applied to large images. Finally, the number of segments should be pre-specified. Therefore, it is difficult to use FCM for the fully unsupervised realtime image segmentation.

A number of algorithms [4][5] have been proposed for accelerating the computational speed and/or reducing memory requirement of FCM. Most of these algorithms are implemented by software, and only moderate acceleration can be achieved. In [6][7], hardware implementations of FCM are proposed. However, the design in [6] is based on analog circuits. The clustering results therefore are difficult to be directly used for digital applications. Although the architecture shown in [7] adopts digital circuits, the architecture aims for applications with only two classes. The architecture may then not be useful for applications demanding the clustering of larger number of classes.

With the above observation, our earlier work [8] introduced a digital FCM architecture which can process more than two classes. Although the architecture is effective, its area cost is very high. The large hardware resource consumption arises from the employment of broadcasting scheme for membership coefficients and centroid computation at centroid level. As a result, the area cost grows with the number of segments. The FCM architecture may then only be used for clustering applications with small number of segments. Moreover, the architecture does not provide the function of determining the number of segments. The FCM architecture presented in [9] is able to reduce the area cost. However, the number of classes still needs to be pre-specified. These architectures are therefore not suited for the implementation of fully unsupervised realtime image segmentation.

The goal of this paper is to present a novel FCM architecture for fully unsupervised realtime image segmentation. In order to eliminate the large storage size for membership matrix, our implementation combines the usual iterative updating processes of membership matrix and cluster centroid into a single updating process. In our approach, the updating process is divided into three steps: pre-computation, membership coefficients updating, and centroid updating. The pre-computing step is used to compute and store information common to the updating of different membership coefficients. This step is beneficial for reducing the computational complexity for the updating of membership coefficients.

The membership updating step computes new membership coefficients based on a fixed set of centroids and the results of the pre-computation step. The weighted sum of data points and the sum of membership coefficients are also updated incrementally here for the subsequent centroid computation. This incremental updating scheme eliminates the requirement for storing the entire membership coefficients.

Following the updating process of membership matrix and cluster centroid, a cluster validation process is performed to find the optimal number of segments. The Xie-Beni index [10] is employed for this purpose because of its simplicity and effectiveness. Partial results of the updating process can be used for the computation of this index. In addition, an efficient pipeline architecture is proposed to further enhance the throughput of the computation.

For each class number, the updating process of membership matrix and cluster centroid, and the cluster validation process are performed sequentially. The resulting Xie-Beni index is stored, and compared with that associated with other class numbers. The class number with minimum index value is then selected as the class number for the image segmentation.

The proposed architecture has been implemented on field programmable gate array (FPGA) devices [11] so that it can operate in conjunction with a softcore CPU. Using the reconfigurable hardware, we are capable of constructing a system on programmable chip (SOPC) system for the physical performance measurement. Experimental results show that the proposed architecture has the advantages of high speed computation, low area cost and low error rate for image segmentation. In addition, because of its effectiveness, the proposed architecture can also be directly used for other clustering applications where the number of clusters is desired to be determined in an unsupervised manner such as spike sorting [12].

The remaining parts of this paper are organized as follows: Section 2 gives a brief review of the FCM algorithm. Section 3 describes the proposed FCM architecture. Experimental results are included in Section 4. Finally, the concluding remarks are given in Section 5.

II. PRELIMINARIES

This section gives a brief review of the FCM algorithm. Let $\mathbf{X} = \{x_1, \dots, x_t\}$ be a data set to be clustered by the FCM algorithm into c classes, where t is the number of data points in the design set. Each class i , $1 \leq i \leq c$, is identified by its centroid v_i . For the image segmentation applications, \mathbf{X} is an image to be segmented, x_k is a block in \mathbf{X} , t is the number of blocks in \mathbf{X} , and c is the class number. The goal of FCM is to minimize the following cost function:

$$J = \sum_{i=1}^c \sum_{k=1}^t u_{i,k}^m \|x_k - v_i\|^2, \quad (1)$$

where $u_{i,k}^m$ is the membership of x_k in class i , and $m > 1$ indicates the degree of fuzziness. The cost function J is minimized by a two-step iteration in the FCM. In the first step, the centroids v_1, \dots, v_c , are fixed, and the optimal membership matrix is computed by

$$u_{i,k} = \left(\sum_{j=1}^c (\|x_k - v_i\| / \|x_k - v_j\|)^{2/(m-1)} \right)^{-1}. \quad (2)$$

After the first step, the membership matrix is then fixed, and the new centroid of each class i is obtained by

$$v_i = \left(\sum_{k=1}^t u_{i,k}^m x_k \right) / \left(\sum_{k=1}^t u_{i,k}^m \right). \quad (3)$$

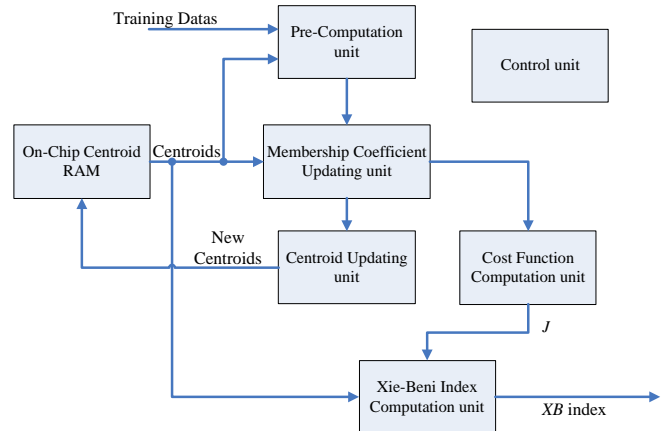


Figure 1. The proposed FCM architecture.

The FCM algorithm requires large number of floating point operations. Moreover, from (1) and (3), it follows that the membership matrix needs to be stored for the computation of cost function and centroids. As the size of the membership matrix grows with the product of t and c , the storage size required for the FCM may be impractically large when the data set size and/or the number of classes become high.

In the FCM, the number of classes c needs to be pre-specified. For the fully unsupervised image segmentation, the class number also needs to be determined. One way to find the optimal class number is to evaluate the clustering results for each c based on a cluster validation index. The class number producing the optimal index value is selected as the actual class number for image segmentation. A commonly used cluster validation index is the Xie-Beni index [10], which is defined as

$$XB(c) = \frac{J}{t \left(\min_{i \neq j} \|v_i - v_j\|^2 \right)}, \quad (4)$$

where J is the cost function of FCM defined in (1).

III. THE PROPOSED ARCHITECTURE

As shown in Figure 1, the proposed FCM architecture can be decomposed into six units: the pre-computation unit, the membership coefficients updating unit, the centroid updating unit, the cost function computation unit, the on-chip centroid RAM, and the control unit.

A. Precomputation Unit

The pre-computation unit is used for reducing the computational complexity of the membership coefficients calculation. Observe that (2) can be rewritten as

$$u_{i,j} = \left(\|x_k - v_i\| \right)^{-2/(m-1)} P_k^{-1} \quad (5)$$

where

$$P_k = \sum_{j=1}^c (1/\|x_k - v_j\|^2)^{1/(m-1)}. \quad (6)$$

Given x_k and centroids v_1, \dots, v_c , membership coefficients $u_{1,k}, \dots, u_{c,k}$ have the same P_k . Therefore, the complexity for computing membership coefficients can be reduced by calculating P_k in the pre-computation unit. For the sake of simplicity, we set $m = 2$ for our design.

Figure 2 shows the architecture of the pre-computation unit, where the x_k is obtained from the on-chip memory of the SOPC system, and v_i is obtained from the on-chip centroid RAM of the FCM architecture. As depicted in Figure 2, the circuit in its simplest form can be divided into two stages, which involve the squared distance computation, and inverse computation, respectively. The circuit can easily be separated into multistage pipeline for enhancing the throughput.

B. Membership Coefficient Updating Unit

Figure 3 depicts the architecture of the membership coefficients updating unit based on (5). It can be observed from Figure 3 that, given a training data x_k , the membership coefficients updating unit computes $u_{i,k}^2$ for $i = 1, \dots, c$, one at a time. Similar to the pre-computation unit, the x_k remains as the input until all the centroids $v_i, i = 1, \dots, c$, have been fetched from the on-chip centroid RAM for the computation of $u_{i,k}^2$. Based on (5) with $m = 2$, it follows that the circuit contains 3 multipliers and 1 divider. Similar to the precomputation unit architecture, the circuit can be separated into multistage pipeline for efficient computation.

C. Centroid Computation Unit

The centroid updating unit incrementally computes the centroid of each cluster. The major advantage for the

incremental computation is that it is not necessary to store the entire membership coefficients matrix for the centroid computation. The centroid updating unit computes the incremental centroid when x_k and $u_{i,k}^2$ are received, and clusters will only be updated when the final centroid is generated after completing the computation of last training vector. Thus, no membership coefficients matrix is needed. Define the incremental centroid for the i -th cluster up to data point x_k as

$$v_i(k) = \left(\sum_{n=1}^k u_{i,n}^m x_n \right) / \left(\sum_{n=1}^k u_{i,n}^m \right). \quad (7)$$

When $k = t$, $v_i(k)$ is then identical to the actual centroid v_i given in (3).

Figure 4 shows the architecture of the centroid update unit, which contains a multiplier, an intermediate on-chip RAM and a divider. The unit has three inputs: centroid index i , training vector x_k and membership coefficient $u_{i,k}^2$.

As shown in Figure 4, both $u_{i,k}^2 x_k$ and $u_{i,k}^2$ are used as the inputs to the intermediate on-chip RAM for computing $v_i(k)$.

D. Cost Function Computation Unit

Similar to the centroid updating unit, the cost function unit incrementally computes the cost function J . Define the incremental cost function $J(k)$ up to data point x_k as

$$J(i, k) = \sum_{z=1}^k \sum_{j=1}^i u_{j,z}^2 \|x_z - v_j\|^2. \quad (8)$$

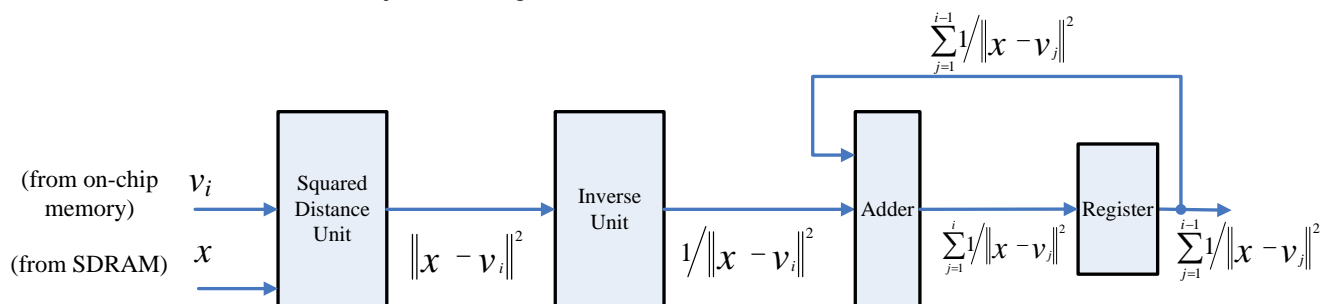


Figure 2. The architecture of precomputation unit.

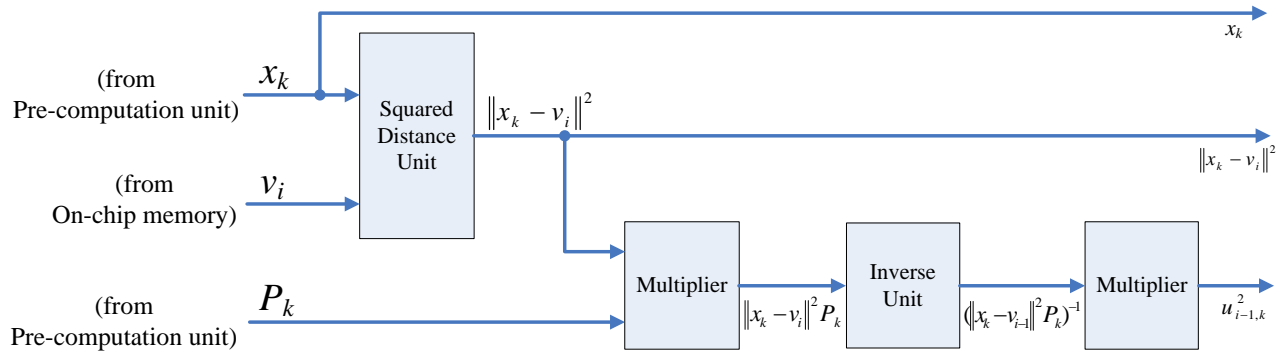


Figure 3. The architecture of membership coefficient updating unit

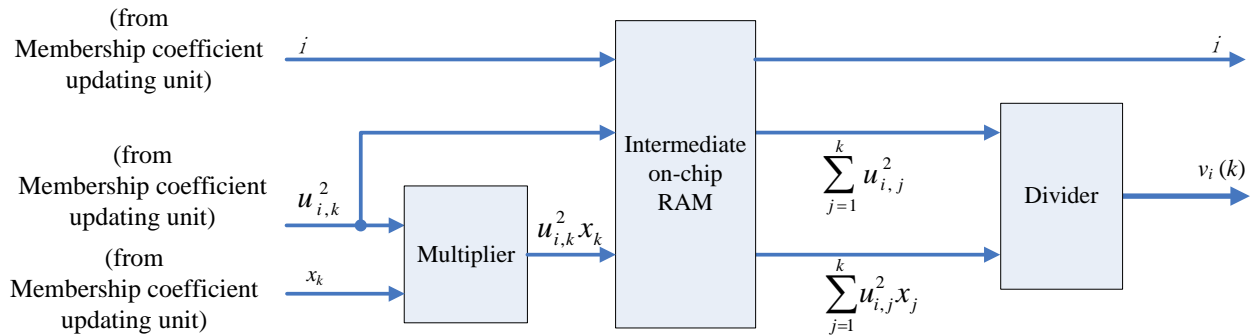


Figure 4. The architecture of centroid computation unit

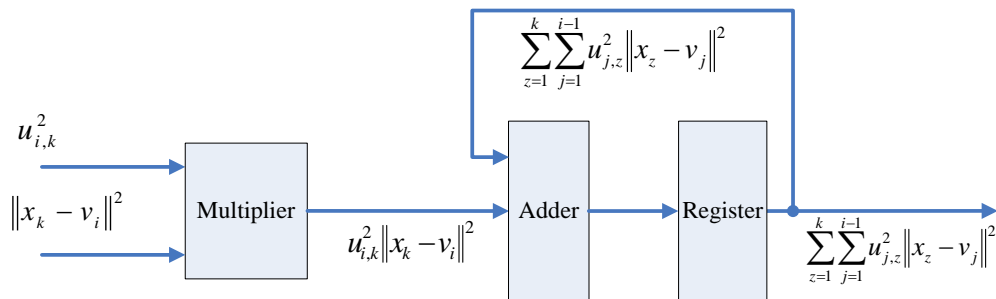


Figure 5. The architecture of cost function computation unit

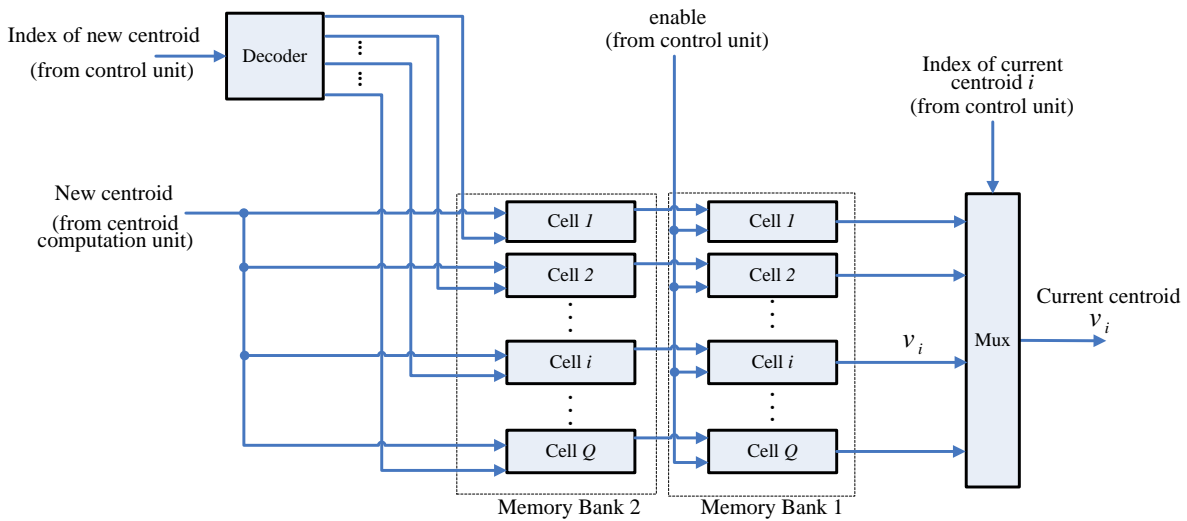


Figure 6. The architecture of on-chip centroid RAM

As shown in Figure 5, the cost function computation circuit receives $u_{i,k}^2$ and $\|x_k - v_i\|^2$ from the membership coefficients updating unit. The product $u_{i,k}^2 \|x_k - v_i\|^2$ is then accumulated for computing $J(i, k)$ in eq. (7).

When $i = c$ and $k = t$, $J(i, k)$ then is identical to the actual cost function J given (1). Therefore, the output of the circuit becomes J as the cost function computations for all the training vectors are completed.

E. On-Chip Centroid RAM

This unit is used for storing the centroids for FCM clustering. As shown in Figure 6, there are two memory banks (Memory Bank 1 and Memory Bank 2) in the on-chip centroid RAM. The Memory Bank 1 stores the current centroids v_1, \dots, v_c . The Memory Bank 2 contains the new v_1, \dots, v_c obtained from the centroid updating unit. Only the centroids stored in the Memory Bank 1 are delivered to the pre-computation unit and membership updating unit for the membership coefficients computation. The updated centroids obtained from the centroid updating unit are stored in the Memory Bank 2. Note that, the centroids in the Memory Bank 2 will not replace the centroids in the Memory Bank 1 until all the input training data points $x_k, k = 1, \dots, t$, are processed.

It can also be observed from Figure 6 that there are Q cells in each memory bank, where Q is the upper limit of the number of centroids c . Therefore, the proposed FCM circuit is able to conduct image segmentation with number of classes c less than or equal to Q .

F. Xie-Beni Index Computation unit

The goal of Xie-Beni Index computation unit is to compute $XB(c)$ given in (4). The numerator of $XB(c)$ is actually the cost function. Hence, we can directly use the output of the cost function unit as the numerator of $XB(c)$.

The denominator contains $\min_{i,k} \|v_i - v_k\|$. The corresponding circuit should be implemented in the cluster validity index computation unit. Although the direct implementation of $\min_{i,k} \|v_i - v_k\|$ is possible, the time and area complexity would be $O(c^2)$. Therefore, the complexities would be very high when c becomes large. The proposed circuit is able to reduce the overhead. Figure 7 shows the architecture of Xie-Beni index computation unit, which contains the minimum computation unit, a multiplier, and a divider. The minimum computation unit contains an efficient pipeline for the computation of $\min_{i,k} \|v_i - v_k\|$, as depicted in Figure 8. The circuit can be viewed as a c -stage pipeline, where each stage contains one processing module (PM). The centroids are delivered to the pipeline from on-chip centroid memory one at a time. Each centroid will traverse through the pipeline. As shown in the figure, the latest input entering the pipeline will be broadcasted to all the PMs. Let

$$D_{min}(v_p) = \min_{k, k \neq p} \|v_p - v_k\|^2. \quad (9)$$

Suppose now the centroid v_p arrives at PM i , and the centroid v_q is the newest centroid entering the pipeline. In the PM i , the distance between v_p and v_q will be computed, and will be compared with the current $D_{min}(v_p)$. If $\|v_p - v_q\|^2 < D_{min}(v_p)$, then $\|v_p - v_q\|^2$ will be the new current $D_{min}(v_p)$. As v_p reaches stage c of the pipeline, the current $D_{min}(v_p)$ becomes the actual $D_{min}(v_p)$. When all the centroids have reached the stage c , the actual $\min_{i,k} \|v_i - v_k\|$ can be computed by

$$\min_{i \neq j} \|v_i - v_j\|^2 = \min_p D_{min}(v_p). \quad (10)$$

The time and area complexities of the proposed pipeline are only $O(c)$. The proposed architecture is therefore effective for Xie-Beni index computation. Finally, we note that, because it is necessary to compute the $XB(c)$ for various c values, the pipeline actually will be implemented in Q stages, where Q is the upper bound of the c value.

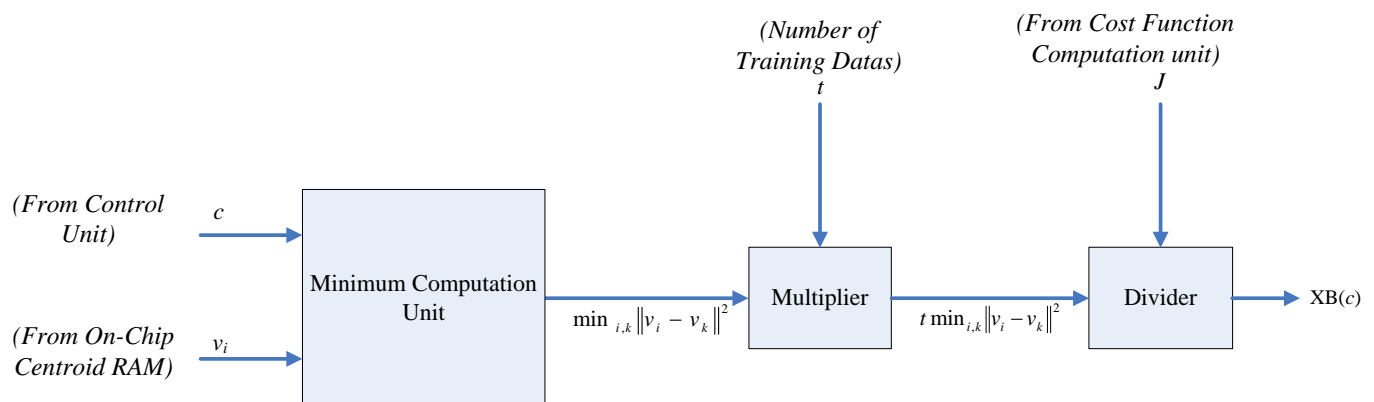


Figure 7. The architecture of Xie-Beni index computation unit.

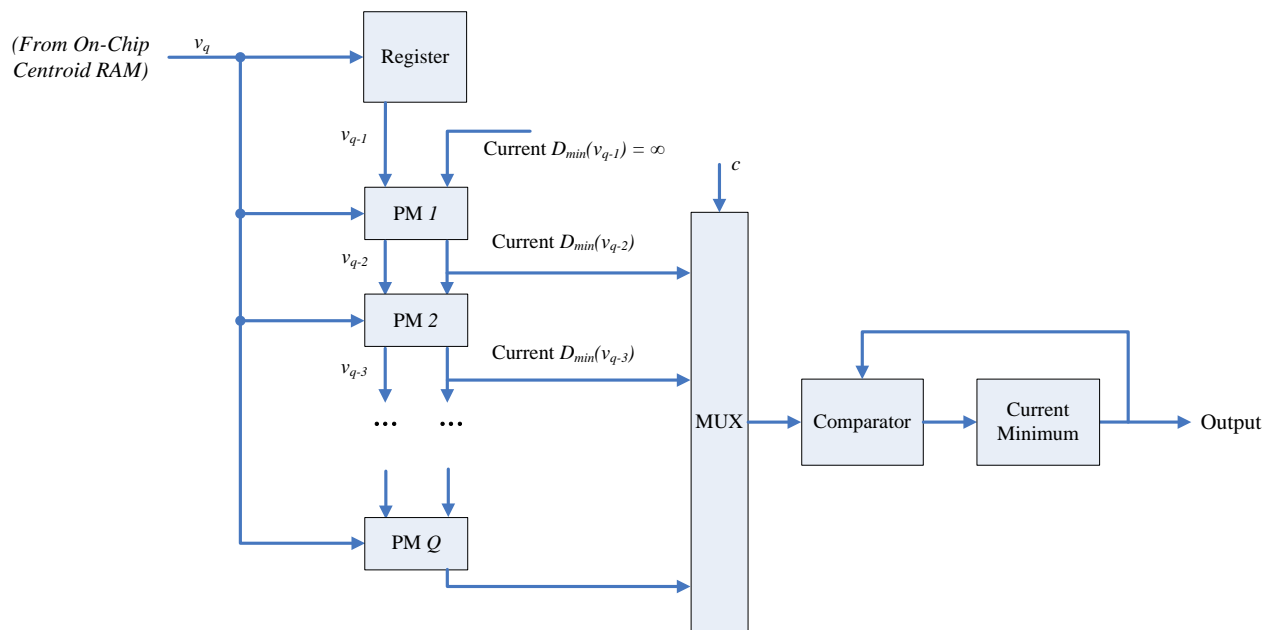
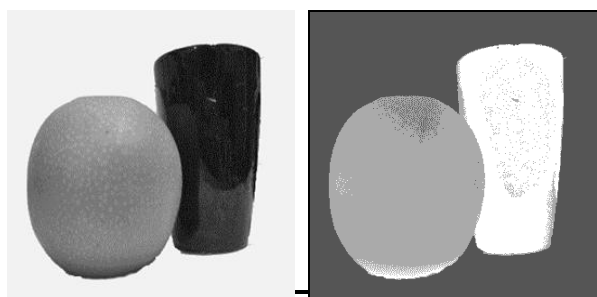


Figure 8. Architecture of minimum computation unit

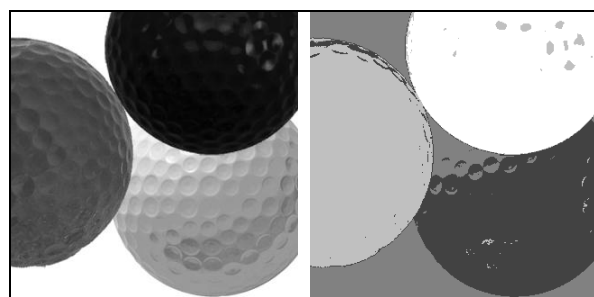


(a) "Strawberry"

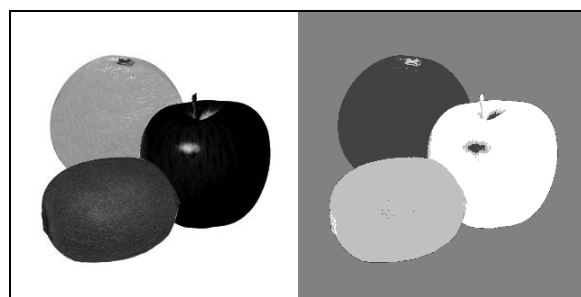


(b) "Pear & Cup"

Figure 9. The original images and their segmentation results produced by the proposed FCM architecture: (a) "Strawberry," (b) "Pear & Cup."



(a) "Golf Balls"



(b) "Fruits"

Figure 10. The original images and their segmentation results produced by the proposed FCM architecture: (a) "Golf Balls," (b) "Fruits."

IV. EXPERIMENTAL RESULTS

This section presents some physical performance measurements of the proposed FPGA implementation. The design platform of our system is Altera Quartus II 8.0 with SOPC Builder and NIOS II IDE.

Figures 9 and 10 shows the segmentation results of the proposed FCM architecture with $Q=10$. Therefore, the circuit is able to conduct fully unsupervised segmentation for images with number of classes c less or equal to 10.

Table I. The estimated and actual number of classes, and the segmentation success rate of the proposed FCM architecture for the images shown in Figures 9 and 10

Images	Strawberry	Peer & Cup	Gulf Balls	Fruits
Est. Class Number \hat{c}	2	3	4	4
Actual Class Number c	2	3	4	4
Segmentation Success Rate	98.97%	97.13%	94.77%	98.87%

Table II. The CPU time of various FCM implementations

Q	Proposed Architecture	Basic Software FCM	Fast Software FCM [5]
2	15.47 ms	256.20 ms	62.40 ms
3	30.97 ms	709.45 ms	151.65 ms
4	50.05 ms	1404.70 ms	272.65 ms
5	69.11 ms	2404.70 ms	428.85 ms
6	88.20 ms	3720.30 ms	613.05 ms
7	107.28 ms	5410.90 ms	825.45 ms
8	126.36 ms	7535.90 ms	1067.78 ms
9	145.46 ms	10149.15 ms	1345.03 ms
10	164.55 ms	13389.75 ms	1648.23 ms

All the images have the same dimension 320×320 . The images are separated into 2×2 blocks for FCM training and segmentation. Table 1 shows the estimated and actual number of classes, and the segmentation success rate of these images. The segmentation success rate of an image is defined as the number of pixels which are misclassified divided by the total number of pixels of the image. From Figures 9 and 10, and Table 1, it can be observed that the proposed architecture is able to correctly identify the number of classes with high classification success rate.

The speed of various FCM implementations is revealed in Table 2. The target FPGA device is Altera Stratix III EP3SL150F1152C2N [13]. The speed of the proposed architecture is the CPU time of the softcore NIOS processor [14] using the proposed architecture as the hardware accelerator. The clock rate of the NIOS processor is 75 MHz. The software implementations are running on 2.8 GHz Intel Pentium D processor. Two software implementations are considered: the basic FCM implementation, and the fast FCM implementation [5].

Figure 10 shows the speedup of the proposed architecture over the fast FCM [5]. It can be observed from Table 2 and Figure 11 that the proposed architecture has significantly lower computation time as compared with its software counterparts. Although the NIOS processor is running at a lower clock rate as compared with Intel CPU (i.e., 75 MHz versus 2.8 GHz), it still has higher computational speed because of the efficiency of the proposed architecture for the membership matrix and centroid computation.

The hardware utilization of the proposed architecture for various Q values is shown in Table 3 for Altera Stratix III EP3SL150F1152C2N. It can be observed from the table that the consumption of ALMs and DSP block grow linearly with Q . Nevertheless, only a small fraction of hardware resources are consumed. In particular, when $Q=10$, only 20 %, 27 % and 19% of the ALM, block memory bits, and DSP blocks are consumed by the proposed architecture.

Finally, Table 4 compares the hardware utilization of the proposed architecture with that of the architecture in [8] with block size 2×2 . The target device is Altera Cyclone III EP3C120. The logic elements (LEs) are the hardware resources considered in the table.

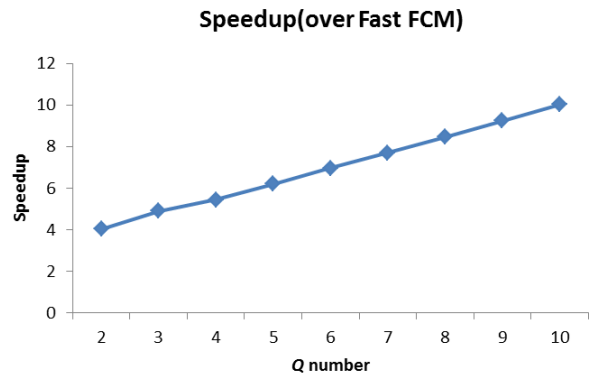


Figure 11. The speedup of the proposed architecture over the fast FCM in [5].

From Table 4, we can see that the proposed architecture has significantly lower utilization of LEs as compared with the architecture in [8]. In fact, the proposed architecture is able to operate up to $Q=64$ with the consumption of only 40% of LEs of the target FPGA. By contrast, the architecture in [8] consumes almost all the LEs when Q reaches 32. All these facts demonstrate the effectiveness of the proposed architecture.

Table III. The hardware utilization of the proposed architecture.

Q	ALMs	Block Memory Bits	DSP Block Elements
2	10738/56800 (18%)	1535264/5630976 (27%)	40/384 (10%)
3	10814/56800 (19%)	1535840/5630976 (27%)	44/384 (11%)
4	10893/56800 (19%)	1535904/5630976 (27%)	48/384 (13%)
5	11056/56800 (19%)	1536992/5630976 (27%)	52/384 (14%)
6	11199/56800 (19%)	1537056/5630976 (27%)	56/384 (15%)
7	11308/56800 (19%)	1537120/5630976 (27%)	60/384 (16%)
8	11405/56800 (20%)	1537184/5630976 (27%)	64/384 (17%)
9	11612/56800 (20%)	1539296/5630976 (27%)	68/384 (18%)
10	11793/56800 (20%)	1539360/5630976 (27%)	72/384 (19%)

Table IV. The LE utilization of various architectures.

Q	Proposed Architecture	Architecture in [8]
4	16553/119088 (14%)	21084/119088 (18%)
8	18504/119088 (16%)	35423/119088 (30%)
16	22568/119088 (19%)	59868/119088 (50%)
32	30827/119088 (26%)	114117/119088 (97%)
64	47412/119088 (40%)	N/A

V. CONCLUDING REMARKS

Experimental results revealed that the proposed architecture is able to correctly estimate the number of classes of an image with segmentation success rate above 94%. For the cases where the upper bound of the number of classes is 10, the proposed architecture consumes less than 30% of the ALMs, block memory bits, and DSP blocks of the Stratix III FPGA device. It also attains speedup of 10 over its software counterpart running on the Intel general purpose CPU. The proposed architecture, therefore, is effective for unsupervised image segmentation with low area costs and high computation speed.

REFERENCES

- [1] J.C. Bezdek, *Fuzzy Mathematics in Pattern Classification*, Cornell University: Ithaca, NY, USA, 1973.
- [2] S.C. Chen and D.Q. Zhang, "Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure," *IEEE Trans. Syst. Man Cybern. B*, 2004, pp. 1907-1916.
- [3] K.S. Chuang, H.L. Tzeng, S. Chen, J. Wu, and T.J. Chen, "Fuzzy c-means clustering with spatial information for image segmentation," *Comput. Med. Imaging Graphics*, 2006, pp. 9-15.
- [4] S. Eschrich, J. Ke, L.O. Hall, and D.B. Goldgof, "Fast Accurate Fuzzy Clustering Through Data Reduction," *IEEE Transaction on. Fuzzy Systems*, 2003, pp. 262-270.
- [5] J. F. Kolen and T. Hutcheson, "Reducing the Time Complexity of the Fuzzy C-Means Algorithm," *IEEE Trans. Fuzzy Systems*, pp. 263-267, Vol. 10, 2002.
- [6] J. Garcia-Lamont, L.M. Flores-Nava, F. Gomez-Castaneda, and J.A. Moreno-Cadenas, "CMOS Analog Circuit for Fuzzy C-Means Clustering," *IEEE Proceedings 5th Biannual World Automation Congress*, 2002.
- [7] J. Lazaro, J. Arias, J.L. Martin, C. Cuadrado, and A. Astarloa, "Implementation of a Modified Fuzzy C-Means Clustering Algorithm for Realtime Applications," *Microprocessors and Microsystems*, 2005, pp. 375-380.
- [8] H.Y. Li, C.T. Yang, and W.J. Hwang, "Efficient VLSI Architecture for Fuzzy C-Means Clustering in Reconfigurable Hardware," *Proc. IEEE International Conference on Frontier of Computer Science and Technology*, 2009, pp. 168-174.
- [9] H.Y. Li, W.J. Hwang, and C.Y. Chang, "Efficient Fuzzy C-Means Architecture for Image Segmentation", *Sensors*, 2011, pp.6697-6718.
- [10] X.L. Xie and G. Beni. "A Validity measure for Fuzzy Clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1991.
- [11] S. Hauck and A. DeHon, *Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation*, Morgan Kaufmann, USA, 2008.
- [12] M.S. Lewicki, "A review of methods for spike sorting: the detection and classification of neural action potentials," *Network Computer Neural System*, 1998, pp. R53-R78.
- [13] Altera Corporation, *Stratix III Device Handbook*, 2011, <http://www.altera.com/literature/lit-stx3.jsp> (accessed on 6 August, 2012).
- [14] Altera Corporation, *NIOS II Processor Reference Handbook*, 2011, <http://www.altera.com/literature/lit-nio2.jsp> (accessed on 6 August, 2012).

Various Discussions and Improvements of Voltage Equalizer for EDLCs Including Secondary Batteries

Keiju Matsui, Kouhei Yamakita, Masaru Hasegawa
Chubu University
Kasugai, Japan
keiju@isc.chubu.ac.jp, mhasega@isc.chubu.ac.jp

Abstract-Among various storage devices, EDLCs offer high energy density and long life span, so a lot of applications may be anticipated in the realm of energy storage devices, such as those used in electric vehicles or electric power stabilization in power systems, etc. However, since the voltage limit of the devices is low, it is necessary to connect them in series in or parallel. In addition, it is required that they be used in the region of their critical voltage limit or capacity limit. In order to apply them efficiently, the devices should be used with balanced voltage. In this paper, a novel voltage equalizer and modified versions are presented, employing a CW (Cockcroft-Walton) circuit. Characteristics of the proposed circuit are analyzed and improved, especially about charging response and charging capability.

Keywords-EDLC; Voltage Equalizer; Voltage Balancer; EDLC; Cockcroft-Walton circuit; Buck-boost chopper.

I. INTRODUCTION

Various energy storage devices, including general secondary batteries, have been reported and examined. Among them, EDLCs (Electric Double Layer Capacitors), which are usually called super capacitors, can offer high energy storage performance in terms of surge power, efficiency, cold temperature operation and large number of energy cycles [1].

For these reasons, in power compensating equipments for voltage fluctuations or instantaneous voltage drop in the power systems, EDLCs are expected to be applied as energy storage equipments. Additionally, in various vehicles, such as electric cars and trains, these applications have just been introduced. In such EDLCs, however, as voltage limit of devices is low, it is necessary to connect them in series or parallel configurations, and to use them in the vicinity of their voltage limit. Consequently, in order to be used efficiently, these devices must be used in a well-balanced manner. Amongst various voltage equalizing techniques, an equalizing method using resistors can be applied as the most fundamental, simple and effective solution [2]. When considering the power losses, however, such methods have restricted application in practice. Another method, using Zener diodes, has been discussed and evaluated [1, 3]. In considering the energy consumption of such Zener diodes, the power capacity of the system may be limited. Methods employing chopper circuits have also

been proposed [3, 4], but the number of switching devices and their accompanied control circuitry is increased, leading to the high cost of such systems. Other original strategies have been presented and discussed, which use inverter circuits and transformers. By utilizing the charge and discharge of EDLCs, their voltages can be held effectively in equilibrium between device components [5]. Though the inverter circuit is complicated and charging operations are needed, such methods are suited to the required increased capacities. The most orthodox method is thought to be the forward converter method, using transformers, which accompany each EDLC, and charge and discharge through their primary and secondary windings [6-10]. In a similar to solution [3], their controls may be complicated by many devices like transformers. Although such devices are necessary, however, their size is very small. Thus, this technique is expected to be widely used in extensive applications like the electric vehicles [14].

Considering the various types of EDLC voltage equalizers, perfect or even adequate solutions have not been obtained. Although, in the future, various other methods will be studied and proposed. In the light of the above research into voltage equalizers for EDLCs, we had initially studied reference [6-9], and derived novel methods which was examined and discussed in [10]. An alternative approach to voltage balancing was presented, employing a Cockcroft-Walton circuit (CW circuit), which was invented long ago [13] for high voltage generation and employs numbers of capacitors and diodes. By means of this CW circuit, EDLCs having different capacitances were made to provide identical voltage. Voltage equalizing is achieved with an ac power supply or buck-boost chopper. [14] Their analyzed results and the mechanism are presented and discussed. In this paper, these results are applied to equalize the small voltages of EDLC cells. Another splendid equalizer was also proposed [11, 12], in which the cells are controlled by means of a string of reference capacitors and double groups of switches. The equalizing operation is a little analogous to the CW circuit, so its principle is interesting. The circuit and its operation, however, is a little complicated, and yet the operational principle is entirely different. Under such background, analytical and experimental results will be presented and discussed.

II. CIRCUIT CONFIGURATION AND OPERATION

Figure 1 shows the proposed fundamental circuit for the equalized charging of EDLCs, using CW (Cockcroft-Walton) circuit. C_1^* to C_5^* , on the left-hand side, indicate, for example, electrolytic capacitors, which have relatively uniform values and can be obtained at low cost. e is ac power supply, which often uses commercial frequency. The purpose is not to supply the output power to these, but to supply relatively reduced power in order to compensate the EDLC of voltage unbalanced.

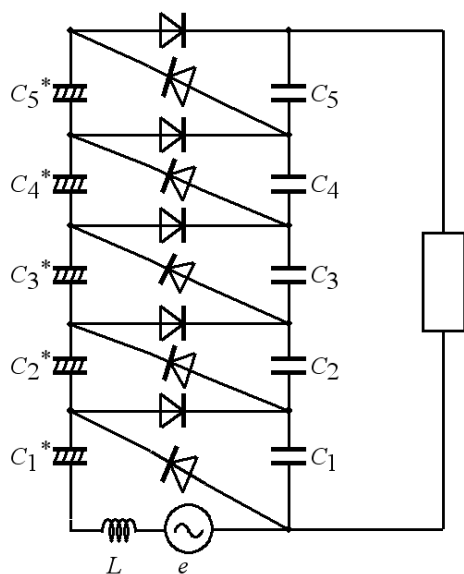


Figure 1. Basic novel voltage equalizer

Let us discuss about charging the C_1 on the first stage. First, C_1^* is charged in the first half cycle to $V_{C1^*} = E$, where E is average value over half cycle. After that, in the second half cycle, C_1 is charged by the sum of power supply e and C_1^* voltage. As a result, C_1 is charged toward $V_{C1} = 2E$. The mechanism of the proposed circuit using CW circuit is analogous to SC (switched capacitor) method [7]. By means of switching between two groups of capacitors, each different value of voltage can be averaged between each device. In SC method, the voltage can be transferred towards the both direction, upper and lower sides, however, in the proposed CW circuit the voltage is transferred only in one direction. However, alternative circuits are attached to the lower terminal of the power supply e , the corresponding voltage can be transferred towards such lower side, In conclusions of this chapter, in the proposed circuit, as shown below the number of switches can be reduced to single from many ones in the conventional voltage equalizer systems. The ac supply voltage shown as e , sometimes called as ac voltage exciter, might have not strict value. Even if the supply voltage is given by rough value,

the corresponding capacitor voltages can become a desired value. The reason can be explained as follows; if the voltage is large than we thought, the total piled up terminal voltage is increased. As a result, the discharged current or ac supply current is also increased, so the voltage drop across inductor L is increased, leading to the reduced ac voltage exciter.

III. BUCK-BOOST CHOPPER METHOD

For another example of circuit configuration, we have proposed a modified circuit, which is constructed using buck-boost chopper. Their chopping frequency is high, and by turning-on and turning-off with suitable duty cycle, a certain ac voltage is generated across the inductance L_1 , which plays the role of ac voltage power supply, as above mentioned ac power supply. In an analogous manner, each voltage on the EDLCs can be controlled uniformly. At the bottom side of the circuit, chopper exciter is constructed by familiar Zeta converter. As such converter, the circuit can play a role of buck-boost chopper, which is employed as chopper exciter without switching surge.

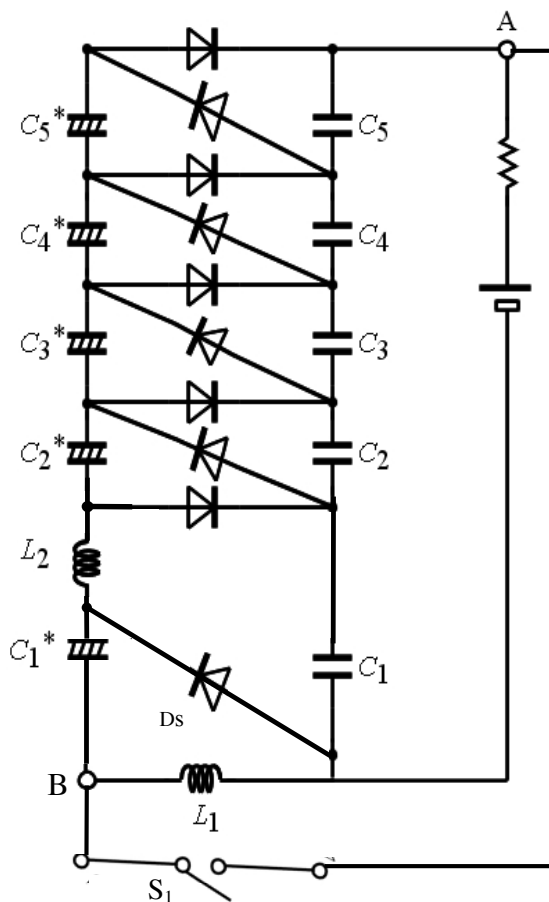


Figure 2. Voltage equalizer using chopper

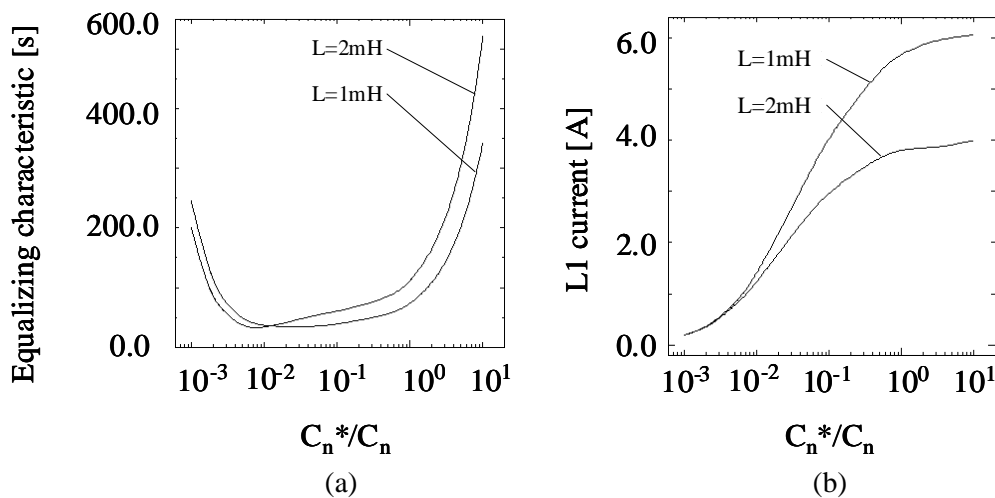


Figure 3. Convergent characteristics

IV. IMPROVEMENTS OF FUNDAMENTAL CW CIRCUIT

A. Improvement of Dynamic Response

In the fundamental CW circuit as voltage equalizer in Figure 1, the boost circuit on the left-hand side is constructed by electrolytic capacitors shown by C_1^* to C_5^* . Even though by increased such capacitance, however, the long charging time is required, leading to long equalization compensating time. In terms of dynamic response, it is disadvantageous as practical applications. From such reasons, such dynamic response characteristic is examined and resolved as follows; The EDLCs, C_n on the right-hand side are kept constant, while electrolytic capacitors, C_n^* on the left-hand side are made gradually increased, by which convergent characteristics can be obtained as shown in Figure 3(a). For reduced value of C_n^* , as the charging and discharging current are much smaller, so convergent time is limited by such reduced capacitance, leading to deteriorated dynamic response. For such case of significantly reduced C_n^* compared to C_n , the dynamic response is poor. An adequate selection for C_n^* could be much important. In the figure, when C_n^* is gradually increasing and C_n^*/C_n reaches beyond 1/100, then the convergent characteristics are much improved as shown.

After keeping straight line curves with nearly constant, the characteristics is becoming deteriorated again from about $C_n^*/C_n=1$. The reason can be explained that the charging current is gradually saturated about from that $C_n^*/C_n=1$ due to rush current suppression inductor L in Figure 1. From such saturated point, the increasing rate of current is suppressed. Figure 3(b) shows the variation of current due to C_n^*/C_n . From about $C_n^*/C_n=1$, the current is saturated by suppression inductor L . By means of such reasons, we can obtain a conclusion that the optimum

convergent characteristic can be obtained between $C_n^*/C_n = 1/100$ and 1.0. The convergence deterioration by further increasing C_n^* is due to above mentioned additional inductor to suppress the excessive rush current. In such way, as C_n^* is increasing, the required electric charge is also increased. It can be seen that the optimum region of C_n^*/C_n exists.

B. Improvement of Dynamic Response by Devising the Circuit Arrangement

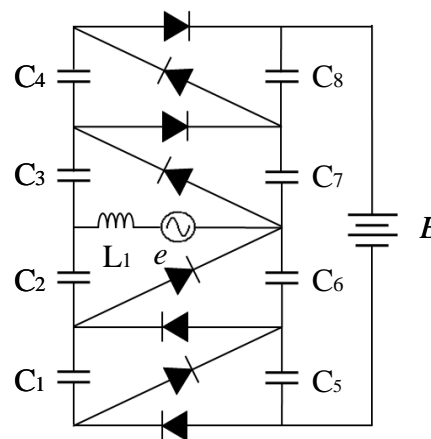


Figure 4. Modified equalizer in the CW circuit

In this section, to improve the dynamic response, the CW circuit is a little modified, where another circuit is added at the lower side as well as the upper one in the fundamental circuit. That circuit is shown in Figure 4. In this figure, the voltage of C_2 and C_3 are charged in half compared to the other capacitor one. Though the circuit operation has two equalizing directions, the principle of the voltage equalization is the same as the fundamental

one in Figure 1. The convergent characteristic for the conventional one is shown as case A and the proposed modified one is shown as case B in Figure 5. With compared between both cases, the convergence time is improved at twice over the whole region, where the horizontal axis shows the ac supply voltage. As it can be seen, as the voltage is increasing, the convergent time is becoming gradually improved. For modified one as case B, dynamic response can be much reduced to half as mentioned.

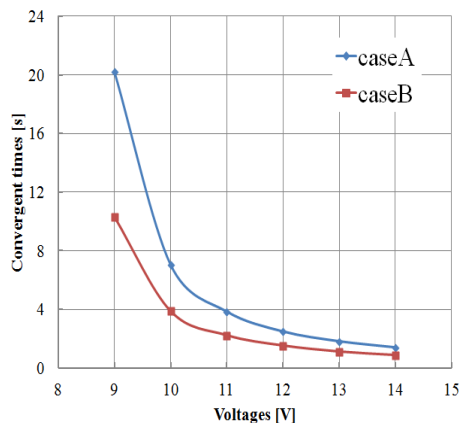


Figure 5. Convergent characteristics for driving voltage

V. IMPROVEMENT OF DEVICE UTILIZATION FACTOR

In previous publications, the authors have discussed a novel voltage equalizer under various view points. Using the CW circuit, the circuit is constructed by a chopper or an ac power supply, and the boost circuit of electrolytic capacitors on the left-hand side. If the boost circuit on the left-hand side is replaced by EDLCs, such capacitors could be sufficiently utilized by energy storage devices that device utilization factor for energy storage capability could be much increased.

A. Ac Power Supply Method

Figure 6(a) is a fundamental voltage equalizer using ac power supply, e as an exciter. Both C_1 to C_3 on the left-hand side and C_4 to C_6 on the right-hand side are entirely EDLCs to charge and discharge with voltage equalization function. L_1 to L_3 are for rush current suppression, where L_2 is for charging and L_3 is for discharging C_1 to C_3 with respect to external power supply. After C_1 is charged by ac power supply e , the following discharge current due to reversed voltage of e is prevented by diode D_1 . At this time, the electric charge of C_1 is transmitted to C_4 through L_3 - C_1 - C_4 , and can be obtained an equality, $V_{C1} = V_{C4}$

At the second stage, as each capacitor can be operated like the first stage operation. In such way, subsequent operation at the upper side, an analog operation can be

repeated like one of fundamental circuit in Figure 1. Thus, C_1 to C_6 can be charged in equalization. This circuit merit is that as the whole capacitors are employed as storage devices, the device utilization factor can be much improved. As the whole capacitors can play the role of storage devices having identical voltage, there is no need to make a particular specification as compared to the basic CW equalizer having different value of electrolytic capacitors and voltage.

B. Chopper Circuit Method

Figure 6(b) shows another proposed method using chopper circuit instead of ac power supply in Figure 6(a). By means of switching operation of S_1 , the voltage is applied across the L_1 , by which the similar operation is performed compared to that of Figure 6(a). Thus, C_1 to C_6 can be made equal voltages.

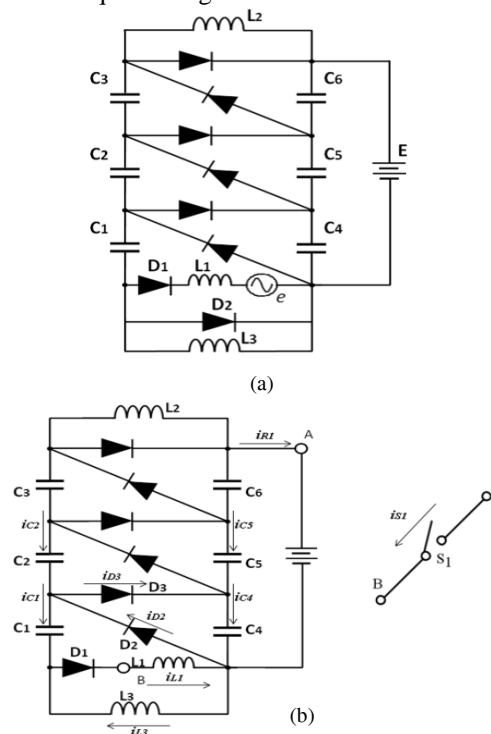


Figure 6. Novel equalizers. ac voltage exciter (a) and chopper exciter (b)

The operation mechanism can be described as follows. By means of S_1 turning-on, magnetic energy is stored in L_1 , whose energy is discharged by the subsequent turning-off and stored in C_1 . As far as L_1 energy is concerned, there is no surge generation through closed loop during turning-off. In the ac voltage exciter method in Figure 6(a), there is unnecessary current loop like L_1 - e - L_3 , during positive polarity ac voltage, where L_1 , energy is lost towards L_3 . For a case of chopper excitation in Figure 6(b), however, such unsatisfactory current could be much reduced, leading to make circuit specification simpler.

VI. OPERATION CHARACTERISTICS

A. Ac Power Supply Method

Figure 7 shows the voltage equalizing characteristics for ac voltage excitation in Figure 6(a), where as voltage is 35 V, frequency is 1 kHz, external dc voltage is 45 V, capacitance of C_1 to C_6 is 0.2F, where C_2 and C_4 is 0.1F, which are assumed to be deteriorated by aging. The initial voltage is given by $C_1 = C_3 = C_5 = C_6 = 11.25V$, $C_2 = C_4 = 22.5V$, respectively. It can be seen that after the excitation starts, each EDLC, C_1 to C_6 is converging towards desired voltage 15 V in about 9 sec. Even though different value of capacitance, each voltage is converging toward such desired voltage.

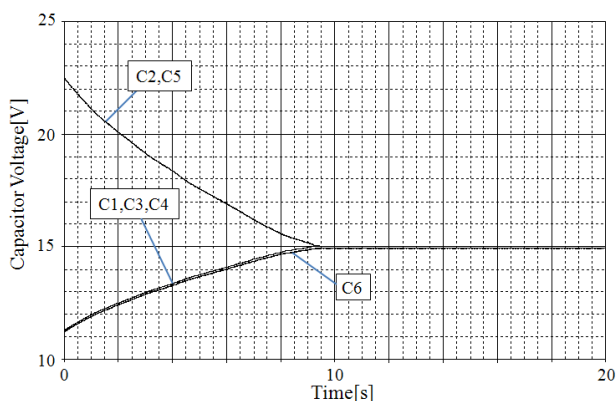


Figure 7. Voltage converging characteristics by ac supply exciter

B. Chopper Circuit Method

Figure 8 shows the convergent characteristic for Figure 6(b). The duty cycle of chopper, $d = 0.02$, dc power supply $E = 45V$, C_1 to $C_6 = 0.2F$, where C_2 and C_4 are assumed to be deteriorated by aging. That is, 0.1F. The initial voltages are $C_1 = C_3 = C_4 = C_6 = 11.25$ and $C_2 = C_4 = 22.5V$. EDLCs of C_1 to C_6 can be converged to the desired value of 15V in about $t = 47s$. As it can be seen, it is required to take a long converging time. The reasons could be described as follows; because of unstable specification due to reduced duty cycle and of the significant influence of the external dc power supply having constant voltage source, it would require to take a long time to reach in a stable state.

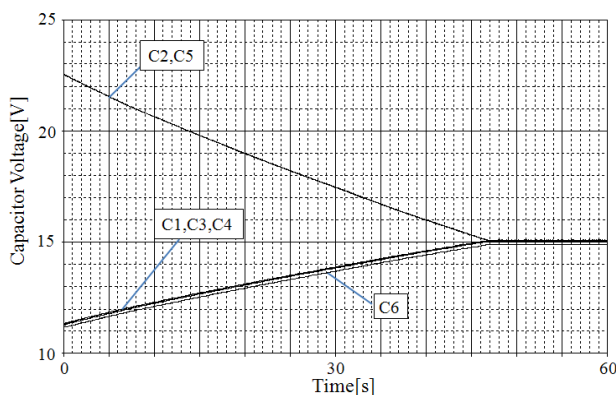


Figure 8. Voltage balance characteristics by chopper exciter

Figure 9 shows various operational waveforms when the convergent characteristics can be obtained by chopper excitation method in Figure 6(b). Because of stable state after reaching to convergent, each current is fairly reduced. By means of switch S_1 turning-on, the current i_{S1} is increased linearly. As a result, the direct connected next inductor L_1 is also increasing in a similar as i_{L1} .

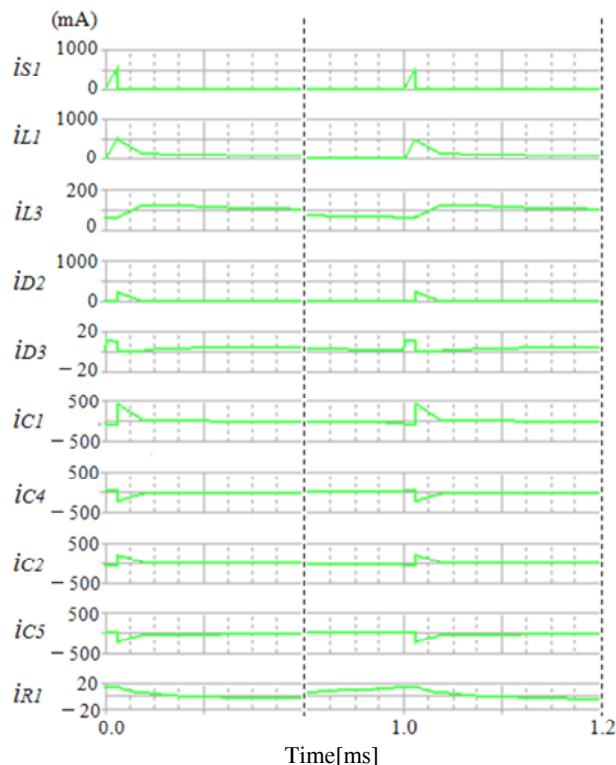


Figure 9. Current waveforms for chopper exciter

VII. CONCLUSION

A novel voltage equalization using a CW circuit with ac supply or chopper has been proposed and discussed. Various modified version have been also considered. It is easy to apply, because of its simple and concise construction. The purpose of this system is not to obtain boosted power, but to correct unbalanced voltage. Since such compensating power is not large, the ac power supply or chopper is small in size.

On the first stage in this paper, Zeta converter as buck-boost chopper is presented as a novel version, where the circuit operation is analyzed and specified. The proposed method needs a corresponding number of electrolytic capacitors. However, it may be possible that such capacitors can be replaced by EDLCs, in which case,

the system capability in capacity could be somewhat increased. In such purpose, a novel voltage equalizer is newly proposed. Since every device is constructed by EDLCs, the circuit specification could be simple and the dynamic response could be expected to be much improved because of transmittal current can be more increased.

The proposed method has a slight disadvantage, in terms of circuit response, because the command is gradually delivered from the bottom to the top side. As far as the circuit construction is concerned, however, the proposed configuration piles up multiple devices in succession, so the extension of a circuit is very easy.

In this paper, voltage equalizer for EDLC is discussed. For various secondary batteries like lithium-ion battery, however, the proposed system could be applied in a similar manner. Especially, in an electric vehicle, such batteries are much expected to be practically employed at low cost. If the proposed system could be applied to such ones, simple voltage equalizer might be realized in the near future.

REFERENCES

- [1] Michio Okamura: "Electric Double Layer Capacitors and its Energy Storage Systems", 3rd Edition, Nikkan Kogyo Shinbun-sha, 2005
- [2] Akitoshi Minemura, Masahiro Yashiro, Yasuyoshi Kaneko, and Shigeru Abe: "Equalization of the Voltages Using Passive Resistors for Electric Double Layer Capacitors", The 2007 National Convention Record of IEE Japan, no. 4-018, 2007
- [3] Philippe Barrade, Serge Pittet, and Alfer Rufer, "Energy storage system using a series connection of EDLCs, with an active device for equalizing the voltages" IPEC-Tokyo-2000, pp. 1555-1560, 2000
- [4] Alfer Rufer and Pilippe Barrade, "A EDLC-Based Energy-Storage System for Elevators with Soft Commutated Interface", IEEE Transaction on Industry Applications, vol. 38, no. 5, Sept/Oct 2002
- [5] Takatsugu Kishi and Toshihisa Shimizu, "A Study of Voltage Balancer for Electric Double Layer Capacitors", Technical Meeting on Semiconductor Power Converter SPC-04-37, 2004
- [6] Kazuya Mori, Akio Hasebe, Kiko Tsuruga, Takahiko Itoh, and Sumiko Seki, "Voltage Balancer for Electric Double Layer Capacitors", The 2001 National Convention Record of IEE Japan, no. 4-207, 2001
- [7] Eiji Sakai, Koosuke Harada, S. Muta, and Kiyomi Yamasaki, "Switching Converters using Double-Layer Capacitors as Power Backup", The 19th International Telecommunication Energy Conference, Proceedings of IEEE-Intelec 1997, pp. 611-616, Oct 1997
- [8] Nasser H. Kutkur, Deepak M. Divan and Donald W. Novotny, "Charge Equalization for Series Connected Battery Strings", IEEE Transaction on Industry Applications, vol. 31, no. 3, pp. 562-568, May/June, 1995
- [9] H. Sakamoto, K. Murata, E. Sakai, K. Nishijima, K. Harada, S. Taniguchi, K. Yamasaki, and G. Akiyoshi, "Voltage Balanced Charging of Series Connected Battery Cells", The 20th International Telecommunication Energy Conference, Proceedings of IEEE-Intelec 1998, pp. 311-315, Oct 1998
- [10] Keiju Matsui, Hiroto Shimada, and Masaru Hasegawa, "Novel Voltage Balancer for an Electric Double Layer Capacitor by using Forward Converter", The 4th International Telecommunication Energy Special Conference, Vienna Austria, Proceedings of IEEE-telecon 2009, II.3-1, pp. 1-6, May, 2009
- [11] Kimihiro Nishijima, Hiroshi Sakamoto, and Koosuke Harada, "Voltage Equalizing System for Series Connected Battery Cells", IEICE Trans. On B, vol. J84-B, no. 9, pp. 1701-1708, Sep 2001
- [12] Jonathan W. Kimball, B. T. Kuhn, and P. T. Krein, "Increased Performance of Battery Packs by Active Equalization," IEEE Vehicle Power and Propulsion Conference, pp. 323-327, Sep 2007.
- [13] J. D. Cockcroft and E. T. S. Walton: "Further development on the method of obtaining high velocity positive ions", Proc. Royal Society London, UK, 1932.
- [14] Keiju Matsui, Isamu Yamamoto, Masaru Hasegawa, and Hiroto Shimada, "A Novel Voltage Balancer for EDLCs Using Cockcroft-Walton Circuit", 2008 National Convention Record IEE Japan, vol. 4, 4-138, pp. 230-231, March 2008
- [15] Nasser H. Kutkur, Herman L.N.Wiegman, Deepak M. Divan, and Donald W. Novotny, "Design Considerations for Charge Equalization of an Electric Vehicle Battery System", IEEE Transaction on Industry Applications, vol. 38, no. 5, pp. 28-35, Sep/Oct 1999
- [16] Keiju Matsui, T. Suzuki, H. Shimada, M. Hasegawa, and K. Ando, "Further Development on Voltage Balancer for EDLCs Employing Cockcroft-Walton Circuit", Proceedings of IEEE -Intelec -2009, p. PES-4.1-6

ASIP for Multi-Standard Video Decoding

Jae-Jin Lee, KyungJin Byun and NakWoong Eum
 Multimedia Processor Research Team
 Electronics and Telecommunications Research Institute
 Daejeon, Korea
 {ceicarus, kjbyun, nweum}@etri.re.kr

Abstract—Multiple international video standards in the market have been developed successfully for many commercial products. Application-specific instruction processor is a new design methodology to develop optimized processor. This paper proposes a new application-specific instruction set processor based on 6-stage pipelined dual issue VLIW+SIMD architecture and compiler for multi-standard video decoding. The processor takes 130K in gate count at 125MHz in 130nm technology. Compared to the existing ARM processor, the proposed processor results in about 20% speed improvement as well as smaller hardware complexity.

Keywords—multimedia processor; application-specific instruction processor; video decoding.

I. INTRODUCTION

In the implementation of embedded systems, the designers confront with decision of architectures which is combination of ASICs [1], FPGAs [2], ASIPs (Application Specific Instruction-Set Processors) [3], DSPs [4], and GPPs (general purpose processors) [5]. These decisions are mainly based on performance, power consumption, flexibility and silicon area of systems. ASIPs are powerful solutions when the contradicting requirements such as performance and flexibility have to be jointly satisfied with a single task block. The flexibility of ASIP is caused by the necessity to support multi-standard video codecs such as AVS [6], VC-1 [7], and H.264 [8] in a single platform. On the other hand, next generation video codecs require extreme demands on throughput and processing of continuous data streams at high rates.

Video compression technologies have been dramatically evolved by many researchers and industries. Successful multiple international standards in the market have been released for last two decades. In particular, ISO/IEC WG11/MPEG and ITU-T SG16/VCEG have developed MPEG-1/2/4 and H.261/262/263 to compress raw digital videos since early 1990 [9][10][11]. Subsequently, the MPEG and VCEG jointly standardized the H.264/AVC [8] which is suitable for various network environments and gives the highest coding efficiency in 2003. In recent years, various video codecs such as VC-1 and so on have been commercialized, aside by the standard codes developed by the international standardization bodies. This leads a huge amount of multimedia contents to be compressed with the

increasing number of video coding techniques and distributed over various networks and devices.

ASIP is a new design methodology to develop optimized processors for specific applications by adding specific instructions into base instructions for eliminating functional hot spot of applications [3]. In terms of video decoding [12][13][14], the ASIP has higher performance than DSP because of its optimized application specific instructions, and has better flexibility and reusability than ASIC because any applications can be implemented with software.

The remainder of this paper is organized as follows. In the next section, ASIP and compiler for multi-standard video decoding are briefly overviewed. In Section 3, we have evaluated the proposed ASIP. Finally, we summarize the paper and conclude it mentioning future works.

II. ASIP AND COMPILER

As shown in Figure 1, the proposed ASIP providing a separate data and program memory (Harvard architecture) consists of dual issue VLIW (Very Long Instruction Word) + SIMD (Single Instruction Multiple Data) core, program/data cache interface, general purpose register file consisting of 16 32-bit registers, special purpose register file for SIMD instructions and bus interface to access external memory.

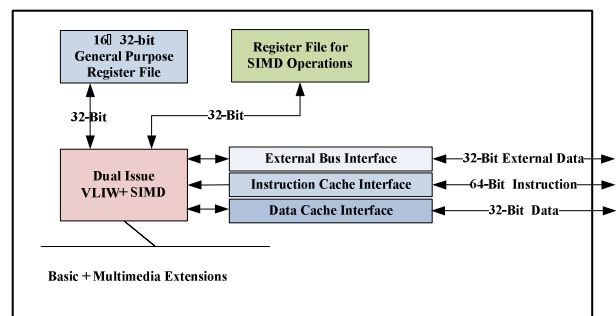


Figure 1. Block diagram of ASIP

The behavior, the structure, and the I/O interface have been described using LISA (language for instruction set architecture) [15]. It parses the description and generates the tools and models necessary for software design and architecture implementation such as assembler, disassembler, linker, ISS (Instruction Set Simulator). C compiler has been generated using the C-compiler designer tool of CoWare [16]. It provides a rich set of optimization

and restructuring engines that include typical high level optimizations such as copy and constant propagation, code motion, loop unrolling, loop fusion, and etc.

A. Pipeline and Bypass Logic

The proposed architecture is based on a pipeline with 6 stages as shown in Table I. The pipeline is fully bypassed, i.e., instructions reading from register R can directly follow the instruction writing to the same register.

TABLE I. PIPELINE OF THE PROPOSED PROCESSOR

Stages	Descriptions
PF(PreFetch)	Branch address or zero-overhead loop detection
FE(Fetch)	Fetch from the instruction memory
DC(DeCode)	Decode the instruction and read the operands
EX(EXecution)	Execute the ALU or logical operations
MEM(MEMory)	Read or write the memory
WB(WriteBack)	Write the results back into registers

The bypass logic [17][18] ensures a consistent read access between the instructions and makes sure that the latest result for a register is read by the instruction. A bypass is required when an instruction X in the execute stage (EX) is producing a result that is read by the following instruction Y. Instruction Y is at that time in the decode (DC) stage and requesting the result. A bypass allows instruction Y to access the result before it is actually written back into the main register file.

The MAC (Multiply-Accumulate) is very beneficial to speed-up many different type of applications. As shown in Figure 2, the proposed architecture has the pipelined dual cycle MAC supported by C compiler by sharing multiplier.

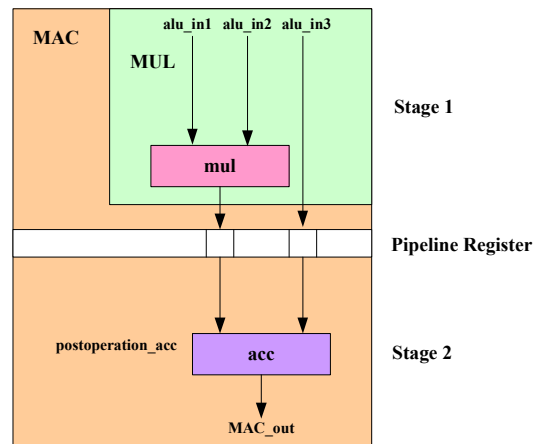


Figure 2. Dual cycle MAC

Figure 3 provides a block diagram of the bypass logic. It can be seen that the main register file is accessed in the DC stage. The operands are immediately pushed into the pipeline registers “op1”, “op2” and “op3”. If a custom instruction would need these operands already in the DC stage, the values can also be written into a signal instead of a register. Thus, those signals can be used to any combinatorial data-path in the DC stage. In EX stage, the latest operand value is written into the signal “alu_in1”, “shifter_in1” and “shifter_in2”. Once the result is computed the “writeback_dst” operation need to be activated (not shown here) and the register address “BPR” as well as the writeback value “WBV” need to be written.

B. Instruction Set

The Instruction set of the proposed processor consists of basic load/store, arithmetic, logic, branch and trap instructions, and multimedia extensions for multi-standard video decoding as shown Table II.

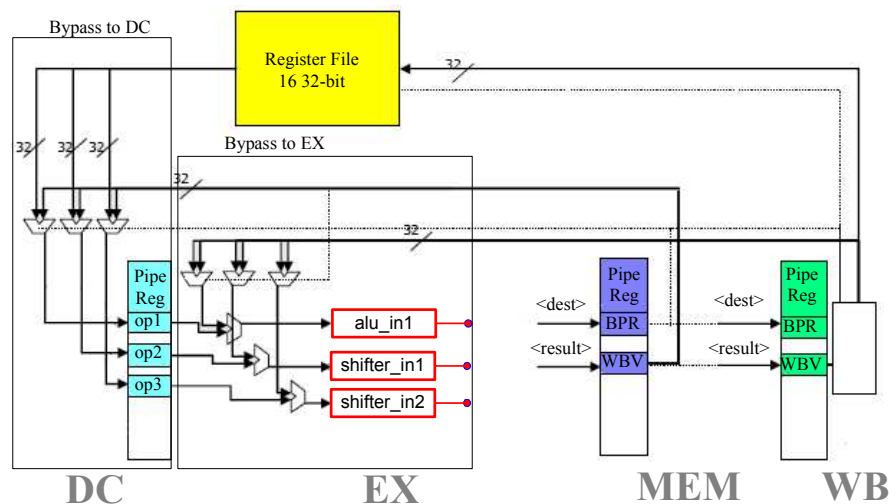


Figure 3. Block diagram of bypass logic

TABLE II. MULTIMEDIA EXTENSIONS FOR MULTI-STANDARD VIDEO DECODING

Instructions	Description
lmax	Compute maximum value of the two operands ex) int32 lmax_ckf(int32 data1, data2)
lmin	Compute minimum value of the two operands ex) int32 lmin_ckf(int32 data1, int32 data2)
smul_4	Computes the signed multiplication of 4x8-bit input operands ex) int32 smul_4_ckf(uint32 data1, uint32 data2)
umul_4	Computes the unsigned multiplication of 4x8-bit input operands ex) uint32 smul_4_ckf(uint32 data1, uint32 data2)
labs_4	Compute SIMD ABS ex) int32 labs_4_ckf(int32 data1, int32 data2)
bilf	Performs bi-linear filtering operation of 4x8-bit data1 and data2 ex) uint32 bilf_ckf(uint32 data1, uint32 data2, uint32 round_flag)
lclip	Performs clipping operation of data1 ex) uint32 lclip_ckf(int32 data1, int32 min, int32 max)
bext	Performs byte extension of data1 Extension type is determined by flag value (MSB or LSB) ex) uint32 bext_ckf(uint32 data1, uint32 flag)
add_clip4	Performs SIMD addition and clipping operations 2x16-bit data1 and data2 and 4x8-bit data3 ex) uint32 add_clip4_ckf(uint32 data1, uint32 data2, uint32 data3)
clz	Counts the leading zeros of data1 ex) int32 clz_ckf(uint32 data1)

As show in Table II, the proposed processor has special SIMD instructions for multi-standard video decoding. To extract these multimedia extensions, we have performed in depth profiling of existing multi-standard video decoding software such as MPEG-2, MPEG-4, AVS, VC-1 and H.264/AVC.

By applying the proposed multimedia extensions, we can effectively improve the performance of the video decoding algorithm. For example, the ‘add_clip4’ instruction is employed to add reconstructed residual values and prediction values and then clips the outcomes to 0~255 for four pixels at the same time in the ‘Reconstruction’ module. Since the prediction value is 8 bits/pixel and the residual value is maximum 16 bits/pixel in general video decoders, we have implemented an ‘add_clip4’ instruction to process four pixels at the same time, as shown in Figure 4.

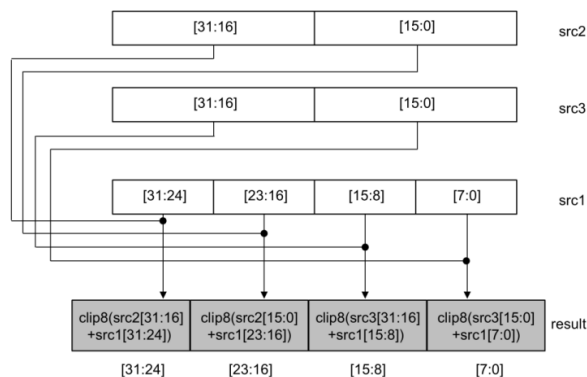


Figure 4. The operation of ‘add_clip4’ instruction

The number of inputs for ‘add_clip4’ is three and each input is 32 bits. 8-bit prediction values for the successive four pixels are entered into the first 32-bit parameter (src1) and the 16-bit residual data for the successive four pixels is entered into the second and the third parameters (src2, src3). Four successive pixels with addition and clipping operation can be processed in a single cycle. ‘clz’ instruction is efficiently used for the optimization of context adaptive variable length decoding.

C. C compiler

The compiler for the proposed architecture is developed with the C-compiler designer tool. Multimedia extensions are mapped by CKF, inline assembly and Matcher rules [19], respectively.

CKF stand for compiler known function. It is used to implement certain special instruction combination which would not usually be output by the compiler. The advantage of CKFs over inline assembly is that it gives more control to the C-compiler designer than to the user. Designers need not be aware of the assembly instructions that are required to implement the functionality.

C Compiler for ASIP provides a rich set of optimization and restructuring engines that include typical high level optimizations such as copy and constant propagation, code motion, loop unrolling, loop fusion and etc.

III. IMPLEMENTATION AND EXPERIMENTAL RESULTS

Although multimedia extensions proposed in this paper can be used to replace complicated operations efficiently, they are not enough to achieve the desirable performance by adopting only multimedia extensions. Essentially, C-level code optimization of the video codecs is required for real-time applications with minimum power consumption.

8-bit 4:2:0 format videos are widely used in the market. Most software video decoders mainly employ 8-bit memory access for the decoded picture buffers. However, most embedded processors support physical 32-bit memory access for memory load and store operations. Even though one pixel value is accessed by general software implementations, four bytes are loaded or stored into external physical memory. In video decoders, memory access is usually a bottleneck with higher resolution videos and this pixel-wire memory access could make the data access rate worse. Thus, we need to optimize the software with word aligned memory access for decoded picture buffers. For example, in the motion compensation module, a motion vector would be (5, 5) which are not multiples of four. The MPEG-2 decoder needs to load 17×17 pixels from the reference location (5, 5) because MPEG-2 employs half-pixel interpolation. However, the proposed decoder loads 20×20 pixels from the pixel (4, 4) by simultaneously considering the word-alignment memory access and the half-pixel interpolation. For H.264/AVC, as it uses the quarter-pixel interpolation with 6-tap FIR filter, two more left and upper pixels can be loaded and three more right and lower pixels are loaded. In other word, all the (21×21) pixels from (3, 3) to (23, 23) should be loaded regardless of the sub-pixel locations. As a result, (21×24) pixels from (3, 0) to (23, 23) are loaded by the proposed algorithm due to the word-alignment. The word-aligned position is calculated and the amount of alignment is defined by

$$\begin{aligned} aligned_x &= (((mv_x-2) \gg 2) \ll 2) \quad (1) \\ aligned_y &= mv_y-2 \end{aligned}$$

The LISA processor design platform offers the possibility to generate structured RTL model by grouping operations into functional units. Each functional unit in the LISA model represents an entity or a module in the HDL model. The generated Verilog RTL code for application specific processor is synthesized using Synopsys Design Compiler and implemented by SMIC 130nm cell library. The Figure 7 shows architecture of multi-core SoC (MOSAIC) including eight application-specific instruction processors and Table III shows features of the implemented multi-core SoC. As shown in Figure 7 the multi-core architecture consisting of four clusters including two ASIPs, DMA, TCM (Tightly Coupled Memory), inter-core buffer and communication manager. Two ASIPs are clustered for pipelined operations and each cluster is the basic unit for implementation of multi-core system. To reduce communication overhead, three types of hierarchical communication architecture such as private cache, shared cache and inter-core buffer have been proposed. PCIe interface is used for communication between host and target and video controller is for displaying decoded image.

The multi-standard video decoding algorithms are mapped into multi-core architecture by novel parallelization method called MB (macroblock) row-level parallelism [20].

Four CIF(352×288) video sequences such as ‘Foreman’, ‘Mobile’, ‘Paris’, and ‘Tempete’ are used for evaluating

decoding performance of the propose processor. Table IV shows detail encoding parameters of the test sequences and Table V shows the results of speed-up achieved by adopting multimedia extensions into four video decoding algorithms in terms of the decoding cycle. Compared to other decoders, speed-up of MPEG-2 decoder is very high because a large portion of the MPEG-2 decoding algorithm has been optimized with the proposed multimedia extensions

Figure 5 shows EVM board decoding VGA video stream encoded by MPEG-4 video standard. In addition to multi-standard video decoding, we have mapped various detection algorithms such as motion, lane and face detection algorithm into multi-core SoC.

The proposed processor results in about 20% speed-up in terms of processing cycles, compared to conventional ARM1020E processor [25]. Figure 6 shows the number of cycles required for ARM1020E and the proposed ASIP to process inverse transform and quantization of H.264 CIF test sequences.

TABLE III. SPEED-UP BY MULTIMEDIA EXTENSIONS

Sequences	MPEG-2	MPEG-4	AVS	H.264
Foreman	2.30	1.82	2.07	1.01
Mobile	2.07	1.58	1.68	1.01
Paris	2.29	1.76	1.84	1.01
Tempete	2.15	1.64	1.73	1.01
Average	2.20	1.7	1.83	10.1



Figure 5. EVM board of multi-core SoC (MOSAIC)

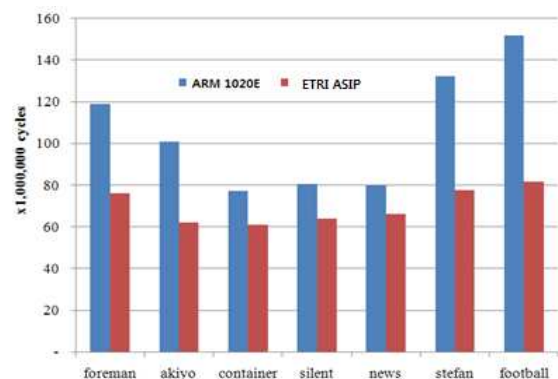


Figure 6. ARM 1020E vs. ETRI ASIP

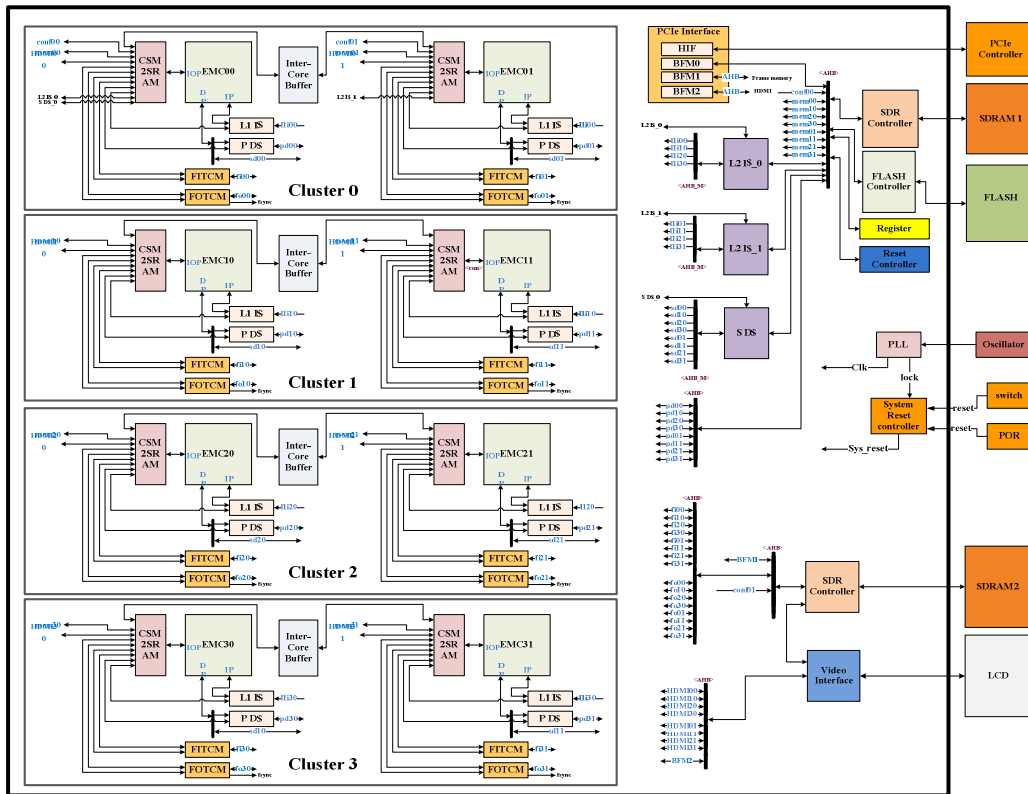


Figure 7. Architecture of multi-core SoC

TABLE IV. CHARACTERS OF MULTI-CORE SOC

Features	Features
Desig Rule	SMIC 130nm , 1P8M CMOS, Core 1.2V/Pad 3.3 V
Frequency	Core/SDRAM : 125MHz, PCIe : 62.5 MHz, Video Interface : 27 MHz
Internal PLL	8 ~ 175MHz/Programmable
Gate Count	1.3 M
Internal SRAM	789.7 KB
Power Consumption	225mA, 1.2V@125MHz
ChipSize/Package	8.12x8.12 mm2/308 FBGA

TABLE V. ENCODING CONDITIONS

Features	MPEG-2	MPEG-4	AVS	H.264/AVC
Profile@Level	Main @High	Advanced simple@L5	Jizhun@6.0	High@4.2
Coding Structure	IPPP	IPPP	IPPP	IPPP
Number of Frames	30	30	30	30
Encoder	TM 5 [21]	XviD [22]	RM5.2j [23]	JM 16.2 [24]
QP	rate control	rate control	rate control	27(I), 28(P)
Bitrate (foreman)	48 KB/s	44 KB/s	56 KB/s	48 KB/s
Bitrate (mobile)	238 KB/s	224 KB/s	252 KB/s	240 KB/s
Bitrate (paris)	80 KB/s	80 KB/s	88 KB/s	84 KB/s
Bitrate (tempete)	173 KB/s	172 KB/s	172 KB/s	176 KB/s

Figure 8 shows the decoding time speed-up according to the number of processing cores.

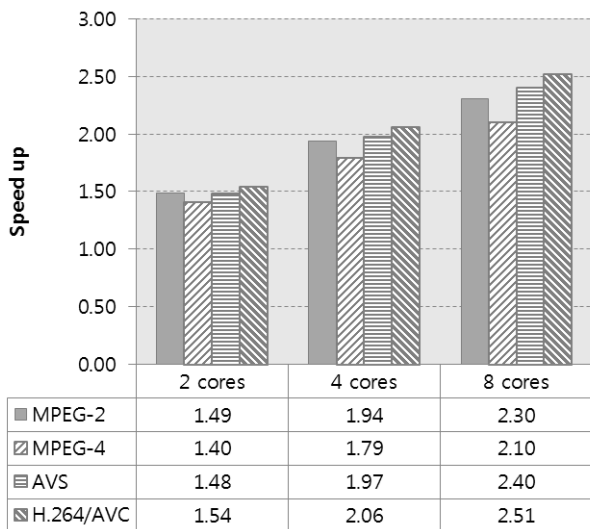


Figure 8. Speed-up according to the number of processing cores

In the case of MPEG-2 decoder, speed-up is 1.49x on 2 cores, 1.94x on 4 cores and 2.30x on 8 cores. In the case of MPEG-4, a speed-up is 1.40x on 2 cores, 1.79x on 4 cores and 2.10x on 8 cores. The AVS and H.264/AVC decoders also yielded a similar speed-up in accordance with the number of cores. Even with a multi-core implementation for video decoders, it is not easy to achieve more than 3x speed-up due to the sequential entropy decoding part and MB-to-MB dependency.

IV. CONCLUSION AND FUTURE WORK

This paper proposes a new application specific processor based on 6-stage pipelined dual issue VLIW+SIMD architecture and compiler for multi-standard video decoding. The proposed processor whose approximate gate count is about 130K runs at 125MHz in SMIC 130nm technology. The proposed processor results in about 20% speed-up in terms of processing cycles, compared to conventional ARM1020E processor without quality degeneration for the decoding of the H.264 CIF test sequences. For Full HD multi-standard video decoding, multi-core platform consisting of 64 ASIPs is under development.

V. ACKNOWLEDGEMENT

This material is supported by Ministry of Knowledge and Economy (MKE) and Korea Evaluation Institute of Industrial Technology (KEIT), Republic of Korea under Contract No. 10035152, Energy Scalable Vector Processor - Basic Technology.

REFERENCES

[1] Keith Barr, "ASIC Design in the Silicon Sandbox: A Complete Guide to Building Mixed-Signal Integrated Circuits," McGraw Hill, Dec. 2006.

[2] Arifur Rahman and Jason H. Anderson, "FPGA Based Design and Applications (Integrated Circuits and Systems)," Springer, Nov. 2012.

[3] A. Hoffmann, T. Kogel, A. Nohl, G. Braun, O. Schliebusch, O. Wahlen, A. Wiefierink, and H. Meyr, "A Novel Methodology for the Design of Application-Specific Instruction-Set Processors (ASIPs) Using a Machine Description Language," IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, vol. 20, pp. 1338-1354, Nov. 2001.

[4] Richard G. Lyons, "Understanding Digital Signal Processing (3rd Edition)," Prentice Hall, Nov. 2010.

[5] John L. Hennessy and David A. Patterson "Computer Architecture, Fifth Edition: A Quantitative Approach," Morgan Kaufmann, Sep. 2011.

[6] AVS-Group, "Information Technology - Advanced Coding of Audio and Video - Part 2: Video," advanced Audio and Video Standard (AVS1-P2), 2005.

[7] "VC-1 Compressed Video Bitstream Format and Decoding Process (SMPTE 421M-2006)," SMPTE Standard, 2006.

[8] ISO/IEC 14496-10 International Standard (ITU-T Rec. H.264)

[9] ISO/IEC 11172: "Information technology-coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s," Geneva, 1993.

[10] ISO/IEC 13818-2: "Generic coding of moving pictures and associated audio information-Part 2: Video," 1994, also ITU-T Recommendation H.262.

[11] ISO/IEC 14496-2: "Information technology-coding of audiovisual objects-part 2: visual," Geneva, 2000.

[12] F. Pescador, C. Sanz, M.J. Garrido, E. Juarez and D. Sampler, "A DSP Based H.264 Decoder for a Multi-Format IP Set-Top Box," IEEE Trans. Consumer Electronics, vol. 54, pp. 145-153, Feb. 2008.

[13] Y. Chen, E. Li, X. Zhou and S. Ge, "Implementation of H.264 encoder and decoder on personal computers." Journal of Visual Communication and Image Representation, vol. 17, pp. 509-532, April 2006.

[14] Y.-L. Lee and T.Q. Nguyen, "Analysis and Efficient Architecture Design for VC-1 Overlap Smoothing and In-Loop Deblocking Filter", IEEE Trans. Circuits Syst. Video Technol. vol.18 , pp. 1786-1796, Dec. 2008.

[15] S. Pees, A. Hoffmann, V. Zivojnovic and H. Meyr, "LISA-Machine description language for cycle-accurate models of programmable DSP architectures," Design Automation Conf., pp. 933-938, June 1999.

[16] C-compiler Design Guide, CoWare, 2011

[17] Processor Designer Training Manual, CoWare, 2011.

[18] LISA Language Reference Manual, CoWare, 2011.

[19] Compiler Designer Reference Manual, CoWare, 2011.

[20] J.Y. Lee, J.J. Lee and S.M. Park, "Multi-core platform for an efficient H.264 and VC-1 video decoding based on macroblock row-level parallelism," IET Circuits, Devices & Systems, vol. 4, pp. 147-158, Mar. 2010.

[21] <http://www.mpeg.org/MPEG/video/mssg-free-mpeg-software.html>

[22] MPEG-4 XviD, <http://www.xvid.org/Xvid-Codec.2.0.html>

[23] AVS RM5.2j, <http://www.avs.org.cn/fruits/en/softList.asp>

[24] Joint Video Team (JVT) reference software JM16.2, <http://iphome.hhi.de/suehring/tml/>

[25] ARM, <http://www.arm.com>

New Design Approach of an FIR Filters Based FPGA-Implementation for a Bio-inspired Medical Hearing Aid

Lotfi Bendaouia

*Equipe de Traitement de l'Information et des Systèmes
CNRS ENSEA UMR 8051
Cergy, France
lotfi.bendaouia@ensea.fr*

Hassen Salhi

*Département d'électronique et d'informatique
Université Saad Dahleb
Blida, Algérie
labset@yahoo.fr*

SiMahmoud Karabernou, Lounis Kessal

*Equipe de Traitement de l'Information et des Systèmes
CNRS ENSEA UMR 8051
Cergy, France
karabernou@ensea.fr, lounis.kessal@ensea.fr*

Fayçal Ykhlef

*Architecture des Systèmes et Multimédias
CDTA
Baba Hassen, Algérie
fykhlef@cdta.dz*

Abstract—The work focuses a new design approach to a hardware implementation of a Bio-Inspired Medical Hearing Aid which has the specificity to be portable and hence needs reduced resources and low power. This paper describes how we appropriately applied hardware optimization in the aim to reach on computationally intensive DSP algorithms for use to improve performance and efficiency of a hearing aid device on FPGA. Deaf people suffer from their social disease, so a device which could correct their hearing loss is needed. Even the technology advances these embedded devices can still be optimized for low cost. Contributions, mainly focus area reduction and hence low power consumption and dissipation. We propose a new design approach to meet the specifications of this embedded system.

Keywords—Hearing aid; DWT; FIR; area; area; Latency; Power consumption.

I. INTRODUCTION

Voice conversation is one of the most important tools for communication whereas some people don't have profit from this opportunity because of their hearing loss. Speech is badly detected by these impaired persons yielding to poor intelligibility. This happens mostly in noisy and reverberant environment [1][2].

Since many years, prosthesis and cochlear implants have been used but deaf persons are still feeling uncomfortable and suffer from this social handicap.

A Lot of researchers were held for speech enhancement leading to many contributions for algorithms development and circuits' design with less complexity and fast processing [3][4][5].

Knowing that it is difficult, say impossible, to reproduce the natural hearing for impaired persons, our contribution is to make the filtered signal more closely to the original one for better intelligibility and hearing comfort.

The DSP algorithms [6] were implemented using traditional DSP or general purpose microprocessors. It was released that they have limited capabilities for processing

high volume data efficiently at real time. The trends were shifted to specific processors such as Asics in order to meet the increased complexity and performance requirements of these algorithms but with high cost function.

The rapid growth in the industrial technologies has participated in the development of several and performed hardware digital signal processing application systems. The implementations of intensive computational DSP algorithms become for researchers a day to day application area for digital hardware platforms [7][8].

FPGA [9] has taken a large area of use because of some advantages over Asic technologies.

Several hardware platforms [10][11][12] were designed with different combinations and optimizations of filters structures. They were simulated and synthesized using Xilinx or Altera FPGA development kits.

The impact of these hardware optimization techniques on the overall DWT hardware system are analyzed and the tradeoffs between the pertinent hardware performance metrics particularly power consumption, latency, resource utilization and operating frequency are considered and investigated [13].

Today, FPGAs are highly preferred to the relatively high capacity and low cost, short design cycle and short time to market. They afford the capability of constant reconfiguration to meet application performances which are highly preferred.

Recent FPGA includes enhanced signal processing capabilities of high performance logic and inherent parallelism enabling FPGAs to have special Multiply-Accumulate (MAC) blocks within its hardware.

By using FPGA, the design can be simulated and then synthesized with low cost. The hardware design is then ready for fabrication and use.

Our work is focused in the implementation of an efficient multi level one dimension Discrete Wavelet Transform (DWT) on FPGA for medical hearing aid application.

The proposed architecture combines hardware optimization techniques to develop a flexible DWT architecture that has a high performance and is suitable for portability, high processing speed and power efficiency [9][10] in order to optimize the hardware, we have reduced FPGA resources by using some techniques and orientated the VHDL [14] program in order to have a synthesized IP-based on customized DSP slice resources of the FPGA.

The paper is divided into four main sections. An overview of deafness is presented in the next section. Section three illustrates how the DWT algorithm is modelled to the cochlea structure. In Section four, we present the implementation of the FIR filters and show the simulations, the synthesis and the appropriate approximations. Performance analysis of the whole system and the perspectives for future work are presented in the conclusion.

II. OVERVIEW OF DEAFNESS

Impaired persons are affected by some perturbation phenomena, namely noise and echo, which reduces the intelligibility and hence their capability for understanding and communicating. Some studies were conducted to assess the status of noise and echo stripping.

A. Noise Reduction

Noise overlaps speech signal and can be suppressed or reduced by a low pass filtering as shown by Figure 1. However, when applying such action on a signal, we sometimes rescue to the elimination of some singularities of the signal which can contain significant information.

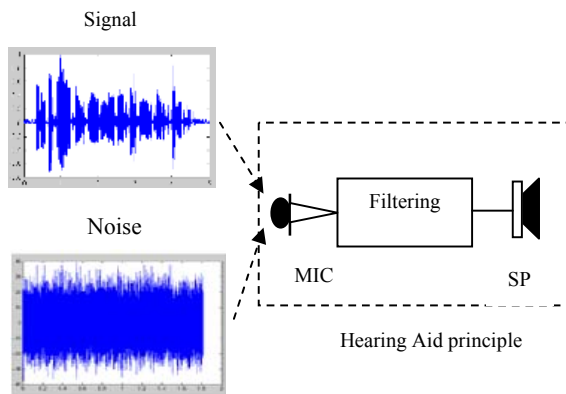


Figure 1. Speech signal versus noise separation .

In order to improve intelligibility, noise should be reduced but cannot be fully eliminated from speech because nothing is known about this latter and how far is it modeled with speech [15].

Care should be taken when applying denoising algorithms to avoid making any severe degradation of the resulting signal. The time-scale analysis is the perfect solution for speech processing as we will in Section III. Noise overlaps the speech signal in both time and

frequency; so, it is difficult to remove it completely. Hence, greatest attention was made in the development of noise reductions techniques.

B. Echo Cancellation

The Acoustic feedback refers to the acoustical coupling between the loudspeaker and the microphone of the hearing aid. The amplified sound sent through the loudspeaker is sometimes fad back into the microphone as shown in Figure 2.

As a result, the original signal gets distorted and then poor intelligibility occurs. One direct solution is to reduce the gain, but this limitation generates low energy making signals falling below the hearing threshold and no compensation is made to the impaired persons.

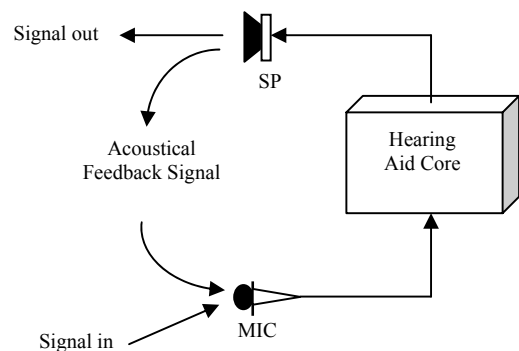


Figure 2. Acoustical feedback in hearing aids .

Acoustic feedback suppression techniques are suitable to increase the maximum gain of the system without making it unstable, as shown in Figure 3.

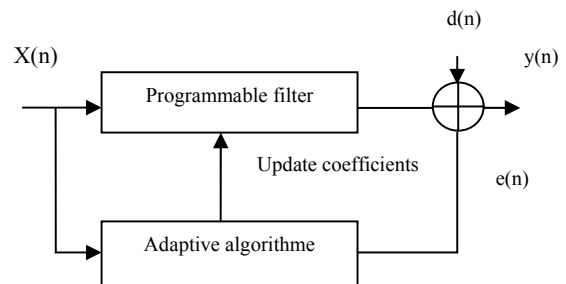


Figure 3. Adaptive filtering diagram .

Wiener adaptive filtering techniques [16] consider stationary signals and use, LMS [17], NLMS [18] or RLS [19] algorithms. For non-stationary signals, generalized techniques are used based on Kalman filtering [20].

The estimated error is given in [16] by:

$$e(n) = X(n) - \sum_{j=1}^m a_j X(n-j) \quad (1)$$

The objective is to find the optimal coefficients which make the error as small as possible. In order to do that, we should minimize the energy of the total error given by the equation:

$$E = \sum_{n=0}^{N-1} e(n)^2 = \sum_{n=0}^{N-1} [X(n) - \sum_{j=1}^m a_j \cdot X(n-j)]^2 \quad (2)$$

By making $\partial E / \partial a_j = 0$, we can obtain the coefficients a_j from the m generated equations.

III. THE BIO INSPIRED ALGORITHMIC MODEL

Before making the algorithm model, we first make an overview on how the basilar membrane acts as a group of mimicked filters.

A. Basilar Membrane Modeling (BMM)

The cochlea is an opened spiral tube lying in the middle ear. The opening in its base makes possible the penetration of the sound signals. The closed end is called the apex. The sound is detected and coded according to its frequency but it is place coding on the basilar membrane. The sounds of high frequencies are detected at the base, whereas those of low frequencies are detected at the apex. The frequencies are distributed along the basilar membrane in a very precise manner as represented by Figure 4.

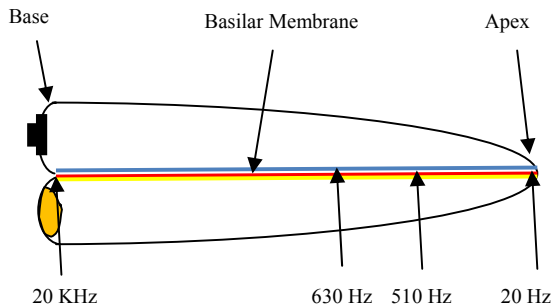


Figure 4. Longitudinal view of the cochlea .

The filters within the cochlea are distributed in bands along the basilar membrane and are responsible of the selectivity of the sound frequency in the ear. These filters can be modelled in a pseudo-logarithmic way. They are either triangular (Mels) or rectangular (ERB). The bands are linear up to 500 Hz and logarithmic beyond. Each bandwidth can be determined using the formula [16]:

$$\Delta f = 25 + 75 \left[1 + 1,4 \left(\frac{f}{1000} \right)^2 \right]^{0,69} \quad (3)$$

where f is the central frequency of the band.

B. Survey of the DWT

The Discrete Wavelet Transform [22] is an important approach for the analysis of a transient signal. The connection was made between the wavelet transform and multi-rate filter bank trees by Mallat since 1989 [13]. From signal processing point of view, the wavelet transform of a sequential signal is to recursively decompose a sampled sequence of a signal into two components in octave bands. A recursively asymmetric decomposition levels, leads to a similar bands distribution as the basilar membrane which make the DWT the adequate algorithm [23]. Three level wavelet decomposition is shown in Figure 5.

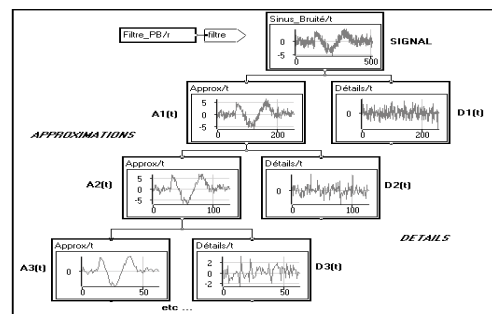


Figure 5. The level decomposition of Wavelet analysis .

The input signal is spread into two signals by the Low Pass Filter giving what we call the approximation of the signal and, by the High pass filter giving a detail of the signal at the first level. The process can be repeated on for other levels using a symmetric or asymmetric decomposition. The wavelet coefficients can be used to reconstruct the original signal without any distortions. The LPF and HPF are Finite Impulse Response (FIR) Filters. FIR filters are the basis of the DWT. For a dyadic representation, the basic analysis / synthesis structure of the DWT is represented by the Quadratic Mirror Filters (QMF) shown in Figure 6.

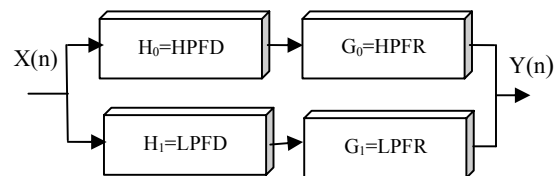


Figure 6. Quadratic Mirror filter structure .

where the H_x filters state for Decomposition (D) and G_x for Reconstruction (R). HPF means High Pass Filter and LPF is the Low Pass Filter. These filters are related by the following equations:

$$H_0(z) \cdot G_0(z) + H_1(z) \cdot G_1(z) = z^{-T} \quad (4)$$

$$H_0(-z) \cdot G_0(z) + H_1(-z) \cdot G_1(z) = 0 \quad (5)$$

By setting: $G_0(z) = H_1(-z)$ and $G_1(z) = -H_0(-z)$, we satisfy:

- Perfect reconstruction with latency (T)
- No aliasing

The coefficients are obtained by Matlab using Wfilters for different wavelet. They are stored as real and then converted to fix point numbers so as they can be treated in the hardware design circuit.

The selection of the FIR filters is due to coefficient sensitivity, round off noise, stability and are suitable for high speed applications [24].

The FIR filters use a convolution principle of the input signal $X(n)$ by the impulse response $h(n)$. The output $Y(n)$ is given by the following mathematical expression:

$$Y(n) = \sum_{k=0}^{L-1} h(k) \cdot X(n - k) \quad (6)$$

The FIR filter is composed of multipliers, adders and delay units. Recent FPGA includes DSP48A1 elements making ideal to implement DSP functions.

The n input samples from the data set are presented at the input of each DSP48A1 slice.

Each slice can be used to multiply these samples with the corresponding coefficients within the DSP48A1. The outputs of the multipliers are combined in the cascaded adders. A basic DSP48 slice is shown in Figure 7.

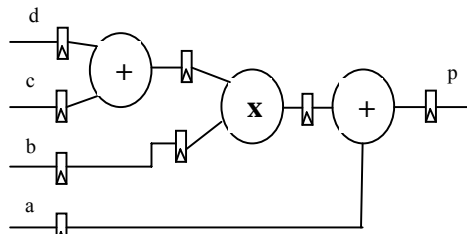


Figure 7. DSP Slice 48 A1 with Pre-Adder .

The sample delay logic is denoted by Z^{-1} , the (-1) represents a single clock delay. The delayed input samples are supplied to the one input of the multiplier. The coefficients represented by $(h(0) \text{ to } h(N-1))$ are supplied to the other input of the multiplier through individual ROMs, RAMs, registers or constants. The output $Y(n)$ is merely the summation of a set of input samples, and in time, multiplied by their respective coefficients. The DSP48A1 lying inside the FPGA is suitable for low power dissipation and high throughput based pipelining and parallel processing [25].

IV. ARCHITECTURE IMPLEMENTATION

The filter structure of FIR of length L (called order of the filter) is represented in Figure 8. This structure describes the relationship between the input and output sequences.

The input samples are delayed and multiplied by the suitable coefficients and then added to give the output at time n .

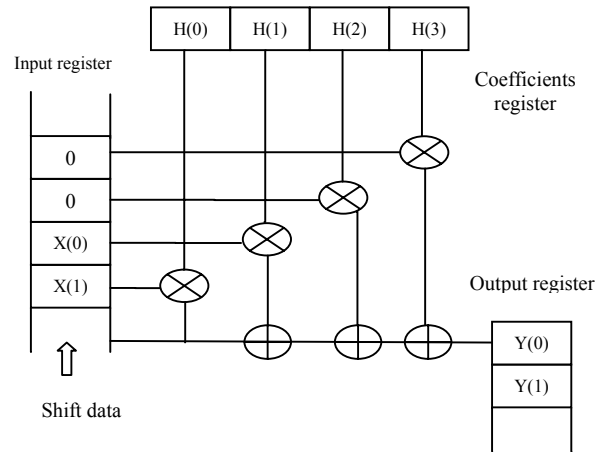


Figure 8. Convolution principle in FIR .

The architecture of each FIR block includes a FIFO register for data input, a register for the coefficients and the operators. The data output is stored in memory. The FIR block is presented in Figure 10.

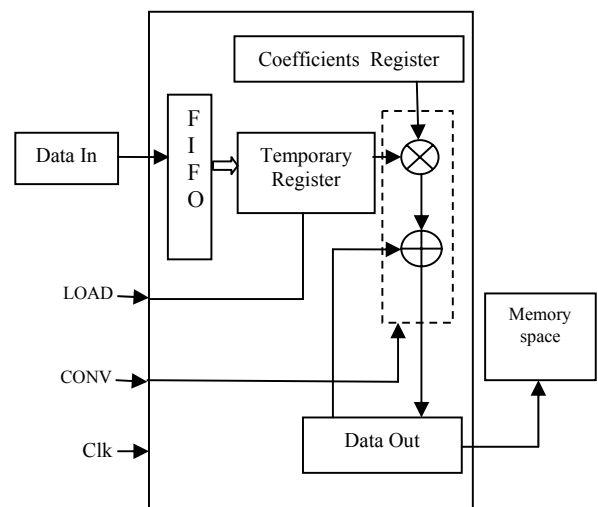


Figure 9. Basic FIR filters architecture Design .

The FIFO at the input is filled by the input data samples. An input register of length N is used to store the input sequence $X(n)$ taken from the FIFO. These samples are convolved with the coefficients which have been already stored in a coefficients register. The output sequence is also stored in a FIFO register (memory).

V. EXPERIMENTAL RESULTS

The experiments were held with a real man speaking speech signal “the Discrete Fourier Transform of a real value signal is conjugate-symmetric”. The wave signal is sampled at a frequency $F_s = 22050$ Hz. We take blocs of 10000 samples by Hamming windowing. For our experiments, Daubechies 4 was chosen. The filters coefficients are presented in Table I.

TABLE I. GENERATED FILTERS’ COEFFICIENTS FOR DB4

Coefficients	LPFD	HPFD	LPFR	HPFR
1	- 0.0106	- 0.2304	0.2304	- 0.0106
2	0.0329	0.7148	0.7148	- 0.0329
3	0.0308	- 0.6309	0.6309	0.0308
4	- 0.1870	- 0.0280	- 0.0280	0.1870
5	- 0.0280	- 0.1870	- 0.1870	- 0.0280
6	0.6309	0.0308	0.0308	- 0.6309
7	0.7148	- 0.0329	0.0329	0.7148
8	0.2304	- 0.0106	- 0.0106	- 0.2304

The input data samples and the coefficients were quantified and approximation tests were done for 8, 10, 12, 14, 16 bits. The better approximation was obtained for 12 bits and over.

The 12 bits quantization has been chosen for the rest of the experiment. The input samples and the coefficients have been converted to signed fix point data using a Q1.11 format. The output results after multiplications and additions are of format Q5.22. So, a truncation is done giving an output data of Q1.11 format with negligible errors. The samples are converted from floating to fixed point numbers, so they can be treated by the VHDL programs and then compared to those obtained by Matlab. A flow chart is given by Figure 11.

A. Simulations

In the VHDL simulations process, we generated the outputs of the design which are compared to the outputs generated at the algorithmic level using Matlab. We used for accuracy in metric estimation, the peak error and the Mean Square Error (MSE). A maximum performance is achieved with an error of less than 0.3 %.

In Figure 10, we make a comparison between the data obtained from the resulting output samples stored in the output files, using Matlab textread and plot functions.

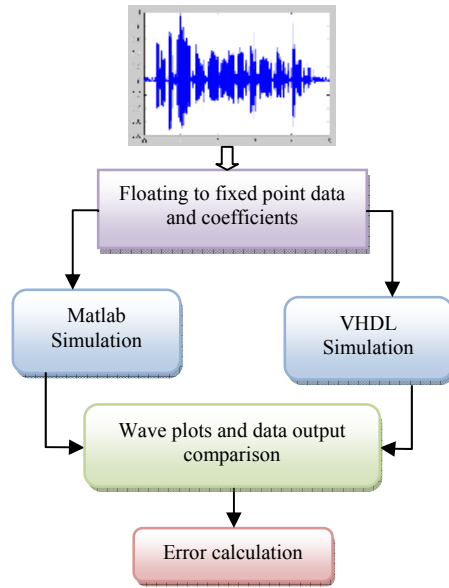


Figure 10. Matlab Versus VHDL simulation analysis .

B. Synthesis

For post-synthesis, we used the EP2C70F896C6 device of Cyclone II family at 100 Mhz clock. A VHDL Netlist containing Altera simulation primitives was generated and has been used again for correct compilation and simulation.

The design was synthesized using QuartusII giving the circuit at the Register Transfer Level (RTL) shown by Figure 11. The architecture needs registers of N bits if the data input is N bits but, our suggested architecture only needs L bits FIFO register and therefore we get an optimized structure for implementation.

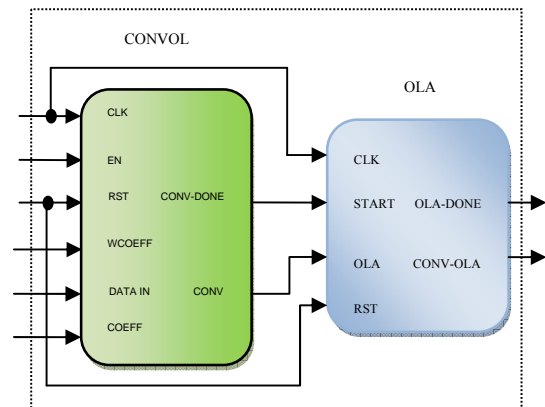


Figure 11. FIR block Diagram at RTL .

As shown by Figure 11, the basic FIR bloc is divided into two processing units. The first calculates the convolution coefficients with the L first bits and then with the L next bits with a save f the overlap data. Only a few bits are saved dealing to a gain of space memory.

The technological schematic is given by Figure 12. We can see that the circuit use combinatory components such as adders and multipliers. The system is controlled by a state machine which synchronizes the data acquisition and data treatment processes.

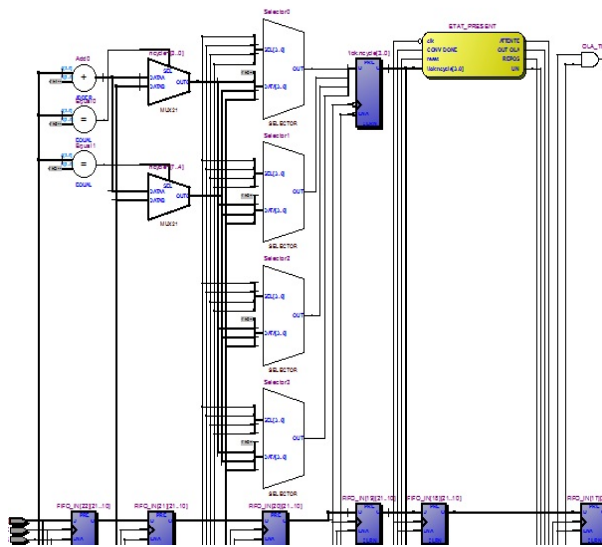


Figure 12. FIR Technological Block Diagram .

The performance metrics show the efficient contribution of FPGA resources in the implementation of the FIR filters which are the basic elements for the global system design (DWT). This hardware design has permitted the reduction of the used area; the used resources are less than 4% and the optimization of the latency and the power consumption. The result summary for resources utilization from synthesis is given by Table II.

TABLE II. EDA NETLIST FINAL REPORT

Logic elements	TF	DF
LUTs	1668 (2%)	6157 (9%)
Registers	1868 (3%)	3420 (5%)
Memory	240 (< 1%)	80640 (7%)
Multipliers	32 (21%)	96 (64%)

It is also compared to that obtained in previous work using direct form design. From the plotted graphs, we can see that the curves are totally overlapped, meaning of the very accurate results; see Figure 12.

VI. RELATED WORKS

The direct form (DF) FIR filters architecture was first implemented by Baganne et al. [26]. This architecture was modular and had low design complexity, low hardware

latency and could be easily expanded to further levels of decomposition. However, this architecture had a large critical path delay, needs more resources and high power dissipation.

We have first implemented this architecture in order to compare it with our new design approach [27]. For optimization, we apply the appropriate hardware pipeline and parallel techniques within the DF and use polyphase filters instead of decimation process. In this new architecture, we used the transpose form (TF) filters to reduce critical path delay. We also use an Over Lapp Add (OLA) computation method in order to overcome the memory space and the waiting time for filling the data in the input FIFO register.

VII. CONCLUSION AND FUTURE WORK

We have developed and tested a DSP system for hearing impaired persons. It incorporates wide bandwidth and a great deal of flexibility in adjusting the overall speech processing algorithm.

The DSP48A1 slices have been integrated into FPGA for Digital Signal Processing purpose. They are assembled in columns using less wiring which reduces internal connections and avoid critical paths and hence time latency. The use of DSP48A1 has allowed a high performance of the system and about 20 % of the power dissipation is gained.

The principle objective of this paper was to present the new design approach of a low cost and reconfigurable FPGA platform. It can be used to test the DWT algorithms with different parameters as to meet the specifications for different hearing pathologies.

We are currently pursuing research to improve our current algorithms and architecture design for further noise reduction and echo cancellation. Knowing that the perception of speech is highly subjective in nature, the system is subjected to be tested on human subjects in the real world environment. The patient’s response will determine the success or failure of this DSP system.

REFERENCES

- [1] N. Magotra and S Sirivara “Real time digital speech processing strategies for the hearing impaired” Tewas Instruments, Application Report, USA April 2000
- [2] J. Pang and S. Chauhan “FPGA design of speech compression by using discrete wavelet transform” Proceedings of the World Congress on Engineering and Computer Science WCECS08 Oct. 22-24, USA 2008.
- [3] Z. J. Mou and P. Duhamel “Fast FIR Filtering: Algorithms and Implementation” Signal Processing Vol. 13 N° 4 pp. 377-384 Dec. 1987.
- [4] L. Bendaouia, SM. Karabernou, L. Kessal, H. Salhi and F. Ykhlef “Fast DWT based FPGA implementation for medical application” IEEE International Conference on Phealth, Lyon, France June 2010.

[5] S. Powell and P. Chan “Reduced complexity programmable FIR Filters” IEEE Int. Symposium on Circuits and Systems pp. 561-564 May. 1992

[6] All programmable FPGAs <http://www.xilinx.com> , [Retrieved:April, 2012]

[7] F. J. Taledo Moreo, A Leraz Cano, J. J. Martinez Alvarez, J. Martinez Alajarin and R. Ruiz Merino, “Compression system for the phonographic signal” Journal of sociocybernetics Vol. 7 N°2 pp. 770-773, winter 2009.

[8] J. Chilo and T. Lindblad “Hardware of 1D wavelet transform on an FPGA for infrasound signal classification” IEEE Transaction on Nuclear Science Vol. 55 issue 1 pp. 9-13, 2008

[9] Tim Erjavec, “Introducing the Xilinx targetd design platform”, www.eetimes.com [Retrieved , February 2, 2009].

[10] K. Parki, “VLSI Architecture for discrete wavelet transform” IEEE Transaction VLSI Systems Vol. 1 pp. 191-202 June 1993.

[11] K.R. Borisagar, “Speech processing using wavelet transform and implementation for digital hearing aids” International conference on engineering trend, Pune, Dec. 2008.

[12] N.A. Ghamry and S.E. Habib, “An efficient FPGA Implementation of a wavelet Coder/Decoder” International conference on Microelectronics ICM 2000, Tahrn, October, 2000.

[13] R. Hourani, W Alexander and T. Raithatha “Automated design space exploration for DSP applications” Journal of Signal Processing Systems, Springer, 2009

[14] S. Chan, W. Liu and K. Ho “Multiplier less perfect reconstruction modulated filter banks with sum of powers of two coefficients” IEEE Signal Processing Letters Vol. 8 N° 6 pp. 163-166 June 2001.

[15] B. Cope, P.Y.K. Cheung and L. Howes “Performance comparison of graphics processors to reconfigurable logic: a case study” IEEE Transactions on Computers Vol. 59 N°4 April 2010

[16] “2D adaptive noise removal filtering”. www.mathworks.com [Retrieved:May, 2012]

[17] T. Fillon “Traitement numérique du signal pour une aide aux malentendants” Thesis ENST France April, 2005.

[18] K. R. Rekha, B. S. Nagabishan and N. R. Natary, “FPGA implementation of NLMS algorithm for the identification of unknown system”, International journal of engineering, science and technology, Vol. 2 (11) 2010.

[19] R.C.D dePaiva, W.P. Biscainho and S.L. Netto, “On the application of RLS adaptive filtering for voice pitch modification” Proc. Of the 10th International conference on digital audio effects , Bordeaux France, Sept. 10-15 2007.

[20] J. Flanagan and M. Saslow “Speech analysis, synthesis and perception” Springer, New York 2nd edition 1972.

[21] G. Kartik, M. Kumar and M. Rahman, “Speech enhancement using gradient based variable size adaptive filtering techniques” IEEE International Journal of Computer Science & Emerging Technologies Vol. 2, issue 1, Feb. 2011

[22] S. G. Mallat, “A theory for multiresolution signal decomposition : the wavelet representation” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 11 pp. 674-693, July 1989

[23] C. Chakrabarti, M. Vishuanath and R.M. Owens, “Architecture for wavelet transforms” VLSI Signal Proc. VI IEEE Special Publications pp. 507-515, 1993

[24] X. Hu, L. DeBrunner and V. DeBrunner, “An efficient design for FIR filters with variable precision” Proceeding IEEE International Symposium on Circuits and Systems Vol. 4 pp. 365-368 May 2002

[25] FPGA, DSP48A1 user Guide, www.datasheets.org.uk/250-DSP ,, [Retrieved:August 13, 2009]

[26] A. Baganne, I. Bennour, M. Elmarzougui, R. Gaiech and E. Martin, “A multi level design flow for incorporating IP cores: case study of 1D wavelet IP integration” in Design, Automation and Test in Europe Conference and Exhibition, pp. 250-255, 2003

[27] L. Bendaouia, SM. Karabernou, L. Kessal, . Salhi and F. Ykhlef, “DWT based FPGA implementation of a reconfigurable platform for a bio-inspired medical hearing aid” International Conference on Systems, Modeling and Design, Istanbul, Turkey Feb. 3rd-5th 2012

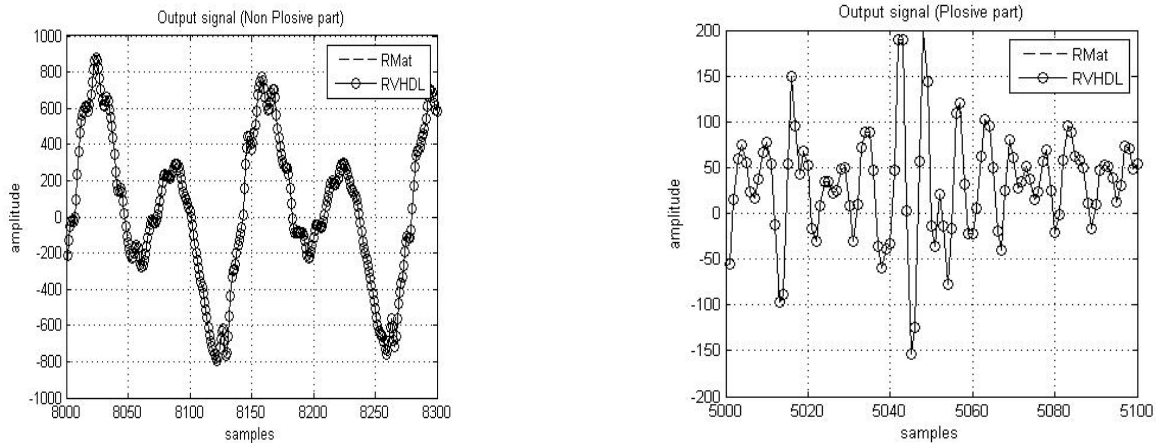


Figure 13. Matlab versus VHDL Output data analysis and comparison.

Reliable CMOS VLSI Design Considering Gate Oxide Breakdown

Kyung Ki Kim

School of Electronic and Electrical Engineering

Daegu University

Daegu, South Korea

e-mail: kkkim@daegu.ac.kr

Abstract—As technology scales down into the nanometer region, the reliability mechanism caused by time dependent dielectric breakdown (TDDB) has become one of the major reliability concerns. TDDB can lead to performance degradation or logic failures in nanoscale CMOS devices, and can cause significant increase of leakage power in the standby mode. In this paper, the TDDB effects on the delay and power of the nanoscale CMOS circuits are analyzed using ISCAS85 benchmark circuits that are designed using a 45-nm CMOS predictive technology model. Based on the TDDB analysis, a reliable CMOS VLSI design methodology using a redundancy system has been proposed.

Keywords-TDDB; Reliability; Gate oxide breakdown; Time dependent dielectric breakdown; Aging effect

I. INTRODUCTION

As MOSFET [1] technology is scaled down more aggressively, various aging phenomena (or reliability mechanisms) such as negative bias temperature instability (NBTI), hot carrier injection (HCI), and time-dependent dielectric breakdown (TDDB) on the MOSFET device have become one of the most important issues in the nanoscale MOSFET technology [1]. These mechanisms lead to device aging, resulting in performance degradation and eventually design failures during the expected system lifetime [2]-[4]. Recently, the oxide thickness of less than 2nm is common in state-of-the-art technologies, and TDDB becomes one of the key challenges among these reliability mechanisms [5][6].

Moreover, the saturating trend for supply voltage scaling causes a large electric field in the gate oxide, which generates gate tunneling currents. The lifetime of a particular gate oxide thickness is determined by the total amount of charge that flows through the gate oxide by the tunneling current. Therefore, nanometer devices are more prone to oxide breakdown compared to micrometer devices. The oxide breakdown is categorized into hard breakdown (HBD) and soft breakdown (SBD): HBD causes a catastrophic failure of the device and the entire circuits. On the other hand, SBD leads to parametric variations such as energy, delay, and noise margin of a gate although it does not bring about functional failures [5].

Figure 1 shows the variation of leakage path current

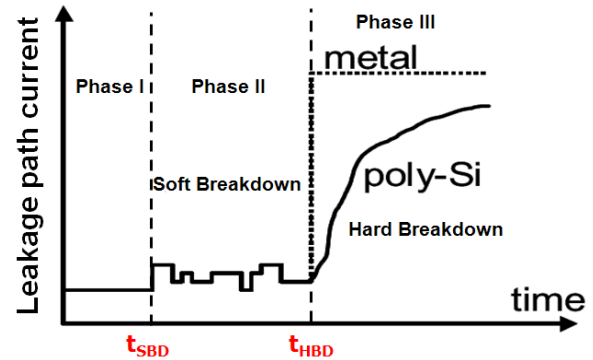


Figure 1. Wear-out and breakdown model for thin gate oxides[6].

depending on the wear-out and breakdown phase for thin gate oxides. At the end of the phase I, the traps start to be increased. During Phase II, moving traps lead to random fluctuation in the leakage current, power and delay, where the device is still functional. In phase III, the conduction path is created in the oxide, the leakage current is exponentially increased, and then, finally, the conduction path causes a catastrophic failure of the device [6].

TDDB is typically treated statistically using the Weibull distribution [7]. For this reason, usually a large gate oxide area has to be used in order to be able to detect the breakdown. Gate oxide breakdown manifests itself as an increase of gate current while retaining its insulating properties (soft break down-SBD or hard break down-HBD) that happens when I-V curve becomes linear, i.e., manifests a resistor-like behavior. For thinner gate oxides ($\leq 30 \times 10^{-10}$ m), SBD is the most likely event. In this paper, TDDB effect only in NMOS devices is considered because the probability of PMOS gate-oxide breakdown is at least an order less than that of NMOS breakdown [8].

Recently, many researches on the TDDB effect have been proposed, but they have mainly focused on device physics not circuit-level analysis. Although a few researches on the circuit-level analysis of TDDB in nanoscale CMOS circuits have been proposed, they have concentrated on mathematical modeling and performance degradations only in simple digital circuits. In this paper, we propose a new simple reliable CMOS VLSI design methodology using a redundancy system.

The remaining of the paper is organized as follows. Section II introduces the TDDB modeling in nanoscale

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology. (2011-0014255)

CMOS circuits. Section III shows our proposed reliable system, followed by the conclusion in Section IV.

II. TDDB MODELING

At current operating voltages for MOSFETs, the degradation mechanism can be approximated by an exponential function of voltage applied across gate oxide and Arrhenius function of temperature as follows:

$$TDDB = C \cdot \exp\left(\frac{E_a}{kT}\right) \cdot \exp(-\beta V_g) \quad (1)$$

where V_g is the voltage across gate oxide, E_a is the activation Energy, k is the Boltzman's constant, T is the junction temperature, C and β are the technology specific constants.

Gate oxide breakdown is sensitive to:

- Voltage across the gate oxide (V_g): The higher V_g and the higher V_g duty cycles the shorter TDDB. For this reason, worst case condition for gate oxide occurs when the device is operated in DC mode (100% duty cycle).
- Junction temperature
- Gate oxide thickness: The thinner the gate oxide the more difficult it is to get good quality oxides and interfaces.
- Gate oxide area: The bigger the gate oxide area the higher the gate oxide breakdown induced failure rate. SRAMs typically have intensive gate oxide area and can be good vehicles to test gate oxide lifetime.

In this paper, for post-breakdown analysis at the circuit-level, a MOSFET with the oxide breakdown is modeled using two breakdown resistors (R_{BD}) as shown in Figure 2 [8]. The R_{BD} value ranges from $G\Omega$ (no TDDB) to a few hundreds of $K\Omega$ (HBD). The time-dependent gate to source/drain resistor model was experimentally verified in Lombardo et al. [9].

The worst stress for NMOS and PMOS gate oxide happens when NMOS and PMOS are in ON state. For this reason, in calculating product the gate oxide FIT, we make the assumptions that duty cycle of the voltage across gate oxide is 50%. However, when PMOS and NMOS are off, respectively gate/drain and gate/source overlaps are stressed, i.e, gate oxide in overlap region is continuously stressed. The FIT contribution from this stress condition is small due to the relatively smaller gate oxide area in the overlap regions.

Based on time dependent dielectric breakdown data (TDDB), there is a maximum voltage across gate oxide that a given technology can support, for a given gate oxide area and failure rate. The oxide thickness, for a technology, is determined so that this voltage is not exceeded, in a dc sense, for any circuit design. However, when signals are driven, there is a certain amount of overshoot/undershoot in the waveform that can result in accelerated gate oxide wear-out and lower reliability, if appropriate limits are not established.

In Figure 3, it is assumed that the NMOS (MN1) of the 1st inverter is stressed by the gate oxide breakdown; while Input signal is changed from logic "0" to "1", the NMOS of

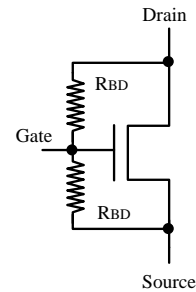


Figure 2. TDDB model for NMOS

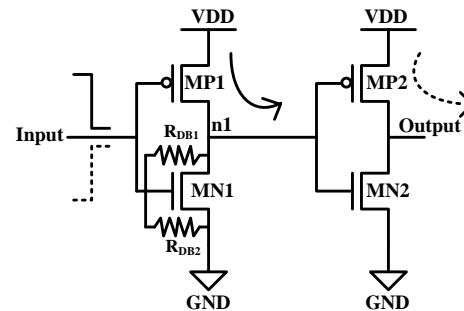


Figure 3. TDDB resistance in NMOSFET

the 1st inverter is turned on; the node n1 capacitor starts to be discharged through MN1, but the discharge time is longer than the normal discharging time because of the breakdown resistor. As a result, this makes the charging time at Output node longer, which means the propagation time from Input to Output will be longer comparing to the normal case without TDDB. The propagation time depends on the breakdown resistor value: if the resistor is large, the charging time at Output node will be a little increased; however, if the resistor is small, the charging time at Output node will be much more increased due to the small voltage-swing at node n1 caused by the small breakdown resistor.

On the other hand, while Input signal is changed from logic "1" to "0", the PMOS of the 1st inverter is turned on; the node1 capacitor starts to be charged through MP1; however, the rising time and rising voltage at node n1 are affected by the breakdown resistor. As a result, this makes the charging time at Output node shorter, which means the propagation time from Input to Output will be shorter comparing to the normal case without TDDB. In this case, if the resistor is large, the discharging time at Output node will be much more decreased because the a small voltage-swing at node n1 (caused by the breakdown resistor) moves up the high-to-low transition at Output node; however, if the resistor is too small, the charging time at Output node will be a little decreased because less voltage-swing (compared to the voltage-swing when the resistor is large) at node n1 makes the current driving force of the 1st inverter weak.

Putting the abovementioned two transitions together, if the breakdown resistor is small, the propagation time from Input to Output will be increased; if the breakdown resistor is

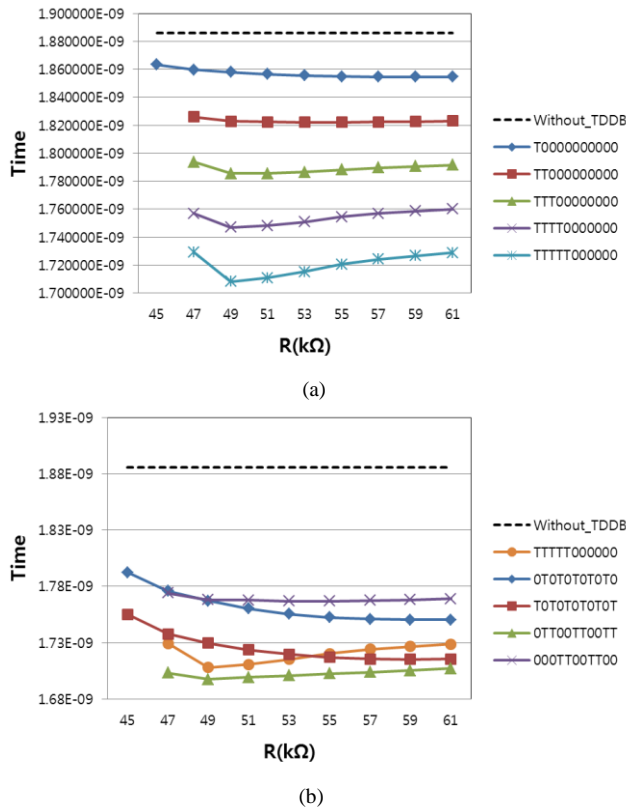


Figure 4. Delay vs. breakdown resistor considering the spatial correlation and the number of inverters with the breakdown resistor, where inverter-chain has 11 inverters; “T” means the inverter with the breakdown resistor; and “0” means the inverter without the breakdown resistor: (a) Delay vs. resistor size vs. the number of inverters with the resistor, (b) Delay vs. resistor size vs. the spatial correlation.

large, the propagation time from Input to Output will be decreased. This means that the propagation time in the case with SBD might be shorter than the time in the normal case without Tddb; and as a gate oxide goes to HDB, the propagation time might be longer than the time in the normal case without Tddb.

In addition, the total delay of the inverter chain depends on the spatial correlation between stressed devices and unstressed devices; that is, the total delay can increase or decrease depending on the location of the stressed device and the number of stressed device [10]. Figure 4 shows the breakdown resistor impact on the Nand chain circuit delay. As expected, as the breakdown resistor size decreases, the delay of the Nand chain becomes decreased around up to 49KΩ, then the delay gets increased and finally the inverter chain goes to a functional failure. Figure 4 (a) shows the effect of the number of Nand gates with the breakdown resistor on the Nand chain delay: as the number of inverters with breakdown resistor increases, the total delay gets shorter. In Figure 4 (b), it is presented that the total delay of the inverter chain depends on the spatial correlation between stressed devices and un-stressed devices. Although the spatial correlation can change the total delay, the total number of the stressed inverter has a great influence on the total delay.

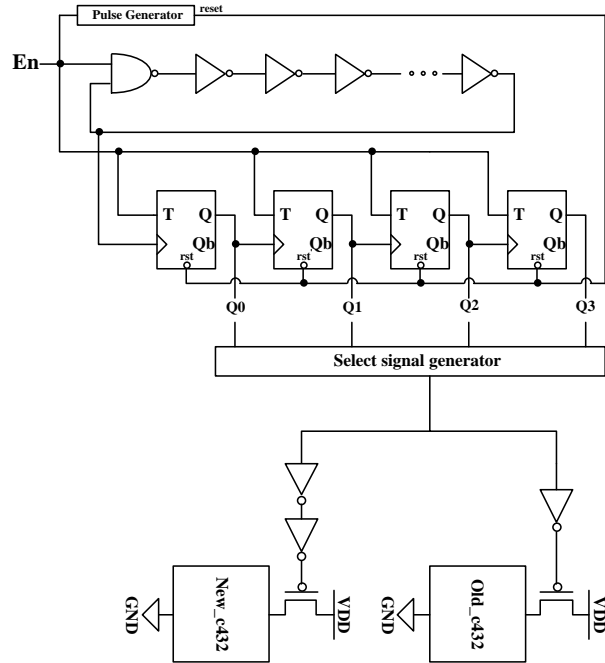


Figure 5. Proposed reliable C432 circuit

III. PROPOSED RELIABLE CMOS VLSI DESIGN

Figure 5 shows a reliable C432 ISCAS benchmark circuit using a redundancy system. The targeting circuit consists of old_c432 and new_c432. The proposed reliable circuit turns off the old_c432 circuit when the test device circuit suffers from HBD and leads to functional failure and turns on the new_c432 circuit with no Tddb according to the indication of the Tddb monitoring circuits consisting of a ring oscillator and T-flip flops. At the beginning point, both of the old and new circuits do not suffer from Tddb effects and “Select” signal output is logic ‘0’. At this time, the PMOS header of the old_c432 circuit is turned on, but the PMOS header of the old_c432 circuit is turned off. The primary inputs and outputs of the c432 circuits are connected to flip-flops that are controlled by the “Select” signal output.

In the proposed circuit, the ring oscillator operates as a replica circuit of a designed digital system. In the stress mode, the external signal En asserts logic ‘0’ in order to turn off the ring oscillator. On the other hand, in the sensing mode, to monitor the Tddb effect on the ring oscillator, the external signal En asserts logic ‘1’ in order to turn on the ring oscillator. During the sensing mode, when a soft breakdown presents at the ring oscillator, the gate to source/drain leakage current gets increased due to the decreasing of the breakdown resistor. Therefore, the counter consisted of four T flip-flops generates different numbers of pulses depending on the total delay of the ring oscillator with the breakdown resistor. When the ring oscillator goes to functional failure owing to the HBD effect, the ring oscillator finally stops oscillating. The proposed HBD sensor circuit detects this moment using an NOR gate. When the ring

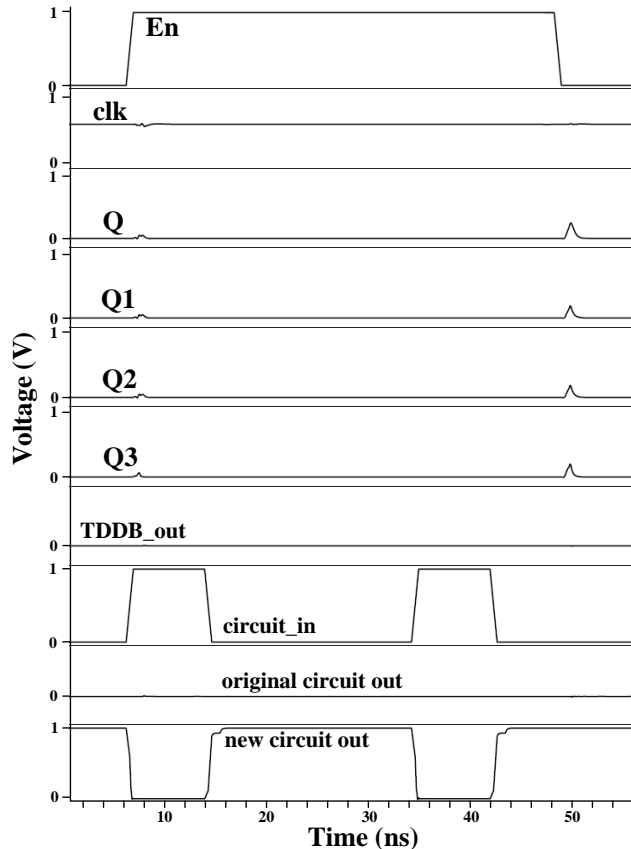


Figure 6. The transition waveform for the proposed monitoring circuits when HBD is generated.

oscillator has HBD resistors on inverters, the “Sensor Output” signal is changed to logic “0”.

When “Select” signal output is logic ‘1’, the primary inputs and outputs are connected to new_c432. At the initial time, the select signal output is logic ‘0’, and the primary inputs and outputs are connected to old_c432. When HBD is generated in the old_c432 circuit, “Select” signal will be changed to logic ‘1’ due to the output of the TDDB monitoring circuit. The PMOS header of the old_c432 circuit will be turned off, and the PMOS header of the new_c432 circuit will be turned on. Therefore, the impact of the HBD on digital circuit can be completely avoided using the proposed design methodology. Figure 6 shows transition waveforms, where HBD in the C432 circuit brings about functional failures. When the circuit leads to functional failures, the binary counter generates ‘0’ outputs, which is different from the circuit with no TDDB or SBD. On the other hands, the monitoring circuit generates meaningful output only when the HBD is generated in the replica circuit (ring oscillator) after long time. When the replica circuit has the HBD effect, the outputs of monitoring circuit (Q0, Q1, Q2, and Q3 are all ‘0’) are asserted to the TDDB signal generator circuit. The TDDB_out signal from the signal generator circuit is changed to logic ‘0’, and then old_c432 circuit is blocked, but the new_c432 circuit will work instead of old_c432.

IV. CONCLUSION AND FUTURE WROK

In this paper, we proposed a reliable CMOS design methodology considering the gate oxide breakdown in the 45nm CMOS technology. The proposed design is based on the TDDB monitoring circuit, the simulation results shows that the impact of TDDB due to severe HBD on the digital circuit can be compensated by the proposed circuit design. Finally, we can extend the life time of the targeting circuit and prevent from performance degradation as well as functional failures due to TDDB. Future efforts aim at applying our proposed monitoring circuit to real systems. Moreover, a reliable system considering all the aging effects (NBTI, PBTI, HCI, and TDDB) will be designed and implemented in the nanoscale technology.

REFERENCES

- [1] R. J. Baker, “CMOS-Circuit design, layout, and simulation,” Wiley and IEEE Press, 2010.
- [2] A. B. Kahng, “Design challenges at 65nm and beyond,” Design, Automation and Test in Europe (DATE) Conference, pp. 1-2, Nice Acropolis, France, March 2007.
- [3] J. W. McPherson, “Reliability challenges for 45nm and beyond,” Proceeding of IEEE Design Automation Conference (DAC), pp. 176-181, San Francisco, USA, July 2006.
- [4] X. Li, J. Qin, and J. B. Bernstein, “Compact modeling of MOSFET wearout mechanisms for circuit-reliability simulation,” IEEE Transactions on Device and Materials Reliability, Vol. 8, Issue 1, pp. 98-121, March 2008.
- [5] M. Choudhury, V. Chandra, K. Mohanram, and R. Aitken, “Analytical model for TDDB-based performance degradation in combinational logic”, Design, Automation & Test in Europe (DATE) Conference, pp. 423-428, , Dresden, Germany, March 2010.
- [6] H. Wang, M. Miranda, F. Cattoor, and D. Wim, “Impact of random soft oxide breakdown on SRAM energy/delay drift,” IEEE Transactions on Device and Materials Reliability, Vol. 7, Issue 4, pp. 581-591, Dec. 2007.
- [7] B. Kaczer, R. Degraeve, M. Rasras, K. Van de Mierop, P. J. Roussel, and G. Groeseneken, “Impact of MOSFET gate oxide breakdown on digital circuit operation and reliability,” IEEE Transactions on Electron Devices, Vol. 49, No. 3, pp. 500-506, 2002.
- [8] R. Rodriguez, R. V. Joshi, J. H. Stathis, and C. T. Chuang, “Oxide breakdown model and its impact on SRAM cell functionality,” International conference on Simulation of Semiconductor Processes and Devices (SISPAD), pp. 283-286, Cambridge, USA, September 2003.
- [9] S. S. Lombardo, J.H. Stathis, B.P. Linder, K.L. Pey, F. Palumbo, and C.H. Tung, “Dielectric breakdown mechanisms in gate oxides,” Journal of Applied Physics, 98(12), pp. 1-36, 2005.
- [10] H. Luo, X. Chen, J. Velamala, Y. Wang, Y. Cao, V. Chandra, Y. Ma, and H. Yang, “Circuit-level delay modeling considering both TDDB and NBTI,” IEEE International Symposium on Quality Electronic Design (ISQED), pp. 14-21, Santa Clara, USA, March 2011.