

# Performance Evaluation of Various Configuration of Adder in Variable Latency Circuits with Error Detection/Correction Mechanism

Kenta Ando

Atsushi Takahashi

Graduate School of Engineering, Osaka University  
 Suita, 565-0871, Japan  
 {ando,atsushi}@si.eei.eng.osaka-u.ac.jp

**Abstract**— In this paper, in order to develop a circuit synthesis method for error detection/correction framework, various ripple-carry-adders (RCA) in which the minimum delay is increased by delay insertion and/or the probability of large delay is reduced by changing the configuration of the circuit components are designed and evaluated. In experiments, we confirm that a circuit obtained achieves a better performance in error detection/correction framework.

## I. INTRODUCTION

Most of digital circuits nowadays are designed as clock synchronous circuits with global clocks. In a typical clock synchronous circuit implementation, a global clock is designed to be inputted to every flip-flop simultaneously, and every primitive computation is executed in one clock cycle. That is, every signal transfer between flip-flops is done in a clock cycle, and the latency of a primitive computation is fixed. In such implementation, the performance of a circuit depends on the clock period, and the maximum delay of primitive computations gives a lower bound of the clock period. Therefore, the reduction of the maximum delay of primitive computations without changing the amount of primitive computations executed in one clock cycle have been pursued in circuit synthesis to maximize the performance of a circuit.

Various methods to reduce the maximum delay have been proposed. However, the reduction of delay by these methods approaches to the limit. Critical paths of a circuit that require larger delay may be seldom activated in a primitive computation. As the manufacturing technology advances, the variation of delay of a circuit element has become larger as the size of the element has become smaller. The difference of delay of a primitive computation between the best cases and the worst cases has become larger. While, the circuit behavior should be guaranteed in all cases. Disadvantages to guarantee the circuit behavior in all cases have become larger.

Variable latency circuits (VLC) in which each primitive computation is not done in fixed latency have potential to ease these disadvantages and to improve the circuit performance remarkably. In VLC, the number of clock cycles

of each primitive computation changes depending on circuit status etc [1]. There are various implementations of VLC. In this paper, VLC that is implemented by using error detection/correction mechanism [2], called VLEDC, is assumed.

A conventional fixed latency circuit can be converted into VLEDC by lapping it by a circuitry that implements error detection/correction mechanism. In paper [3], several conventional adders are evaluated in VLEDC framework. It was shown that the effective clock periods of them are improved by converting them into VLEDC. However, it is not clear whether the maximum performance of addition in VLEDC framework is achieved by converting conventional adders. The performance of a circuit in VLEDC framework depends on the minimum delay, maximum delay, and delay distribution of the circuit. In general, the performance is better if the larger the minimum delay is and/or the lower the possibility of large delay is. However, conventional circuits are usually designed so that the maximum delay is reduced as much as possible to maximize the performance in the conventional framework and are not necessarily fitted to VLEDC framework.

In this paper, in order to develop a circuit synthesis method for error detection/correction framework, various ripple-carry-adders (RCA) in which the minimum delay is increased by delay insertion and/or the probability of large delay is reduced by changing the configuration of the circuit components are designed and evaluated. In experiments, we confirm that a circuit obtained achieves a better performance in error detection/correction framework.

## II. VARIABLE LATENCY CIRCUITS WITH ERROR DETECTION/CORRECTION MECHANISM

In this section, the behavior of a variable latency circuit with error detection/correction mechanism [2] is explained. The *latency* of a circuit is the time required to generate the outputs after the inputs are given, and is usually the multiple of the clock period. VLEDC changes the latency according to the time required to generate the output signals.

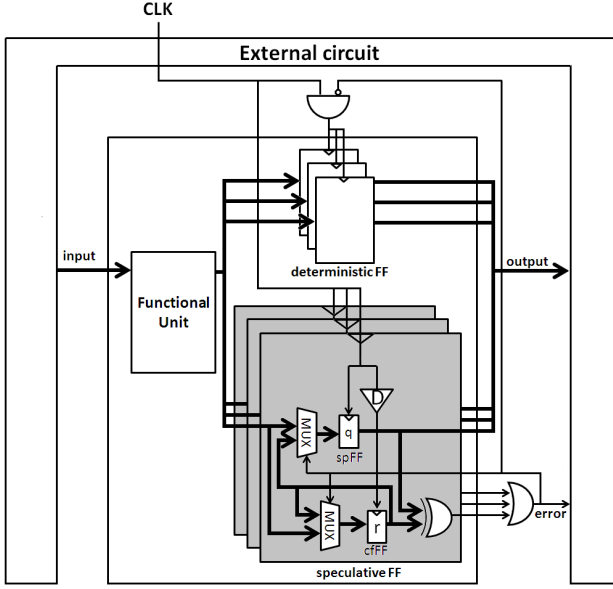


Fig. 1. VLEDC

VLEDC has two execution modes. One is regular mode in which primitive computations are being executed. The other is correction mode in which wrong circuit behavior caused by delay errors is being corrected. VLEDC continues executing primitive computations in regular mode if no delay error occurs. If delay errors occur, VLEDC detects them and corrects the circuit status by replacing wrong values with correct values in correction mode, while suspending the execution of primitive computations. When the circuit status is corrected, VLEDC returns to regular mode and resumes the execution of primitive computations.

The latency of a primitive computation is the time required to execute it in regular mode if no delay occurs. While, the time required in correction mode is added if a delay error occurs.

Fig.1 shows an overview of an implementation of VLEDC where a functional unit is converted into VLEDC. In this implementation, a conventional deterministic flip-flop is replaced by a speculative flip-flop. A speculative flip-flop contains two conventional deterministic flip-flops, called spFF and cFF. A delay error caused at spFF  $q$  is allowed, while no delay error is allowed at cFF  $r$ . The value stored at  $q$  is erroneous but is available earlier. While, the value stored at  $r$  is errorless but are available later. The values stored at  $q$  is used as an “output” signal of this circuit, and the following primitive computations start earlier. The value generated by comparing the values of  $q$  and  $r$  after the correct value is available at  $r$  is used as an “error” signal of this circuit, and the behavior of the following circuits is controlled.

When a primitive computation that takes less than the clock period is executed in this circuit, both  $q$  and  $r$  store the correct value. However, when a primitive computation

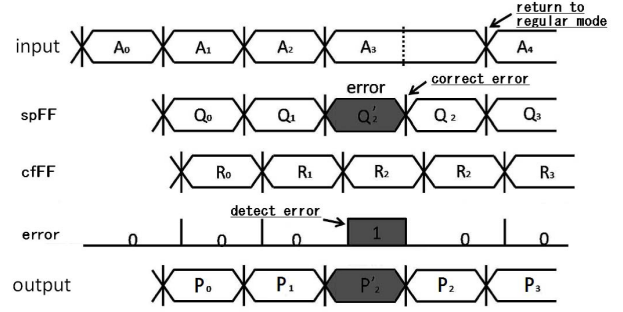


Fig. 2. A timing chart of VLEDC

that takes more than the clock period is executed,  $q$  does not always store the correct value. If two values stored in  $q$  and  $r$  are different, it means that  $q$  does not store the correct value, and that a delay error occurs. When a delay error occurs, it is notified to the following circuit by using “error” signal, and  $q$  is made to store the correct value.

Fig.2 shows the timing chart that indicates a behavior of the circuit.  $A_s$  represents input values, and  $P_s$  represents output values.

### III. TIMING CONSTRAINTS OF CIRCUITS

#### A. Timing constraint

VLEDC contains speculative flip-flops and conventional deterministic flip-flops. Let  $F_s$  and  $F_n$  be the set of speculative flip-flops and deterministic flip-flops in the circuit, respectively. Let  $F = F_s \cup F_n$ .

Let  $T$  be the clock period and  $s(a)$  be the clock timing of flip-flop  $a$ . The clock timing of a speculative flip-flop is defined as the clock timing of spFF in it. The difference of the clock timings of spFF  $q$  and cFF  $r$  in a speculative flip-flop  $p$  is called confirmation margin, and denoted by  $d(p)$ . The confirmation margin is assumed to be non negative. That is,  $s(p) = s(q)$  and  $d(p) = s(r) - s(q) \geq 0$ . In the following discussion, a deterministic flip-flop  $a \in F_n$  is regarded as a speculative flip-flop which consists of spFF and cFF where the confirmation margin is 0.

Let  $P$  be the set of all pairs of flip-flops such that signals are transferred from one to another without through other flip-flops. Let  $d_{\max}(a, b)$  and  $d_{\min}(a, b)$  be the maximum delay and the minimum delay from spFF of speculative flip-flop  $a$  to cFF of speculative flip-flop  $b$  without passing through other flip-flops, respectively. These delays correspond to the signal transfers in regular mode. Similarly, let  $c_{\max}(a, b)$  and  $c_{\min}(a, b)$  be the maximum delay and the minimum delay from cFF of speculative flip-flop  $a$  to spFF or cFF of speculative flip-flop  $b$  without passing through other flip-flops, respectively. These delays correspond to the signal transfers in correction mode.

In order to transfer signal correctly in the regular mode of VLEDC, the following two constraints must be satisfied

by  $(a, b) \in P[8]$ :  
Setup constraint

$$s(a) + d_{\max}(a, b) \leq T + s(b) + d(b)$$

Hold constraint

$$s(b) + d(b) \leq s(a) + d_{\min}(a, b)$$

These two constraints guarantee that the correct signal is stored at cFF of  $b$ . Since a delay error is allowed at spFF, no constraint is imposed on spFF when it is the destination of a signal transfer.

Also, in order to detect delay errors and to correct signal correctly in the correction mode of VLEDC, the following two constraints must be satisfied by  $(a, b) \in P$ :

Setup constraint

$$s(a) + d(a) + c_{\max}(a, b) \leq T + s(b)$$

Hold constraint

$$s(b) \leq s(a) + d(a) + c_{\min}(a, b)$$

Assume that the clock timings of flip-flops in  $F$  are the same and that the confirmation margins of speculative flip-flops in  $F_s$  are the same. Let  $s(a) = 0$  for all  $a \in F$  and  $d(p) = d$  for all  $p \in F_s$  where  $d$  is a non negative constant. Then

$$T \geq \max\{\max\{d_{\max}(a, b) - d \mid a \in F, b \in F_s, (a, b) \in P\}, \max\{d_{\max}(a, b) \mid a \in F, b \in F_n, (a, b) \in P\}, \max\{c_{\max}(a, b) + d \mid a \in F_s, b \in F, (a, b) \in P\}\}$$

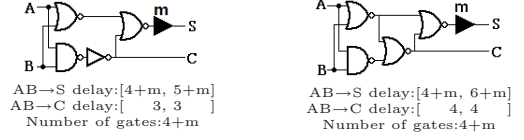
and

$$d \leq \min\{d_{\min}(a, b) \mid a \in F, b \in F_s, (a, b) \in P\}.$$

Roughly speaking, the clock period can be reduced without violating the setup constraint by setting the confirmation margin larger. The confirmation margin can be set larger without violating the hold constraint when the minimum delay is large.

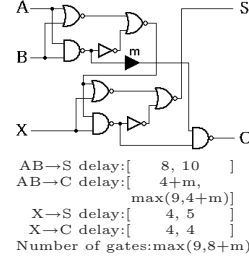
### B. Effective clock period

In a typical fixed latency circuit, one primitive computation is executed in one clock cycle. So the latency of a primitive computation is equal to the clock period and the speed performance of a circuit can be evaluated by the clock period. However, in VLEDC, the number of clock cycles required in one primitive computation changes. In VLEDC, the speed performance of a circuit is evaluated by **effective clock period**  $T_{eff}$ , which is defined as the average latency of primitive computations. When a circuit executes a primitive computation in  $\alpha$  cycles and error correction in  $\beta$  cycles when a delay error occurs, the effective clock period is given by  $T_{eff}(T) = T(\alpha(1-E(T)) + (\alpha+\beta)E(T)) = T(\alpha+\beta E(T))$ ,

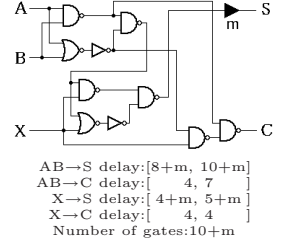


(a)  $H_a(m)$

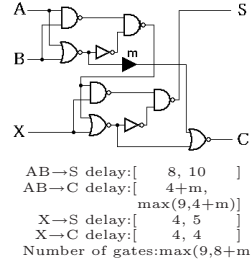
(b)  $H_b(m)$



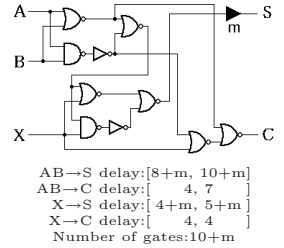
(c)  $F_a(m)$



(d)  $F_b(m)$



(e)  $F_c(m)$



(f)  $F_d(m)$

Fig. 3. Half-adder and full-adder

where  $E(T)$  is the probability of delay error with clock period  $T$ .

Generally speaking, in order to reduce the effective clock period of VLEDC, (1) the minimization of the maximum delay, (2) the maximization of the minimum delay, and (3) the reduction of the probability of larger delay are preferred.

## IV. CONFIGURATION OF ADDERS

In this paper, various 6-bit ripple-carry-adders (RCA) which consist of one half-adder (HA) and five full-adders (FA) are evaluated. In Fig.3, two HAs and four FAs which are used to form RCA are shown.  $H_a$  and  $H_b$  are HA.  $F_a$ ,  $F_b$ ,  $F_c$ , and  $F_d$  are FA. They consists of NAND, NOR, NOT, and BUF gates. BUF consists of the even number of NOT gates in series. BUF gates which are shown by black triangles are inserted to increase delay, if necessary, where its size which is the number of NOT gates contained is represented by  $m$ .  $H_x$  and  $F_x$  with BUF of size  $m$  are denoted by  $H_x(m)$  and  $F_x(m)$ , respectively. Note that  $H_x(0) = H_x$  and  $F_x(0) = F_x$ . In cases that BUF is

TABLE I  
PERFORMANCES OF RCAs USING ONE TYPE OF FULL-ADDER

Name	Configuration		No. of gates	Delay[ns]		Period[ns]		$E(T)$ [%]	$T_{eff}$ [%]	$P$ [%]	$PT_{eff}$ [%]
	Half-adder	Full-adder		max	min	eff.	clock				
$\langle aaaaa \rangle$ (conventional)		$F_a(0)$	49	26	4	22.58	22	2.6	100.0	100.0	100.0
$\langle accccc \rangle$	$H_a(0)$	$F_c(0)$	49	26	4	23.10	22	5.0	102.3	107.0	109.5
$\langle abbbbbb \rangle$		$F_b(0)$	54	24	4	20.89	20	4.4	92.5	111.6	103.2
$\langle addddd \rangle$		$F_d(0)$	54	24	4	21.64	20	8.2	95.8	113.2	108.4
$\langle baaaa \rangle$		$F_a(0)$	49	26	4	22.52	22	2.3	99.7	<b>98.8</b>	98.6
$\langle bccccc \rangle$	$H_b(0)$	$F_c(0)$	49	26	4	23.17	22	5.3	102.6	105.9	108.6
$\langle bbbbbb \rangle$		$F_b(0)$	54	25	4	21.99	21	4.7	97.4	112.7	109.7
$\langle bddddd \rangle$		$F_d(0)$	54	25	4	22.20	21	5.7	98.3	114.3	112.4

inserted to  $F_a$  or  $F_c$ , NOT gate contained is used as the head NOT gate of BUF, and the increase of the number of NOT gates by insertion is reduced by one.

The basic structural comparisons of HAs are as follows. Output C is generated by NOT gate in  $H_a$ . While, output C is generated by NAND gate in  $H_b$ . The numbers of gates of  $H_a$  and  $H_b$  are the same. The maximum and minimum number of gates from an input to an output of HA are the same in both of them.

The basic structural comparisons of FAs are as follows. The number of gates of  $F_a$  is one smaller than  $F_b$ . The maximum number of gates from input A or B to output C of  $F_b(0)$  is one smaller than  $F_a(0)$ .  $F_c$  and  $F_d$  are obtained from  $F_a$  and  $F_b$  by replacing NAND gates with NOR gates and vice versa, respectively.

## V. DELAY AND POWER MODEL

In our delay analysis, the following simple model is used. The gate delay of NAND, NOR, and NOT gates are set to 2[ns], 2[ns], and 1[ns], respectively. No other delay is assumed. The maximum and minimum delays from each input to each output of HAs and FAs are described in Fig.3. For example, the maximum delays to output S and C of  $H_a$  are 1 [ns] shorter than that of  $H_b$ . The maximum delays from input A or B to output C of  $F_b(0)$  are 2 [ns] shorter than that of  $F_a(0)$ .

In our power analysis, we assume that each gate consumes the same constant power during it works, and that the power consumption of a circuit is the sum of powers consumed by gates in the circuit. The energy consumption of a circuit to execute a primitive computation is the total power consumed by the circuit from the inputs are given to the outputs are generated. A change of an input of a gate influences the output of the gate after the gate delay. The gate consumes power during the gate delay if the output of the gate is changed according to the change of the input.

The delay and energy consumption of a circuit to execute a primitive computation varies according to an input vector pairs. The distributions of delay and energy consumption of a circuit to execute a primitive computation depend on the characteristics of the set of input

vector pairs. In this paper, the distributions of delay and energy consumption of a circuit are obtained by assuming that the probability of occurrence of each input vector pair is equal. The energy consumed by error detection/correction circuit is ignored.

## VI. SIMULATION

In our evaluation, each RCA is written by verilog language and is simulated by using VCS. Each RCA is evaluated by the minimum effective clock period  $T_{eff}$ , the average energy consumption  $P$ , and the product of these  $PT_{eff}$ .

Each configuration of 6-bit RCA shown in this paper is represented by the sequence of types of HA and FA and the amount of delay insertion. For example,  $\langle abaaaa \rangle(2)$  consists of  $H_a$  (bit-1),  $F_b$  (bit-2), and  $F_a$  (from bit-3 to bit-6), and the size of each BUF is determined so that the minimum delay is increased by 2 [ns] without increasing the maximum delay.

The distributions of delay and energy consumption of a circuit are obtained by simulating all input vector pairs. The probability of a delay error  $E(T)$  is defined as the probability of delay of a circuit that exceeds the clock period  $T$ . The minimum effective clock period  $T_{eff}$  is determined by evaluating all feasible clock period  $T$ .

Tables.I-III are the result of simulation. In Table I, II, and III, the number of gates, the maximum and minimum delays, the minimum effective clock period, and the clock period and the probability of delay error when the minimum effective clock period is achieved are shown. Also, the normalized evaluation indices where  $\langle aaaaa \rangle$  is used as 100% are shown. The minimum values of  $T_{eff}$ ,  $P$  and  $PT_{eff}$  over all configurations are shown in bold.

### A. Difference of Full-Adders

Table I shows the effects of difference of FAs. In this evaluation, each RCA consists of one type of full-adders. Even though the static evaluations of  $F_a$  and  $F_b$  are the same as  $F_c$  and  $F_d$ , respectively, the minimum effective clock periods are different. The maximum delay and the minimum effective clock period of  $\langle abbbbbb \rangle$  is minimum

TABLE II  
PERFORMANCES OF RCAs USING TWO TYPES OF FULL-ADDER

Name	Configuration						No. of gates	Delay[ns]		Period[ns]		$E(T)$ [%]	$T_{eff}$ [%]	$P$ [%]	$PT_{eff}$ [%]
	bit-1	bit-2	bit-3	bit-4	bit-5	bit-6		max	min	eff.	clock				
$\langle aaaaa \rangle$		$F_a(0)$					49	26	4	22.58	22	2.6	100.0	100.0	100.0
$\langle abaaaa \rangle$			$F_a(0)$				50	24	4	21.09	20	5.5	93.4	101.4	94.7
$\langle ababaa \rangle$				$F_b(0)$			51	24	4	21.09	20	5.5	93.4	105.0	98.1
$\langle abbaaa \rangle$	$H_a(0)$	$F_b(0)$					51	24	4	20.89	20	4.4	92.5	104.2	96.4
$\langle abbaba \rangle$				$F_a(0)$			52	24	4	20.89	20	4.4	92.5	107.6	99.5
$\langle abbbaa \rangle$					$F_b(0)$		52	24	4	20.89	20	4.4	92.5	106.7	98.7
$\langle abbbbb \rangle$						$F_b(0)$	54	24	4	20.89	20	4.4	92.5	111.6	103.2

TABLE III  
PERFORMANCES OF RCAs WITH DELAY INSERTION

Name	Configuration						No. of gates	Delay[ns]		Period[ns]		$E(T)$ [%]	$T_{eff}$ [%]	$P$ [%]	$PT_{eff}$ [%]
	bit-1	bit-2	bit-3	bit-4	bit-5	bit-6		max	min	eff.	clock				
$\langle abaaaa \rangle(2)$			$F_a(0)$				53	24	6	19.81	18	10.1	87.7	104.7	<b>91.9</b>
$\langle abbaaa \rangle(2)$	$H_a(2)$	$F_b(0)$		$F_a(0)$			54	24	6	19.98	18	11.0	88.5	107.5	95.2
$\langle abbbaa \rangle(2)$			$F_b(0)$	$F_b(0)$			55	24	6	19.98	18	11.0	88.5	110.0	97.4
$\langle abaaaa \rangle(4)$				$F_a(0)$			59	24	8	18.65	16	16.6	82.6	112.6	93.0
$\langle abbaaa \rangle(4)$	$H_a(4)$			$F_a(0)$			60	24	8	18.65	16	16.6	82.6	115.4	95.3
$\langle abbbaa \rangle(4)$				$F_b(0)$	$F_b(0)$		61	24	8	18.22	16	13.9	<b>80.7</b>	117.9	95.1
$\langle bbaaaa \rangle(4)$		$F_b(2)$		$F_a(0)$			59	25	8	19.13	17	12.5	84.7	112.5	95.3
$\langle bbbaaa \rangle(4)$	$H_b(4)$			$F_a(0)$			60	25	8	19.13	17	12.5	84.7	115.7	98.0
$\langle bbbbaa \rangle(4)$				$F_b(0)$	$F_b(0)$		61	25	8	18.97	17	11.6	84.0	118.3	99.4

among them. While, the average energy consumption of  $\langle baaaa \rangle$  is minimum among them. The average energy consumption of  $\langle baaaa \rangle$  is small because not only the number of gates is small but also the number of switchings is small since the cases that inputs of a gate in FA at bit-2 are inputted simultaneously are large.

### B. Combination of Full-Adders

Table II shows the effects when two types of full-adders are used. In this evaluation, each RCA except  $\langle aaaaa \rangle$  consists of  $H_a$ ,  $F_a$  and  $F_b$ . In these RCAs, the maximum delay is 24 [ns] when  $F_b$  is used at bit-2. The minimum effective clock period is minimum among them when  $F_b$  is used both at bit-2 and bit-3. The average energy consumption tends to be large when the number of gates is large. Even though the number of gates of  $\langle abbaaa \rangle$  and  $\langle abbbaa \rangle$  are the same as  $\langle ababaa \rangle$  and  $\langle abbaba \rangle$ , respectively, the average energy consumptions of formers are small. The average energy consumption is small when  $F_b$  is used in series, since the probability of switching of the gate that outputs the carry signal of  $F_b$  is small.

### C. Insertion of Delay

Table III shows the effects of insertion of delay. In this evaluation, the delays are minimally inserted by changing the sizes of BUFs so that the minimum delay is increased without changing the maximum delay. The minimum effective clock period of  $\langle abbbaa \rangle(4)$  is minimum among them. The minimum effective clock period of  $\langle abbbaa \rangle(4)$

is 19.3% shorter than  $\langle aaaaa \rangle$ . This is mainly due to the fact that the maximum delay is small and that the shorter clock period is feasible since the minimum delay is larger. In addition, the probability of delay error of  $\langle abbbaa \rangle(4)$  is smaller than the other configurations whose maximum and minimum delays are the same as  $\langle abbbaa \rangle(4)$ .

The minimum delay can be increased up to 8 [ns] without increasing the maximum delay by changing the sizes of BUFs. The minimum delay can be increased more without increasing the maximum delay by inserting delays to arbitrary places. Even if the clock period is set shorter than 16 [ns] by inserting delays more, the effective clock period is not reduced since the probability of delay error increases much.

The average energy consumptions of them are larger than  $\langle aaaaa \rangle$ . PT product of  $\langle abaaaa \rangle(2)$  is minimum over all configurations shown in this paper. Even though the average energy consumption of  $\langle baaaa \rangle$  is minimum over all configurations shown in this paper, the average energy consumption of  $\langle bbaaaa \rangle(4)$  is larger than  $\langle abbaaa \rangle(4)$ . That is, the replacement of  $H_a$  with  $H_b$  does not necessarily reduce the energy consumption.

### D. Distribution of delay

Fig. 4 shows the distributions of delay of 6-RCAs. The maximum delay and the minimum delay of them are 24 [ns] and 8 [ns], respectively. The distributions of delay would change drastically even if the maximum delay and the minimum delay are the same. Table IV shows the probabilities of delay errors of them in the cases where

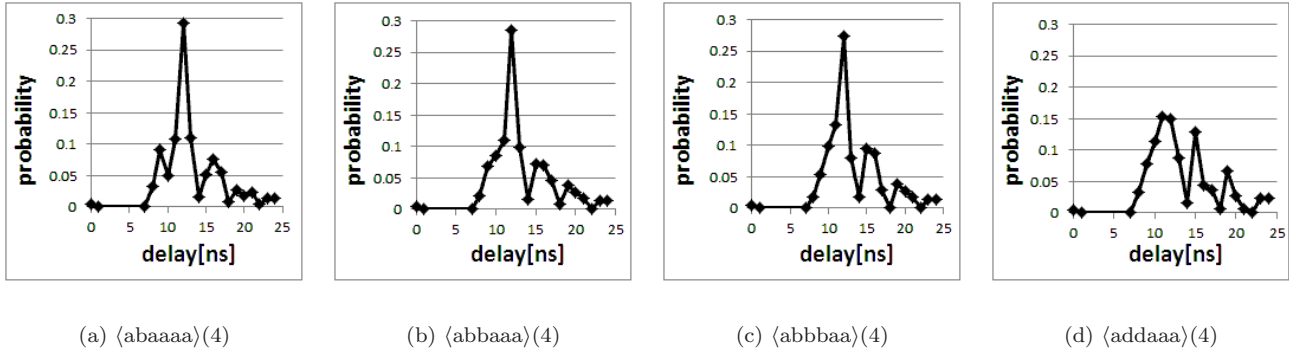


Fig. 4. Distribution of delay

TABLE IV  
PROBABILITY OF ERROR AND EFFECTIVE CLOCK PERIOD

Name	$T=16$		$T=18$		$T=20$	
	$E(T)$ [%]	$T_{eff}$ [ns]	$E(T)$ [%]	$T_{eff}$ [ns]	$E(T)$ [%]	$T_{eff}$ [ns]
$\langle abaaaa \rangle(4)$	16.6	18.66	10.1	19.82	5.5	21.10
$\langle abbaaa \rangle(4)$	16.6	18.66	11.0	19.98	4.4	20.88
$\langle abbbaa \rangle(4)$	13.9	18.22	11.0	19.98	4.4	20.88
$\langle bbbbaa \rangle(4)$	19.1	19.06	14.9	20.68	5.4	21.08

the clock period is 16 [ns], 18 [ns], and 20 [ns]. The distribution of delay should be taken into account to minimize the effective clock period.

## VII. SUMMARY AND FUTURE WORKS

In this paper, we evaluated the performance of various 6bit RCA supposing these RCA work in VLEDC. The improvement of effective clock period by introducing VLEDC can be larger by making the minimum delay larger without increasing the maximum delay. The distribution of delay can be changed so that the probability of delay error is reduced by modification of circuits. The performance of a circuit in VLEDC is improved much if the probability of large delay is reduced by circuit modification.

As future works, a method for evaluating the delay and the energy consumption of a circuit efficiently and a method for improving the distribution of delay of practical circuits are needed to be investigated.

## ACKNOWLEDGEMENTS

This work is supported by VLSI Design and Education Center (VDEC), the University of Tokyo in collaboration with Synopsis, Inc. and Grant-in-Aid for Scientific Research (B) 21300012.

## REFERENCES

- [1] G. Wolrich, E. McLellan, L. Harada, J. Montanaro, and R. Yodlowski, "A High Performance Floating-Point Coprocessor," *IEEE Journal of Solid-State Circuits*, vol. SC-19, pp.690-696, Oct. 1984.
- [2] D. Ernst, N. S. Kim, S. Das, S. Pant, T. Pham, R. Rao, C. Ziesler, D. Blaauw, T. Austin, T. Mudge, and K. Flautner, "Razor: A Low-Power Pipeline Based on Circuit-Level Timing Speculation," *Proc. 36th Annual IEEE/ACM International Symposium on Microarchitecture*, pp. 7-18, 2003.
- [3] Y. Ukon, A. Takahashi, and K. Taniguchi "An evaluation of delay error rate of an adder in terms of clock period," in *IEICE Technical Report (ICD2009-91)*, Vol.109, No.336, pp.77-81, 2009. (in Japanese)
- [4] M.Kurimoto, H.Suzuki, R.Akiyama, T.Yamanaka, H.Ohkuma, H.Takata, H.Shinohara, "Phase-Adjustable Error Detection Flip-flops with 2-Stage Hold-Driven Optimization, and Slack Based Grouping Scheme and Slack Distribution Control for Dynamic Voltage Scaling," *ACM Trans. DAES*, Vol.15, No.2 Article 17, 2010.
- [5] D. Bull, S. Das, K. Shivashankar, G. Dasika, K. Flautner, and D. Blaauw, "A Power-Efficient 32b ARM ISA Processor Using Timing-Error Detection and Correction for Transient-Error Tolerance and Adaptation to PVT Variation," *Proc. IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 284-285, 2010.
- [6] M. Inoue, Y. Ukon, and A. Takahashi, "An evaluation of error detection/correction circuits by gate level simulation," *IEICE Technical Report (VLD2010-141)*, Vol.110, No.432, pp.147-152, 2011. (in Japanese)
- [7] Y. Ukon, M. Inoue, A. Takahashi, and K. Taniguchi, "Behavioral Verification of a Variable Latency Circuit on FPGA," *IEICE Technical Report (VLD2010-142)*, Vol.110, No.432, pp.153-158, 2011. (in Japanese)
- [8] A. Takahashi, Y. Kajitani, "Performance and Reliability Driven Clock Scheduling of Sequential Logic Circuits," *Proc. of Asia and South Pacific Design Automation Conference '97*, pp. 37-42, 1997.
- [9] H. Fujioka, K. Nakamae, *Calculator system —The base of hardware—*, Shokodo, 2007. (in Japanese)