

## PAPER

# Optimization of Body Biasing for Variable Pipelined Coarse-Grained Reconfigurable Architectures

Takuya KOJIMA<sup>†a)</sup>, Naoki ANDO<sup>†</sup>, *Nonmembers*, Hayate OKUHARA<sup>†</sup>, *Student Member*, Ng. Anh Vu DOAN<sup>†</sup>, *Nonmember, and* Hideharu AMANO<sup>†</sup>, *Fellow*

**SUMMARY** Variable Pipeline Cool Mega Array (VPCMA) is a low power Coarse Grained Reconfigurable Architecture (CGRA) based on the concept of CMA (Cool Mega Array). It provides a pipeline structure in the PE array that can be configured so as to fit target algorithms and required performance. Also, VPCMA uses the Silicon On Thin Buried oxide (SOTB) technology, a type of Fully Depleted Silicon On Insulator (FD-SOI), so it is possible to control its body bias voltage to provide a balance between performance and leakage power. In this paper, we study the optimization of the VPCMA body bias while considering simultaneously its variable pipeline structure. Through evaluations, we can observe that it is possible to achieve an average reduction of energy consumption, for the studied applications, of 17.75% and 10.49% when compared to respectively the zero bias (without body bias control) and the uniform (control of the whole PE array) cases, while respecting performance constraints. Besides, it is observed that, with appropriate body bias control, it is possible to extend the possible performance, hence enabling broader trade-off analyzes between consumption and performance. Considering the dynamic power as well as the static power, more appropriate pipeline structure and body bias voltage can be obtained. In addition, when the control of VDD is integrated, higher performance can be achieved with a steady increase of the power. These promising results show that applying an adequate optimization technique for the body bias control while simultaneously considering pipeline structures can not only enable further power reduction than previous methods, but also allow more trade-off analysis possibilities.

**key words:** CGRA, body bias, power reduction, Cool Mega Array

## 1. Introduction

Recent advanced IoTs (Internet of Things) and wearable computing require a relatively high performance with extremely low energy consumption. CGRA (Coarse-Grained Reconfigurable Architecture) is a candidate of accelerators for such devices thanks to its high degree of performance per limited energy budget. The principle of CGRAs consists of an array of small processing elements (PEs) which can execute simple computational operations, and distributed memory modules connected together with an interconnection network. Highly efficient computing can be performed by changing the type of operations and their interconnection.

VPCMA (Variable Pipeline Cool Mega Array) [1] has been proposed as a low power CGRA based on the concept of CMA (Cool Mega Array) [2]. It provides a large PE array without dynamic reconfiguration and a tiny microcontroller

with banked data memory. The pipeline structure in the PE array can be configured so as to fit target algorithms and required performance. Also, VPCMA uses the Silicon on Thin Buried Oxide (SOTB) technology, a type of Fully Depleted Silicon On Insulator (FDSOI). So a balance between performance and leakage power can be kept by controlling the body bias voltages.

Although the basic trade-off of changing the pipeline structure of VPCMA has been discussed in [1], body bias control has not been applied. Here, we propose a bi-objective optimization method of both energy and performance considering simultaneously the body bias voltages, the pipeline structure, and the target application. At first sight, the problem may seem complex, and one could consider to apply multi-objective metaheuristics such as genetic algorithms to tackle it. However, while these methods have successfully been used for various similar cases, they do not always provide optimal solutions. We propose in this work a model and analysis of this problem that allow to solve it quickly by using an ILP (Integer Linear Program) model, with guarantee of optimality. All optimization results are based on parameters from an existing developed design, and the results can be directly applied to a real chip now under evaluation.

The rest of the paper is organized as follows. Section 2 introduces VPCMA, SOTB process technology, and fundamental body bias control for VPCMA. Then, an optimization method is proposed in Sect. 3 with preliminary evaluation for building an ILP. The optimization results are presented in Sect. 4. After discussion comparing with related work in Sect. 5, we conclude with a brief summary in Sect. 6.

## 2. Variable Pipeline Cool Mega Array (VPCMA)

### 2.1 The Architecture of VPCMA

The VPCMA is classified into Straight Forward CGRAs (SF-CGRAs), a class of simple CGRAs. They consist of a pipelined array of processing elements (PEs), memory modules and networks for transferring data between them. Data are read out from the memory modules, transferred to the input of pipelined array through a permutation network, and the results are written back to the memory modules with another permutation network. The control of data transfer is managed by the code of a microcontroller, while operations

Manuscript received September 25, 2017.

Manuscript revised February 5, 2018.

Manuscript publicized March 9, 2018.

<sup>†</sup>The authors are with Dept. of Information and Computer Science, Keio University, Yokohama-shi, 223-0061 Japan.

a) E-mail: wasmii@am.ics.keio.ac.jp

DOI: 10.1587/transinf.2017EDP7308

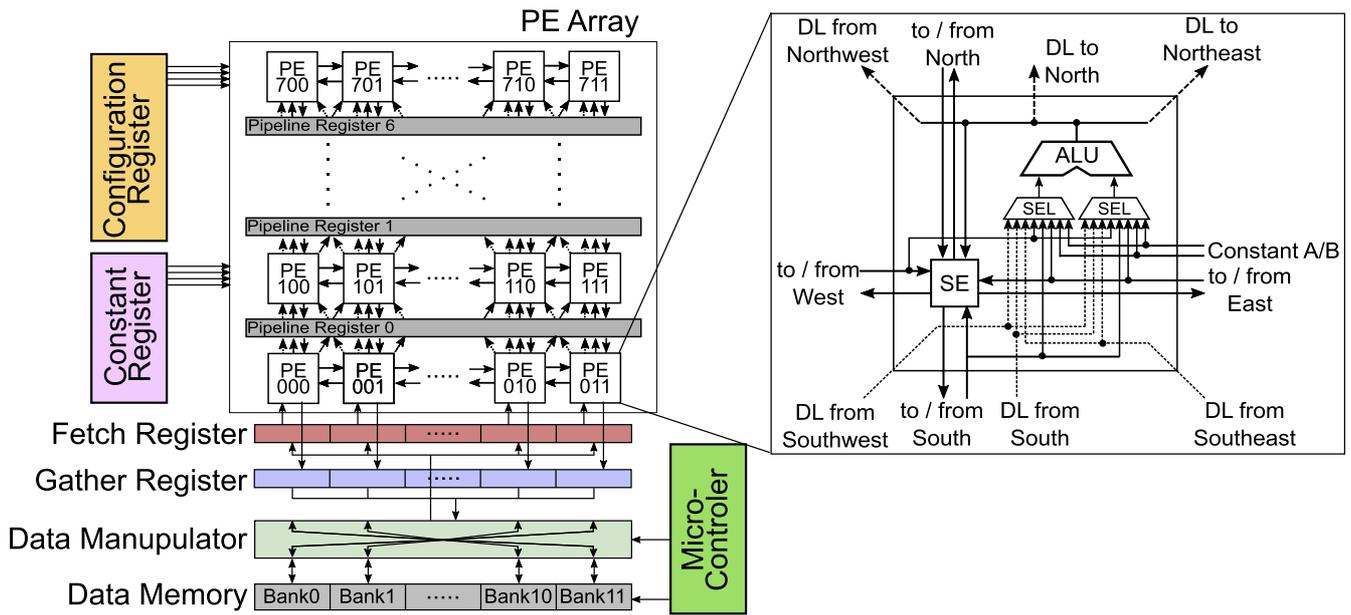


Fig. 1 Diagram of VPCMA

in the pipelined array of PEs are decided by the configuration data that are sometimes switched dynamically. CGRAs such as Pipherench [3], Kilo-core [4] and S5 engine [5] also fall into this classification. Some of these SF-CGRAs can be considered as VLIW (Very Long Instruction Word) computers.

The VPCMA architecture is a simple SF-CGRA that focuses on reducing any energy usage other than that required for computation. The PE array is built with a simple pipelined combinational circuits to eliminate the power needed to distribute a clock to each PE.

As shown in Fig. 1, the VPCMA consists of a large PE array with pipeline registers, a microcontroller and banked data memory. Computation starts when all data are set up in the input “Fetch register” and the outputs of the PE array are stored in the “Gather register” with a certain delay time. The diagram of a PE is also illustrated in Fig. 1. It consists of an arithmetic logic unit (ALU), input selectors, and a switching element (SE). Operations which can be used in the ALU are shown in Table 1 and its bit width is 25-bit whose MSB (Most Significant Bit) is used for ether carry bit or flag bit. The interconnection network between PEs is a mix of direct interconnections and an island-style network. The output of an ALU is directly spread to the input selectors of the north, northeast, and northwest PEs. In Fig. 1, direct links (DL) indicate these paths. Two lanes of the island-style network are vertically and horizontally provided between SEs in each PE. The results computed in the ALU can be directly forwarded to the “Gather register” through paths to the south direction. Once the data are transferred to these paths, they cannot be reused for computation to prevent the creation of combinational loops.

The pipeline registers are placed between every row of the PE array, as shown in Fig. 1. As they are all independently switchable, the VPCMA can freely change its

Table 1 Available operations in ALU

operation	meaning
NOP	no operation
ADD	addition
SUB	subtraction
MULT	multiplication
SL	left shift
SR	logical right shift
SRA	arithmetic right shift
SEL	conditional move
CAT	catenation
NOT	1’s complement
AND	bit-wise AND
OR	bit-wise OR
XOR	bit-wise XOR
EQL	comparing inputs for equality
GT	comparing inputs for greater than
LT	comparing inputs for less than

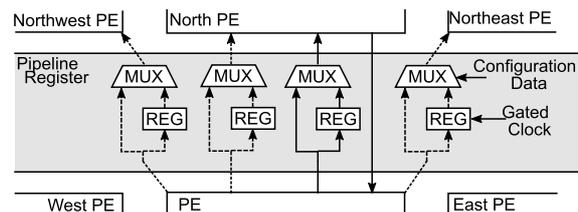


Fig. 2 Detail of a Pipeline Registers in VPCMA

pipeline structure. Figure 2 indicates how pipeline registers are implemented.

Multiplexers can choose whether the signals are from the registers or the bypasses according to configuration data. The micro-controller in requested bypass mode gates the clocks for the registers to reduce consumed power. The south direction path from the north PE does not have any register because it is used for result data.

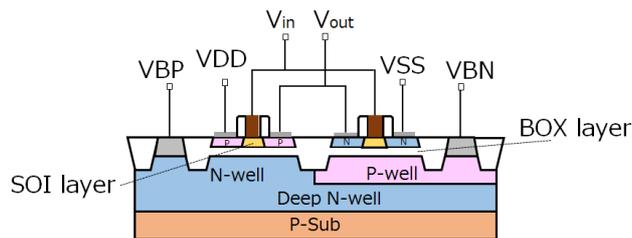


Fig. 3 Cross-sectional view of the SOTB MOSFET

A micro-controller reads the data from the banked data memory (MEM) and distributes them to the fields of the “Fetch register” attached to the input of the PE array according to micro-instructions. It also obtains and places the computation results from the outputs of the PE array in the “Gather register”, and begins writing them back to the data memory. The former and latter operations are called “Fetch” and “Gather” in this paper. “Gather” is reserved on some clocks afterward by dedicated instructions. It flexibly manages multiple data transfer between the banked memory and registers by using data manipulator and vector operations. The mapping between data memory address and Fetch or Gather registers are controlled with a mapping table indicated by a micro-instruction. This structure enables the implementation of various application programs without a power-hungry dynamic reconfiguration in the PE array.

## 2.2 Body Bias Control on SOTB

SOTB is classified as an FD-SOI technology in which transistors are formed on thin buried oxide (BOX) layer. An illustration is provided in Fig. 3. Thanks to its structure, SOTB can operate at a relatively high clock frequency with low supply voltage. Among the numerous benefits of SOTB [6], the delay and leakage power consumption can be widely controlled by the bias voltage supplied to the body (back-gate). Here, we refer to the body-bias voltages of NMOS transistor and PMOS transistor as  $VBN$  and  $VBP$ , respectively.  $VBN$  for NMOS transistors is given to p-well. That is, if  $VBN = 0$  V, the transistor works with a normal threshold level. If reverse-bias ( $VBN < 0$  V) is given, the threshold is raised, thus the leakage current is reduced while the delay is stretched. On the contrary, forward-bias ( $VBN > 0$  V) lowers the threshold which enhances the operational speed with an increase of the leakage current. In the case of PMOS transistors,  $VBP$  is given to the n-well; here the transistors are formed on thin BOX layer, as shown in Fig. 3. Therefore, in this case, zero bias means  $VBP = VDD$  (i.e.  $|VBN| = |VDD - VBP|$ ). When  $VBP > VDD$ , this corresponds to reverse-bias, while  $VBP < VDD$  is for forward-bias. Let us note that the consumption power needed for the body bias control itself is quite small. Indeed, one can cite the body bias generators proposed in [7], [8] which have been tested on a real chip and imply a small overhead.

Here, the bias voltage is equally given to both the NMOS and PMOS, so that  $VBP + VBN = VDD$  is satis-

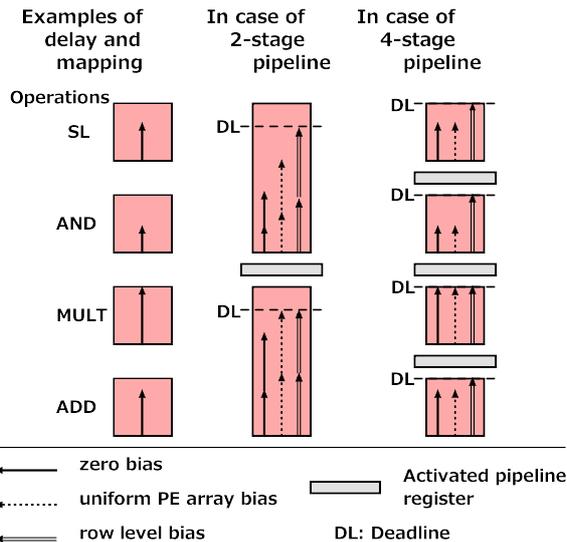


Fig. 4 Row-level body bias control with pipeline registers (2 and 4 stages)

fied. Therefore, we can express the level of body bias solely with the value of  $VBN$ .

## 2.3 Row-Level Body Bias Control for VPCMA

Since the data transfer with the microcontroller and the computation in the PE array are executed in an overlapped manner, their performance should be balanced. In order to keep the balance, body bias control has been used in CMA-SOTB [9]. In that case, the body bias domain is separated into the microcontroller domain including data memory, and the PE array. This can thus be considered as a uniform bias on the whole PE array. However, in the original paper on VPCMA [1], they did not consider body bias control and only used zero bias to study the benefit of a pipeline structure.

In this work, we propose a row-level body bias domain for the PE array, as shown in Fig. 4, to balance the delay time of each pipeline stage. This can allow more flexible choices on the bias voltages compared to a uniform bias where the voltage required to avoid a bottleneck is supplied to all the PE rows, and therefore does not allow them to have reverse bias to balance the pipeline stage delay time, as illustrated in Fig. 4. With a row-level design, each row is implemented with its own body bias domain and receives its own bias voltage. Since all the pipeline registers are outside of the PE array domain and implemented in the same domain as the microcontroller, they can work at the same clock frequency. The delay scaling problem in the clock tree does not have to be taken into account, since all the flip-flops and the whole clock tree are implemented in a single body bias domain. From the layout point of view, the overhead of separating the body bias domains is negligible, since with a common layout policy, the same macro corresponding to a single row is used regardless of whether separation of body bias domains is applied or not. For the use of macro cells, isolation

cells and well separation are needed in any case. Therefore, the eventual overhead to consider would come from a generator which can deliver multiple body bias voltages.

By using a row-level body bias control, we can apply a reverse body bias to every stage whose delay is shorter than the largest one until they become (nearly) equal. Conversely, forward body bias can be supplied to stages whose delay is longer than the shortest one. Even if the delay of each stage is not exactly the same, the difference can still be used to reduce the leakage power.

### 3. Proposed Method

Assuming that the pipeline structure and the body bias voltages are controlled simultaneously, there are several possibilities of trade-off as shown in Table 2.

For instance, we can observe that with a larger number of activated pipeline registers, the power consumption induced by glitches is decreased. Glitches are unneeded signal transitions caused by the different delay times between inputs of the PEs. Without pipeline registers, they are propagated to the PEs in the upper rows and will therefore imply an increase of consumption. Using pipeline registers allows to limit this propagation, and thus the induced power consumption, but at the cost of an overhead related to the registers and the associated clock tree.

This example shows that more advanced analyses are required to assess the trade-off possibilities between performance and power consumption which both depends differently on the pipeline registers configuration and the body bias control. Therefore, we propose in this paper to optimize the choices on the body bias control while simultaneously considering the pipeline structure.

#### 3.1 Problem Definition

On basis of the aforementioned trade-off information, we can define the problem as the following bi-objective optimization problem: given an application, how to optimize the power consumption and the performance of the VPCMA with choices on simultaneously the body bias voltages and the pipeline structure.

The equations required to model this problem can be formulated as follows:

**Table 2** Trade-off between performance and power

	Number of pipelined stage	
	large	small
Performance	high	low
Dynamic power of register and clock tree	increases	decreases
Dynamic power of the glitches	decreases	increases
	Body bias voltage	
	forward bias	reverse bias
Performance	low	high
Static power	decreases	increases

$$VBN_i \in \{-2.0, -1.8, \dots, 0.0, 0.2, 0.4\} \quad (1)$$

$$P_{stat} = \sum_{i=0}^7 P_{leak,row}(VBN_i) + P_{leak,reg} + P_{leak,clk} \quad (2)$$

$$\mathbf{preg} = \{preg_0, preg_1, \dots, preg_6\} \quad (3)$$

$$preg_k = \begin{cases} 1 & \text{if the } k\text{-th} \\ & \text{pipeline register is used} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$$P_{dyn} = f_{req} \times (E_{comb}(\mathbf{preg}) + \sum_{k=0}^6 (E_{reg} + E_{clk})preg_k) \quad (5)$$

$$D_l = \sum_{\substack{\text{PEs in} \\ l\text{-th datapath}}} D_{PE}(VBN) \quad (6)$$

where:

- $VBN_i$  is the body bias voltage supplied to  $i$ -th PE row
- $P_{leak,row}(VBN)$ ,  $P_{leak,reg}$ , and  $P_{leak,clk}$  are the leak power of respectively a PE row on  $VBN$ , a pipeline register, and the clock tree
- $preg_k$  represents the configuration of the  $k$ -th pipeline register, with  $k = \{0, 1, \dots, 6\}$  since the VPCMA implements 7 registers
- $\mathbf{preg}$  is a vector whose elements are  $preg_k$  and expresses the pipeline structure of the PE array
- $P_{dyn}$  and  $P_{stat}$  are respectively the dynamic and static power of the PE array (considering body bias control and pipeline structure)
- $E_{comb}(\mathbf{preg})$ ,  $E_{reg}$ , and  $E_{clk}$  are the energy consumption of respectively the combinational circuits, a pipeline register, and clock tree
- $D_l$  and  $D_{PE}(VBN)$  are the delay time of respectively the  $l$ -th datapath and a PE supplied with  $VBN$ ;  $D_l$  is therefore calculated as the sum of the delays caused by the PEs located in the  $l$ -th datapath.

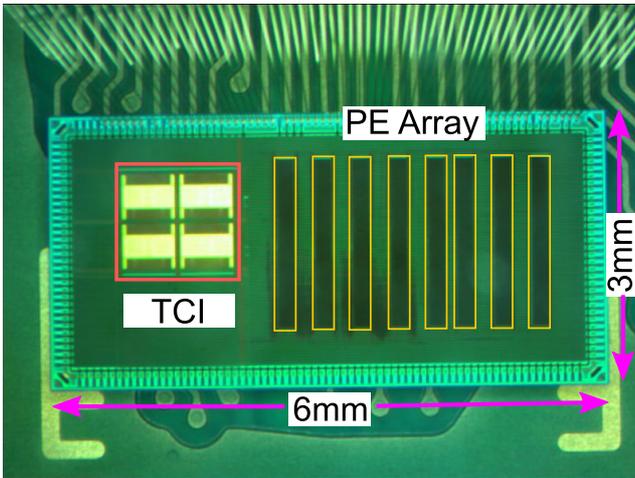
Note that  $E_{comb}$  depends on the pipeline structure (i.e.  $\mathbf{preg}$ ) because of the glitch propagation effect. In this work, the optimization problem is to minimize the sum of  $P_{dyn}$  and  $P_{stat}$ .

#### 3.2 Preliminary Evaluations

The parameters in the above model such as  $P_{leak,row}$  or  $P_{comb}(\mathbf{preg})$  are obtained by several simulations. The used environments are shown in Table 3. The design used in the simulations is based on a real VPCMA chip shown in Fig. 5. It is a 6mm × 3mm chip with Renesas 65nm SOTB process and designed with the same environments as shown in the table. It provides a wireless inductive coupling with through chip interface (TCI) on the left side. As the TCI technology falls out of the scope of this paper, we refer the interested reader to [10] for further information. The eight rectangles aligned in the right side correspond to the rows of the PE array in the VPCMA. Each row provides its own body

**Table 3** Simulation environments for preliminary evaluation

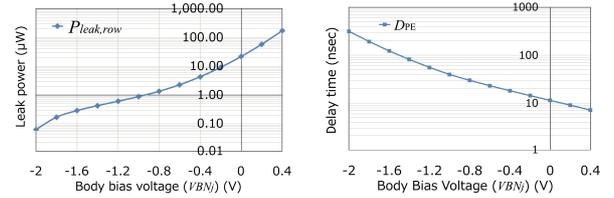
Design	Verilog HDL
Process	Renesas 65 nm
Library name	LPT-8
Synthesis	synopsys Design Compiler 2016.03-SP4
Place and route	synopsys IC Compiler 2016.03-SP4
Temperature	25 °C
Delay and leak power simulation	Synopsys HSIM 2012.06-SP2 $VDD$ : 0.55, 0.65, 0.75, 0.85 V
Dynamic power simulation	Synopsys Prime Time 2012.06-SP2 $VDD$ : 0.55 V
Behavioral simulation	Cadence NC-Verilog 10.20-s131

**Fig. 5** Photo of the VPCMA chip

bias domain. The microcontroller, registers for the banked data memory, Fetch/Gather registers and pipeline registers are distributed within the space not used by the rows and TCI.

$P_{leak,row}$  and  $D_{PE}$  are simulated for each value of  $VDD$  (0.55, 0.65, 0.75 and 0.85 V) with HSIM, by changing the body bias voltages ( $VBN$ ) every 0.2 V from  $-2.0$  V to  $0.4$  V. There are therefore 13 different values.  $P_{leak,row}$  are calculated as an average of two input patterns, all inputs being set either to low level or high level. The simulated  $PE_{leak,row}$  for each  $VBN$  with  $VDD = 0.55$  V is shown in Fig. 6 (a).  $D_{PE,j}$  depend on an operation assigned to the ALU in a PE and its inputs. Hence, the input values which can provide the critical path for each operation are determined using the reports from IC compiler. The  $D_{PE}$  for each  $VBN$  with ADD operation and  $VDD = 0.55$  V is shown in Fig. 6 (b). Both results clearly demonstrate the trade-off between the performance and static power described in Table 2.

$E_{comb}$ ,  $E_{reg}$  and  $E_{clk}$  depend on the running application. Five applications are simulated, as shown in Table 4. For each of them, the dynamic power (at combinational circuit, registers and clock tree) are simulated at a certain frequency and a  $VDD$  of 0.55 V using PrimeTime, and  $E_{comb}$ ,  $E_{reg}$

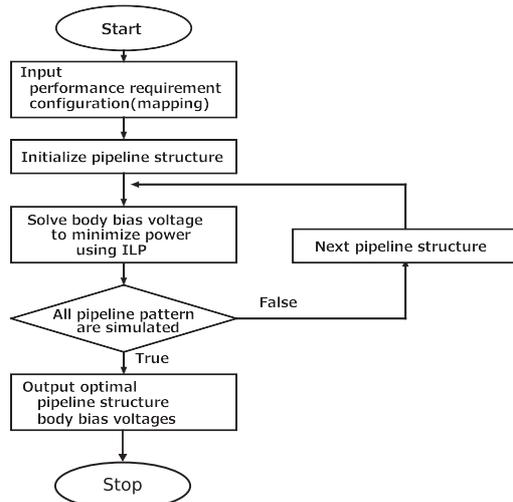
**(a)** Leak power per PE row **(b)** Delay time of PE (ADD operation,  $VDD=0.55$  V)**Fig. 6** Examples of simulation results**Table 4** Simulated applications

Application	Description
gray	24 bit (RGB) gray scale
sepia	8 bit sepia filter
af	24 bit (RGB) alpha blender
sf	24 bit (RGB) sepia filter
dct	8-point DCT

and  $E_{clk}$  are obtained dividing each dynamic power by the frequency. The values for other  $VDD$  voltages used in our simulation (0.65, 0.75 or 0.85 V) are scaled from those of 0.55 V.

An analysis of the solution space shows that its size is  $2^7 \times 13^8$ . Indeed, the VPCMA can configure  $2^7 = 128$  patterns of pipeline structure since for each of the seven registers, it is possible to choose to use it or not. For the row level body bias, given that each of the eight rows in the PE array can select among thirteen possible voltages, there are  $13^8$  possibilities. As a test, for one pipeline structure, it takes 3 hours to elicit and simulate all these possibilities on a 1.6GHz dual-core Intel Core i5 with 8GB of DDR3 RAM.

Given the size of the solution space and the complex formulation of some equations (e.g.  $P_{dyn}$ ), techniques such as metaheuristics could be applied, since they have been used successfully for similar cases, providing interesting solutions in an acceptable amount of time. However, a close examination of the problem shows that it is possible to formulate this problem as an 0-1 ILP (0-1 Integer Linear Problem) (hereinafter, referred to as “ILP”) which, unlike metaheuristics, gives a guarantee of optimality. Indeed, when the pipeline structure is fixed, that is,  $preg_i$  is fixed,  $P_{dyn}$  is constant. Therefore, with the remaining equations being linear, it is possible to formulate this problem as only 128 ILPs (one for each pipeline structure). Moreover, its bi-objective nature can be simplified by considering the performance as a constraint that needs to be reached. Since the design focus of the VPCMA is low power, the problem can be re-formulated as follows: given an application and a fixed pipeline structure, how to optimize the power consumption of the VPCMA while reaching required performance with choices on the body bias voltages. This methodology is then repeated for each pipeline structure, as summarized in Fig. 7. The leak power is minimized by an ILP, while the dynamic power is optimized by the ILP iterations.


**Fig. 7** Algorithm flow-chart

### 3.3 ILP Model

The ILP can then be formulated as follows:

$$isVBN_{ij} = \begin{cases} 1 & \text{if the } i\text{-th PE row} \\ & \text{is set with } j\text{-th } VBN \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

$$\min P_{stat,rows} = \sum_{i=0}^7 \sum_{j=0}^{12} P_{leak,row,j} isVBN_{ij} \quad (8)$$

subject to

$$\sum_{i=0}^7 isVBN_{ij} = 1 \quad \forall j = \{0, 1, \dots, 12\} \quad (9)$$

$$D_l = \sum_{\text{PEs in } l\text{-th datapath}} \sum_{j=0}^{12} D_{PE,j} isVBN_{ij} \quad (10)$$

$$D_l \leq D_{req}, \quad \forall \text{datapath } l \quad (11)$$

$$isVBN_{ij} = \{0, 1\}, \quad \forall i = \{0, 1, \dots, 7\}, \quad \forall j = \{0, 1, \dots, 12\} \quad (12)$$

where  $P_{leak,row,j}$  and  $D_{PE,j}$  are the leak power of a row and the delay time of a PE on  $j$ -th  $VBN$ , the constraint (9) ensures that the row level body bias is respected (same body bias for the PEs on the same row) and (11) expresses that the required performance  $D_{req}$  is reached. It is worth noting that  $P_{leak,reg}$  and  $P_{leak,clk}$  are constant (not controlled by body bias) and therefore do not have to be included in the objective function.

## 4. Evaluation

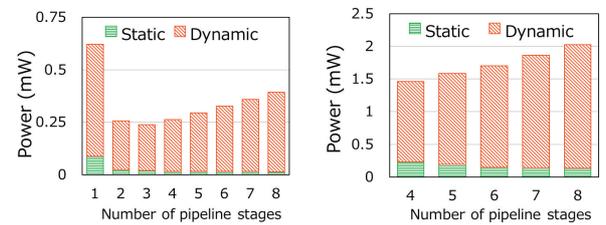
### 4.1 Optimization Results

To analyze the possibilities of the proposed method, we perform the power optimization for several different performance requirements and for each application described in

**Table 5** Examples of optimization results

Requirement of $7.8 \times 10^8$ ops/s								
Pipeline register $i$	0	1	2	3	4	5	6	
$preg_i$	0	0	1	1	0	0	0	
Row number	0	1	2	3	4	5	6	7
$VBN$	-0.8	-0.8	-1.0	-0.6	-1.2	-1.2	-1.4	-1.4

Requirement of $3.9 \times 10^9$ ops/s								
Pipeline register $i$	0	1	2	3	4	5	6	
$preg_i$	0	0	1	1	1	0	0	
Row number	0	1	2	3	4	5	6	7
$VBN$	0.0	0.0	-0.2	0.2	0.2	0.0	0.0	-0.2


 (a) Req. of  $7.8 \times 10^8$  ops/s (b) Req. of  $3.9 \times 10^9$  ops/s

**Fig. 8** Optimal power for each pipeline stages

Sect. 3. Two examples of results are presented in Table 5 and Fig. 8 where the performance is described as the number of executed operations per second, the simulated application is “gray” and  $VDD$  is set at 0.55 V. These results clearly demonstrate that the optimal pipeline structure and body bias voltages are different depending on the requirement and with the proposed method it is possible to solve them exactly.

In case of  $7.8 \times 10^8$  operations/sec, the result of  $preg_i$  indicates that the number of optimal pipeline is three. As shown in Fig. 8 (a), when low performance such as  $7.8 \times 10^8$  operations/sec is requested, static power is extremely low due to strong reverse bias including  $-1.0$  V and the dynamic power accounts for most of the consumption. The single stage structure implies a large dynamic power because of the glitch propagation, as explained in Sect. 3.

On the contrary, when high performance such as  $3.9 \times 10^9$  operations/sec is requested (Fig. 8 (b)), static power is not as small as in the low performance case. Indeed, in order to achieve the requirement, the third and fourth PE row are given forward bias such as 0.2 V which causes an increase of static power. In Fig. 8 (b), the power when the number of pipeline is 1, 2, and 3, is not shown since such low stage pipeline cannot satisfy the performance requirement even if forward bias is applied.

### 4.2 Performance and Energy Reduction

To evaluate the energy reduction achieved by the proposed method, we simulate other policies of body bias control as comparison basis:

- control for the whole PE array (uniform)

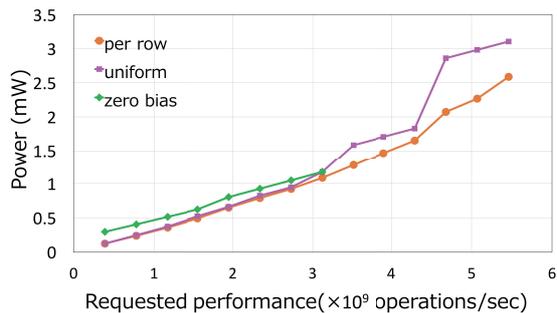


Fig. 9 Comparisons between each method ( $VDD = 0.55$  V)

Table 6 Optimized  $VBN_i$  in case of  $5.46 \times 10^9$  operations/sec (gray)

Uniform control								
$i$	0	1	2	3	4	5	6	7
$VBN_i$ (V)	0.4							
Row-level control								
$i$	0	1	2	3	4	5	6	7
$VBN_i$ (V)	0.2	0.4	0.0	0.4	0.4	-0.2	0.0	-0.8

- no body bias control (zero bias)

As shown in Fig. 9, using the body bias control allows to reach higher achievable performance thanks to applying forward bias. For instance, without body bias control (zero bias), the performance cannot exceed  $3.12 \times 10^9$  operations/sec. However, both the uniform control and the proposed method allow higher performance values.

Furthermore, unlike the uniform control, the proposed method can keep a steady increase of the power even at high performance. This can be explained by the need to apply forward bias to the whole PE array to meet the requirement in the uniform case, which results in a drastic power increase. On the contrary, with the proposed method, forward bias has to be applied only to the row which causes a bottleneck in the critical path. In Table 6, optimized values of  $VBN$  with both controls at the highest performance point are shown. By using values of leak power which are shown in Fig. 6 (a), the leak power with the proposed method is calculated to be 0.6270 mW, while in the uniform case, the leak power is 1.373 mW. However, the dynamic power of the proposed method and uniform control are 1.951 mW and 1.724 mW, respectively. Therefore, the power reduction by the proposed method reaches about 500  $\mu$ W.

To compare the energy between different methods, the average energy of all performances is calculated for each application and for each method. Figure 10 illustrates the reduction ratio of the energy between the proposed method and the other two policies. With the proposed method, it is possible to achieve an energy consumption of 24.5% and 16.1% lower than respectively the zero bias and the uniform cases (the best reduction with “gray” application). In average, the consumption is 17.75% and 10.49% lower than respectively the zero bias and the uniform cases.

To discuss the effectiveness of body bias control using

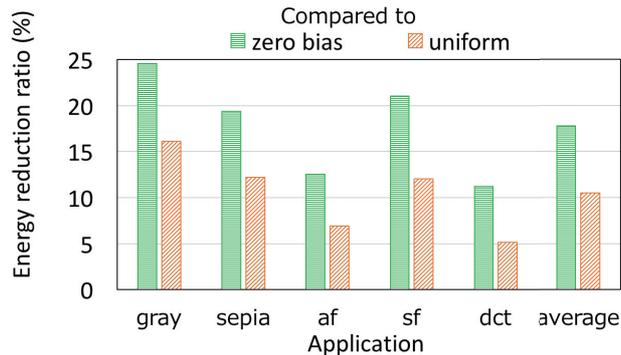


Fig. 10 Energy reduction ratio for each application ( $VDD = 0.55$  V)



Fig. 11 Static power reduction ratio compared to the uniform and the zero bias cases ( $VDD = 0.55$  V)

the proposed method, we also focus on the static power. Figure 11 shows the reduction ratio of the static power where the “gray” application is simulated. The proposed method results in a reduction of 91.9% and 65.8% when compared with respectively the zero bias and uniform cases. Figure 11 also reveals that the static power is not always lower than the uniform case. For example, when  $1.56 \times 10^9$  operations/sec is requested, the static power of the proposed method is higher by 90.9%. Nevertheless, the total power with the proposed method is always lower. At such performance, as an increase of the static power occurs, a change in the optimal pipeline structure is also observed. If the pipeline structure or the body bias control are considered independently, it is then impossible to adjust the balance between the static and the dynamic powers.

### 4.3 Comparison of $VDD$ Control

When the focus is on high performance, there are two ways to achieve it: using forward bias or increasing  $VDD$ . Figure 12 shows that, with the proposed method, using forward bias is better whereas, with uniform control, a higher  $VDD$  is more suitable in some cases. In the uniform policy, we can observe ranges of frequencies where a higher  $VDD$  implies a lower total power. Even if a higher  $VDD$  will increase both static and dynamic powers, it will also improve the performance so reverse bias can be used to decrease the

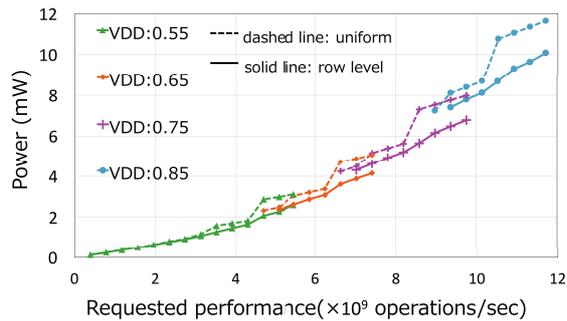


Fig. 12 Optimization result considering VDD

total power. Furthermore, in the range such as  $[4.29 \times 10^9, 5.46 \times 10^9]$ , lower  $VDD$  has to use forward bias to satisfy the performance requirement, which causes a large increase of the leak power. For instance, between  $4.29 \times 10^9$  operations/sec and  $4.68 \times 10^9$  operations/sec, it is more interesting to supply  $VDD$  with 0.65 V rather than 0.55 V. On the contrary, a similar phenomenon is not observed with the proposed method, which suggests to use a  $VDD$  as low as possible depending on the requirement.

Finally, in terms of algorithmic performance, it is worth noting that the proposed method gives a guarantee of optimality and is indeed faster than an explicit elicitation. Compared to the previously-mentioned 3 hours to simulate all the possibilities for a fixed pipeline structure, the ILP takes around 4 minutes in the worst simulated case.

## 5. Related Work

Variable pipeline structure is widely used to select various trade-off between the performance and power. It was applied to a CPU [11], H.264 decoder [12] and routers [13], [14]. Some of them control the power supply voltage when the pipeline structure is changed but a body bias control has not been applied.

Variable body bias control technique has been applied to a dynamically reconfigurable processor [15] and the CMA [16]. However, the former focuses on finding the optimal body bias domain size at the design stage whereas the latter is also searching for the optimal size of body biasing, but targeting instead groups of PE array with combinational circuits, and a genetic algorithm which cannot give guarantee of optimality was used. They did not consider the pipeline processing and so the optimization only focused on body biasing. Although our paper from earlier research stage [17] briefly proposes the concept of the proposal here, it does not include the extensive evaluation results and optimization considering VDD. Since the goal of this paper is multi-objective optimization of both the power and the performance considering simultaneously body bias control and pipeline structure, the optimization methods and results are completely different.

## 6. Conclusion

In this paper, we have proposed a methodology to optimize simultaneously the power consumption and the performance of a variable pipelined CGRA, the VPCMA, while considering both body biasing and pipeline structure. It has been shown that even if the problem seems complex at first sight (solution space of  $2^7 \times 13^8$ ), a proper model and a close analysis allowed to simplify it into ILPs. This enabled fast optimization with guarantee of optimality. The simulation results demonstrated that the proposed method allows to reach low consumption while meeting required performance. Indeed, compared to previous works, we always obtain lower power consumption. We observed that it is possible to achieve an average reduction of energy consumption, for the studied applications, of 17.75% and 10.49% when compared to respectively the zero bias (without body bias control) and the uniform (control of the whole PE array) cases, while respecting performance constraints. Moreover, the range of possible performance can be stretched with appropriate body biasing and pipeline structure, hence enabling broader trade-off analyzes between consumption and performance. In addition, when the control of VDD is integrated, higher performance can be achieved with a steady increase of the power. These promising results show that applying an adequate optimization technique for the body bias control while simultaneously considering pipeline structures can not only enable further power reduction than previous methods, but also allow more trade-off analysis possibilities.

As future works, although all the parameters used for the simulations are based on an existing developed design, tests on a real chip (now under evaluation) have yet to be carried out. Besides, while the obtained results are promising for the development of CGRAs implementing the SOTB technology, it is worth noting that the optimization is currently performed considering a fixed application mapping on the PEs. Since the body bias control and the pipeline structure both depend on the mapping, a change on the latter (for instance, a more compact mapping) may alter the optimality of previously-found bias voltages and pipeline registers configuration. An application mapping tool considering both body bias control and pipeline structure would allow even further optimization and analyzes.

## Acknowledgments

This work is supported by VLSI Design and Education Centr (VDEC), the University of Tokyo in collaboration with Synopsys, Inc. and Cadence Design Systems, Inc.

## References

- [1] N. Ando, K. Masuyama, H. Okuhara, and H. Amano, "Variable pipeline structure for coarse grained reconfigurable array cma," 2016 International Conference on Field-Programmable Technology, pp.231–238, 2016.
- [2] N. Ozaki, Y. Yoshihiro, Y. Saito, D. Ikebuchi, M. Kimura, H.

Amano, H. Nakamura, K. Usami, M. Namiki, and M. Kondo, "Cool mega-array: A highly energy efficient reconfigurable accelerator," Field-Programmable Technology (FPT), 2011 International Conference on, pp.1–8, IEEE, 2011.

- [3] H. Schmit, D. Whelihan, A. Tsai, M. Moe, B. Levine, and R.R. Taylor, "Piperench: A virtualized programmable datapath in 0.18 micron technology," Custom Integrated Circuits Conference, 2002. Proceedings of the IEEE 2002, pp.63–66, IEEE, 2002.
- [4] B. Levine, "Kilocore: Scalable, High Performance and Power Efficient Coarse Grained Reconfigurable Fabrics," Proc. International Symposium on Advanced Reconfigurable Systems, pp.129–158, 2005.
- [5] J.M. Arnold, "S5: the architecture and development flow of a software configurable processor," Proceedings 2005 IEEE International Conference on Field-Programmable Technology, 2005, pp.121–128, IEEE, 2005.
- [6] Y. Morita, R. Tsuchiya, T. Ishigaki, N. Sugii, T. Iwamatsu, T. Ipposhi, H. Oda, Y. Inoue, K. Torii, and S. Kimura, "Smallest Vth variability achieved by intrinsic silicon on thin BOX (SOTB) CMOS with single metal gate," 2008 Symposium on VLSI Technology, pp.166–167, June 2008.
- [7] H. Nagatomi, N. Sugii, S. Kamohara, and K. Ishibashi, "A 361nA Thermal Run-away Immune VBB Generator using Dynamic Substrate Controlled Charge Pump for Ultra Low Sleep Current Logic on 65nm SOTB," Proceedings of the SOI-3D-Subthreshold Microelectronics Technology Unified Conference, pp.1–2, Oct. 2014.
- [8] M. Blagojevi, M. Cochet, B. Keller, P. Flatresse, A. Vladimirescu, and B. Nikoli, "A fast, flexible, positive and negative adaptive body-bias generator in 28nm fdsoi," 2016 IEEE Symposium on VLSI Circuits (VLSI-Circuits), pp.1–2, June 2016.
- [9] H. Su, Y. Fujita, and H. Amano, "Body bias control for a coarse grained reconfigurable accelerator implemented with silicon on thin box technology," 2014 24th International Conference on Field Programmable Logic and Applications (FPL), pp.1–6, Sept. 2014.
- [10] Y. Take, H. Matsutani, D. Sasaki, M. Koibuchi, T. Kuroda, and H. Amano, "3D NoC with Inductive-Coupling Links for Building-Block SiPs," IEEE Transactions on Computers (TC), vol.63, no.3, pp.748–763, March 2014.
- [11] T. Shimada, T. Madokoro, H. Oshima, and R. Kobayashi, "A novel low-power processor with variable pipeline control," Proc. IEEE International Symposium on VLSI-DAT, pp.263–266, 2008.
- [12] C. Lee and S. Yang, "Design of an H.264 decoder with variable pipeline and smart bus arbiter," 2010 International SoC Design Conference, pp.432–435, 2010.
- [13] H. Matsutani, Y. Hirata, M. Koibuchi, K. Usami, H. Nakamura, and H. Amano, "A multi-Vdd dynamic variable pipeline on-chip router for CMPs," Proc. ASP-DAC2012, pp.407–412, Jan. 2012.
- [14] C.-Y. Lee and N.K. Jha, "Variable-Pipeline-Stage Router," IEEE Trans. VLSI system, vol.21, no.9, pp.1669–1682, Jan. 2013.
- [15] J.M. Kühn, H. Amano, O. Bringmann, and W. Rosenstiel, "Leveraging FDSOI through Body Bias Domain Partitioning and Bias Search," Proc. 53rd Design Automation Conference, July 2016.
- [16] Y. Matsushita, H. Okuhara, K. Masuyama, Y. Fujita, R. Kawano, and H. Amano, "Body bias grain size exploration for a coarse grained reconfigurable accelerator," 2016 26th International Conference on Field Programmable Logic and Applications (FPL), pp.1–4, Aug. 2016.
- [17] T. Kojima, N. Ando, H. Okuhara, N.A.V. Doan, and H. Amano, "Body bias optimization for variable pipelined cgra," International Conference on Field Programmable Logic and Application, 2017.



**Takuya Kojima** received BS degree from Keio University, Yokohama, Japan, in 2017. He is a master student in Keio university in the presence.



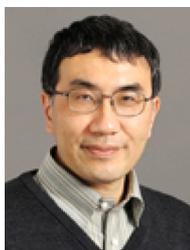
**Naoki Ando** received BS degree from Keio University, Yokohama, Japan, in 2016. He is a master student in Keio university in the presence.



**Hayate Okuhara** received ME degree from Keio University, Yokohama, Japan, in 2016. He is a Ph.D course student in Keio university in the presence.



**Ng. Anh Vu Doan** received his MsC degree in Electrical Engineering (Electronics options) from the Université libre de Bruxelles (ULB), Belgium, in 2009. He then received his PhD degree in Engineering Science and Technology from the ULB in 2015.



**Hideharu Amano** received Ph.D degree from the Department of Electronic Engineering, Keio University, Japan in 1986. He is currently a professor in the Department of Information and Computer Science, Keio University. His research interests include the area of parallel architectures and reconfigurable systems.