

An Adaptive-Clocking-Control Circuit With 7.5% Frequency Gain for SPARC Processors

Tetsutaro Hashimoto^{ID}, Yukihito Kawabe, Michiharu Hara, Yasushi Kakimura, Kunihiko Tajiri, Shinichiro Shirota, Ryuichi Nishiyama, Hitoshi Sakurai, Hiroshi Okano, Yasumoto Tomita, Sugio Satoh, and Hideo Yamashita

Abstract—On-die supply-voltage droops attributed to workload variations degrade the performance of high-performance microprocessors. An adaptive-clocking-control circuit was implemented for mitigating the adverse impact of supply-voltage droops on processor performance. One of the most critical requirements for adaptive-clocking supply-droop mitigation is that clock-frequency adaptation is fast enough to respond to such supply droops. To shorten the clock-frequency-adaptation latency, therefore, the adaptive-clocking-control circuit features the following schemes: the time-to-digital converter (TDC) based on multipath delay line (multipath TDC), thermometer-code-based data-processing logic, and phase-locked loop (PLL) including a direct frequency-reduction mechanism. The multipath TDC reduces quantization errors in droop detection to shorten detection-response latency. The thermometer-code-based logic does not cost extra clock cycles compared with binary-code-based logic. The direct frequency-reduction mechanism enables a PLL to quickly react to clock-modulation instruction without any intervals. These schemes contribute to faster clock-frequency-adaptation response to supply droops. A test chip including the adaptive-clocking-control circuit with SPARC processor cores was fabricated in a 20-nm CMOS process. Experimental measurements indicate that the adaptive-clocking-control circuit achieved a state-of-the-art frequency gain of 7.5%, resulting in an operating frequency as high as 5 GHz.

Index Terms—Adaptive circuit, adaptive clocking, adaptive frequency, supply-droop mitigation, supply-voltage droop.

I. INTRODUCTION

WITH advances in very-large-scale integration technology, an increasing number of transistors have been integrated into server processors for high-performance parallel computing. Workload variations in such processors (e.g., most arithmetic units in a processor concurrently transit from idle to active) induce a sudden current surge, which is referred to as a di/dt event, and a current surge concomitantly induces supply droops in the power-delivery network, as illustrated in Fig. 1. A di/dt event causes the first droop in the power-delivery network. The magnitude and frequency of the first droop depend on the package inductance and on-die capacitance, and its frequency ranges from tens to a few hundred megahertz. Second and third droops are induced concomitantly with a

Manuscript received August 6, 2017; revised October 10, 2017; accepted November 10, 2017. Date of publication December 14, 2017; date of current version March 23, 2018. This paper was approved by Guest Editor Makoto Ikeda. (Corresponding author: Tetsutaro Hashimoto.)

The authors are with Fujitsu Laboratories Limited, Kawasaki 211-8588, Japan (e-mail: thasimo@jp.fujitsu.com).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JSSC.2017.2777101

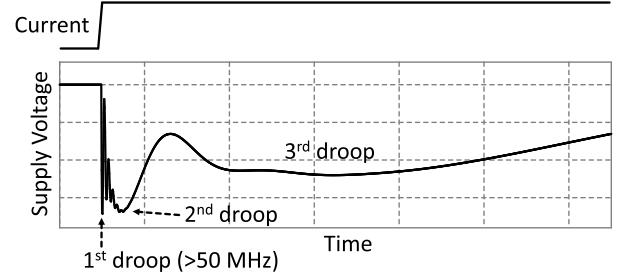


Fig. 1. Supply-voltage droop in on-die power-supply network.

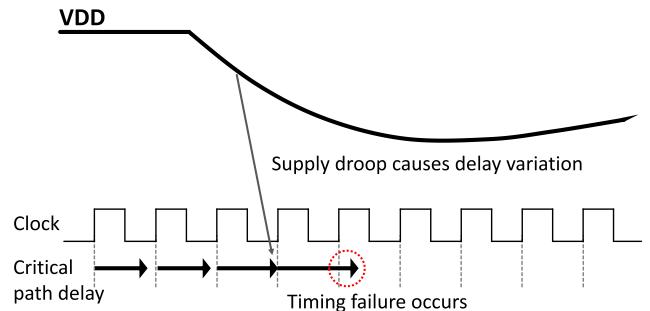


Fig. 2. Timing failures caused by supply-voltage droop.

first droop. The magnitude and frequency of the second droop depend on the inductive traces on the board and on-package decoupling capacitors, and the magnitude and frequency of the third droop are related to the bulk capacitors on the board. It is relatively easy to minimize the third droop by additional bulk capacitors. However, because putting additional on-die capacitance and on-package decoupling capacitors is costly, first and second droops are difficult to suppress.

High-performance processors are more susceptible to supply droops. Large supply droops cause critical paths in a processor to become slow, and timing failures occur if the critical-path delays exceed a clock period, as illustrated in Fig. 2. For correct execution, therefore, the maximum frequency or minimum supply voltage of a processor is determined after adding a frequency or voltage guardband as a margin with the worst case supply droop considered. In particular, high-performance processors for mission-critical servers require an additional guardband with rare worst-case workloads taken into account. Such a guardband is excessive for typical workloads. Thus, processors are operated at a lower frequency or higher voltage than the frequency–voltage curve derived from performance

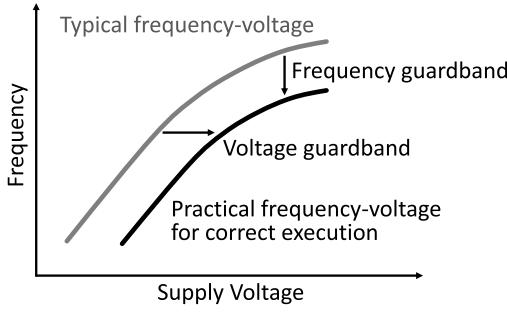


Fig. 3. Guardband approach.

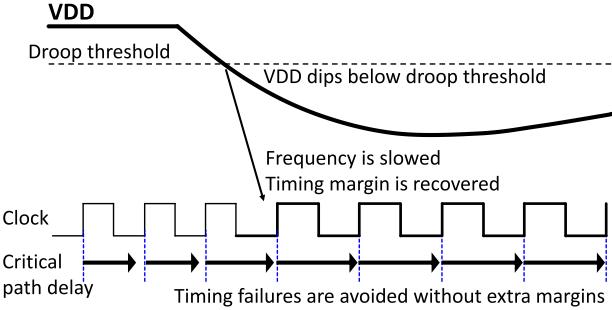


Fig. 4. Timing-margin recovery by adaptive clocking.

tests that utilize typical workloads, as illustrated in Fig. 3. A frequency guardband directly degrades the maximum frequency. A voltage guardband wastes additional power and produces further heat, and thermal constraints also depress the maximum frequency.

Therefore, several adaptive-clocking techniques have been attempted to mitigate the adverse impact of supply-voltage droops on processor performance without an additional frequency or voltage guardband [1]–[7]. With adaptive-clocking applied, when supply-voltage dips below a droop-threshold voltage, clock frequency is slowed so that critical-path delays do not exceed a clock period, as illustrated in Fig. 4. This means adaptive clocking can recover the timing margin in which supply voltage is drooping and prevent timing failures without an additional guardband.

We present our adaptive-clocking-control circuit for fast clock-frequency-adaptation response to supply droops [8]. It features the following schemes.

- 1) Time-to-digital converter (TDC) based on a multipath delay line (multipath TDC) to reduce quantization errors for quick droop detection.
- 2) Thermometer-code-based data-processing logic to eliminate extra clock cycles.
- 3) Phase-locked loop (PLL) whose ring oscillator counts of the digitally controlled oscillator (DCO) can be directly changed for quick reaction to clock-modulation instruction.

The remainder of this paper is as follows. In Section II, the critical requirement for adaptive clocking is discussed. Section III is focused on clock-frequency-adaptation-latency reduction. In Section IV, our adaptive-clocking-control circuit featuring various response-latency-reduction schemes

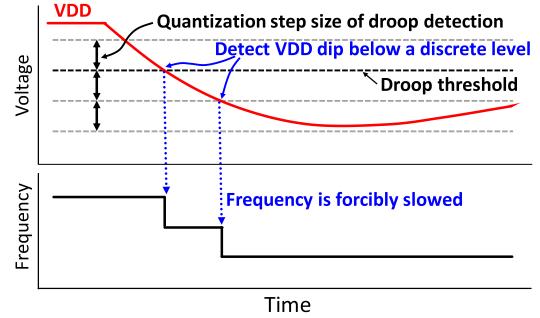


Fig. 5. Clock frequency adaptation by droop-detector-based adaptive-clocking technique.

is presented. Section V presents silicon measurements obtained from a test chip we fabricated that includes our adaptive-clocking-control circuit with SPARC processor cores. In Section VI, a summary of this paper and concluding remarks are given.

II. CRITICAL REQUIREMENT FOR ADAPTIVE-CLOCKING CONTROL FOR SUPPLY-DROOP MITIGATION

For correct execution of processors, it is necessary for adaptive-clocking control to address all first, second, and third droops. One of the most critical requirements for adaptive clocking is that clock-frequency-adaptation latency should be low enough to respond to supply droops. If clock-frequency-adaptation response is too late for supply droops, the effect of adaptive clocking on timing-margin recovery will be diminished. The first-droop noise is the highest frequency among supply droops shown in Fig. 1, and is the most difficult for adaptive-clocking control to mitigate.

Adaptive analog frequency/supply tracking [1]–[3] is an effective technique for addressing the first droop in terms of fast response to supply droops. The output clock frequency of a PLL is promptly modulated with digital supply-voltage variation because this technique mixes digital supply with voltage-controlled oscillator supply and does not require droop detectors or data-processing logics. However, it cannot address second and third droops whose drooping durations are longer than the PLL bandwidths. After a PLL starts to restore its output clock frequency in the middle of the second droop, less timing margin is recovered.

A droop-detector-based adaptive-clocking technique forcibly slows the clock frequency each time a droop-detector detects a supply droop cross a discrete level, as illustrated in Fig. 5. Compared with adaptive analog frequency/supply tracking [1]–[3], droop-detector-based adaptive-clocking techniques [4]–[7] are advantageous in that they can address second and third droops. They can also ignore small droops by setting a droop threshold [5], [7]. If an output clock frequency rapidly rebounds to the original clock frequency after a supply-voltage settles, a supply-current surge will be induced. These techniques can also prevent such a current surge by controlling the clock-frequency-rebounding rate [6], [7]. However, these techniques' clock-frequency-adaptation responses are not fast enough to effectively respond

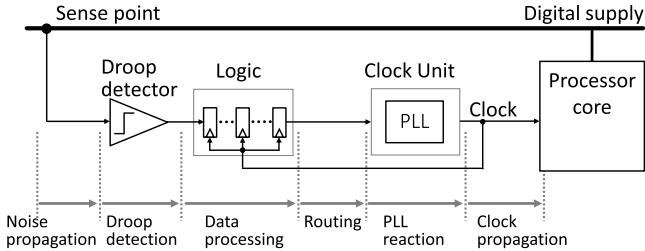


Fig. 6. Overview of droop-detector-based adaptive-clocking-control circuit and total clock-frequency-adaptation latency.

to a first droop because operations in a droop detector, data-processing logic, and a clock unit shown in Fig. 6 takes several clock cycles. Droop detection lags somewhat behind a supply droop. Because the droop-detection signal output from a droop detector is asynchronous and thermometer coded, synchronization and binary conversion, which cost several clock cycles, are generally needed in a logic block between a droop detector and PLL because of processing with binary code. A clock unit also requires a few clock cycles to react to clock-modulation instruction. Thus, a droop-prediction mechanism has been introduced to adapt clock frequency in advance of the first droop and to effectively mitigate this droop [6]. However, its prediction precision was not sufficient for it to be applied to mission-critical servers.

III. CLOCK-FREQUENCY ADAPTATION LATENCY REDUCTION

A droop-detector-based adaptive-clocking technique has an advantage in mitigating second and third droops compared with an adaptive analog frequency/supply tracking technique. The droop-detector-based technique can also ignore small droops by setting a droop threshold. However, because high response latency of clock-frequency adaptation with this technique limits the effect of adaptive clocking on timing-margin recovery, the adaptive circuit requires to shorten its adaptation latency so that its clock-frequency adaptation effectively responds to the first droop. Its total latency is composed of noise propagation, droop detection, data processing, routing, PLL reaction, and clock propagation, as shown in Fig. 6. We focused on droop detection, data processing, and PLL reaction for shortening the total adaptation latency.

A. Droop Detection

A droop detector converts monitored supply voltage from a continuous-time and continuous-amplitude analog waveform to a discrete-time and discrete-amplitude digital waveform. Some adaptive-clocking techniques [4]–[6] adopt a droop detector based on TDC architecture shown in Fig. 7(a). The TDC block measures the phase difference between the compared phase (CMP) and the reference phase that is attributed to the supply-voltage variation fed to the tunable delay line. Each gate delay [Δ in Fig. 7(a)] determines the timing resolution of the TDC, and then the timing resolution determines the quantization step size (alternatively, the voltage resolution, or the quantization error) of droop detection. In Fig. 7, for the sake

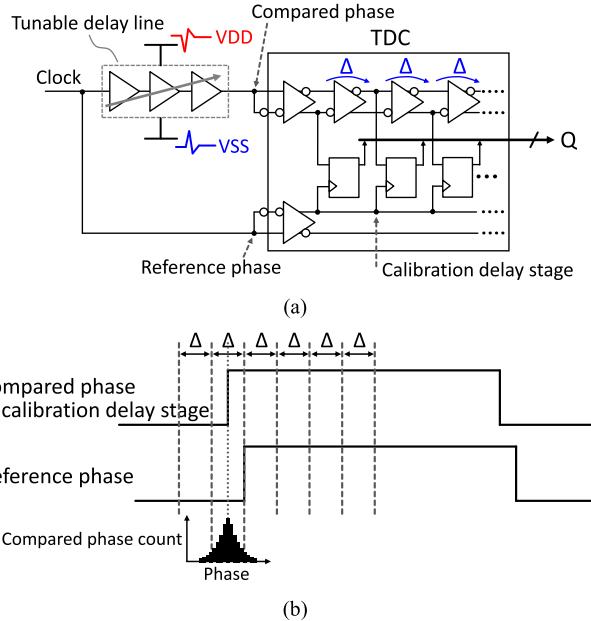


Fig. 7. Droop detector based on TDC architecture. (a) Basic diagram of a droop detector based on TDC. (b) Timing chart of compared and reference phases and histogram of CMP at the calibration delay stage after calibration.

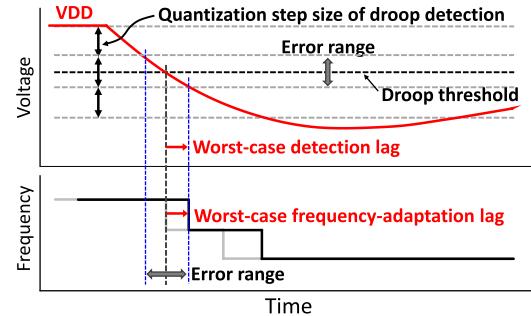


Fig. 8. Effect of quantization step size on clock-frequency-adaptation response.

of explanation, we assumed the second delay stage as the calibration delay stage. The propagation delay of the tunable delay line is calibrated so that the compared edge and the reference edge arrives statistically in the same phase at the calibration delay stage in the TDC block, as illustrated in Fig. 7(b). Thus, the quantization step size of droop detection determines the error range shown in Fig. 8. A certain discrete value is set as a droop threshold, but the analog voltage level indicated by this discrete value can deviate from the true analog droop-threshold voltage within the error range, as shown in Fig. 8. This voltage deviation can cause the droop-detection response to lag further behind a supply droop. The detection response lag causes the clock-frequency-adaptation response lag.

In summary, reducing the quantization error in droop detection can contribute to shortening the total latency of the clock-frequency-adaptation response to supply droops.

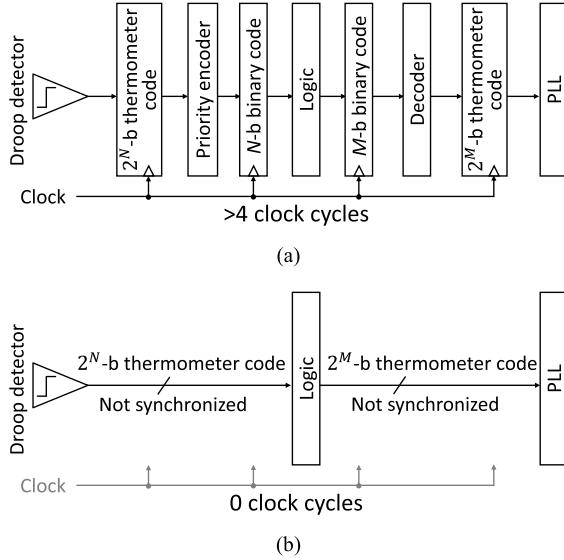


Fig. 9. Comparison between binary-code-based and thermometer-code-based data processing. (a) Binary-code-based data processing. (b) Thermometer-code-based data processing.

B. Data Processing Between Droop Detector and PLL

A logic block between a droop detector and PLL processes the droop-detection signal into a clock-modulation signal. Generally, the droop-detection signal is thermometer coded, and the logic block processes with binary code [4], [6], [7]. Thus, the droop-detection signal needs to be converted once into binary code by a priority encoder. Binary-code-based processing also requires synchronous transmission. Depending on the clock adaptation scheme at a PLL, the clock-modulation signal as binary code needs to be converted into thermometer code. Consequently, it costs at least four clock cycles to deliver the signal from a droop detector to a PLL, as shown in Fig. 9(a).

While the higher order digits of binary code carry more weight, all digits of thermometer code are the same weights. This means that the influence of glitches and skew between bits on logic is even smaller in thermometer code than in binary code. Besides, thermometer code is consecutive-digit change. Accordingly, if a thermometer-coded signal is restricted so as to monotonically decrease or increase, thermometer-code-based processing does not necessarily require synchronous transmission to protect glitches from propagating and to minimize skew between bits. Moreover, it does not necessarily require the use of a priority encoder or decoder. As a result, it costs only a propagation delay to deliver the signal from a droop detector to a PLL, as shown in Fig. 9(b).

C. PLL Reaction

After the droop-detection signal is processed into a clock-modulation signal in the logic block, the clock-modulation signal is delivered to the PLL. If the clock-modulation signal goes through the PLL normal control path, it is required to be synchronized, and then PLL reaction will cost a few clock

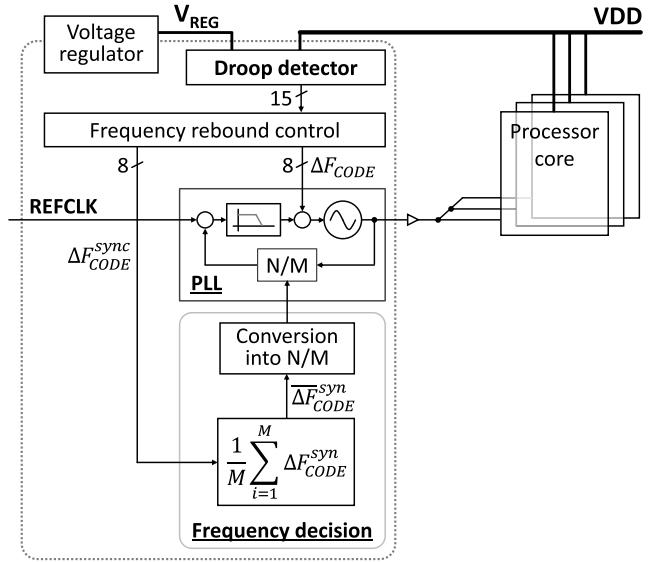


Fig. 10. Overview of adaptive-clocking-control circuit for supply-droop mitigation.

cycles [7]. For PLL quick reaction, therefore, it is necessary to bypass the PLL normal control path and directly modulate its output clock frequency.

IV. CIRCUIT IMPLEMENTATION

An adaptive-clocking-control circuit should provide a fast clock-frequency-adaptation response not to be too late for the first droop. Thus, we set a design target at which the adaptation response should be 8× faster than the first-droop-noise frequency because the clock frequency is required to be slowed in sufficiently advanced before the first droop reaches the lowest point.

Our adaptive-clocking-control circuit consists of a droop detector, thermometer-code-based frequency-rebound control logic, PLL, and frequency-decision logic, as shown in Fig. 10. The droop detector monitors the digital supply (VDD) and converts it into a droop-detection signal as 15-bit thermometer code. The droop detector uses a regulator output voltage as a reference voltage. The logic following the droop detector processes the droop-detection signal into a clock-modulation signal (ΔF_{CODE}). The ΔF_{CODE} is an 8-bit thermometer-coded signal and is fed to the PLL, and a synchronized clock-modulation signal (ΔF_{CODE}^{sync}) is fed to frequency-decision logic. The PLL's output clock frequency is forcibly slowed according to the ΔF_{CODE} by bypassing its normal control path. The frequency-decision logic computes the average ΔF_{CODE}^{sync} and converts it into the PLL's multiplication/division ratio by

$$\frac{M_k}{N_k} = \frac{M_8}{N_8} - (8-k) \frac{\Delta F}{F_{refclk}} \quad (k = 0, \dots, 8) \quad (1)$$

where $k = \lfloor \overline{\Delta F}_{CODE}^{sync} + 0.5 \rfloor$, M_8/N_8 is the multiplication/division ratio for the operating frequency, ΔF is one step size of frequency scaling, and F_{refclk} is the reference clock frequency. If only one first droop is induced, the average ΔF_{CODE}^{sync} remains 8, and consequently, the PLL keeps its

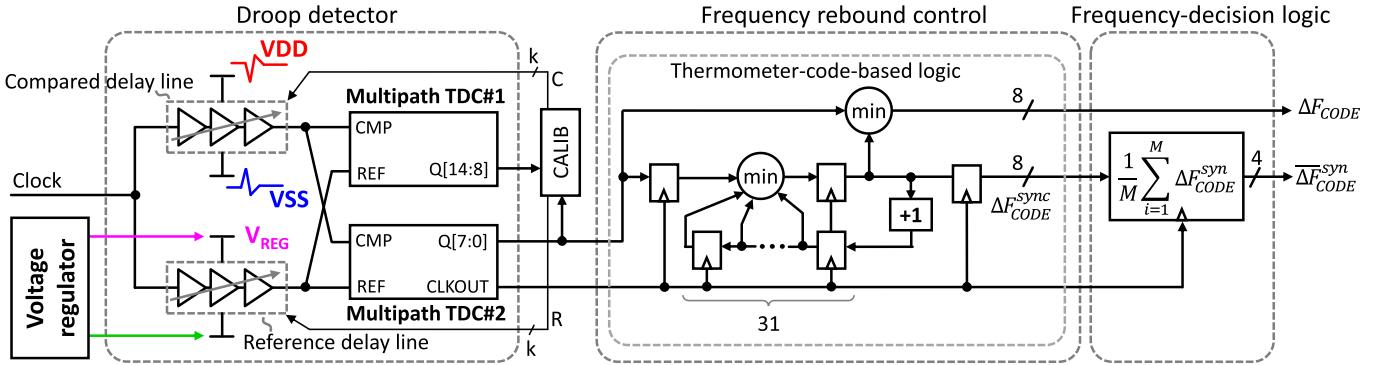


Fig. 11. Block diagram of droop detector, thermometer-code-based frequency-rebound control logic, and frequency-decision logic.

clock frequency at the operating frequency. If first droops are repetitively induced, the average ΔF_{CODE}^{syn} drops below 8, and then, the PLL reduces its clock frequency to adapt to repetitive first droops. This control path prevents the PLL from restoring its output clock frequency while the clock frequency is forcibly slowed by the ΔF_{CODE} . The ratio is also fed to the PLL for addressing the second and third droops.

The frequency-rebound control logic and averaging logic of the frequency-decision logic are synchronized by the same clock source output from the droop detector. Therefore, the droop detector, frequency-rebound control logic, and averaging logic are implemented in one block, as shown in Fig. 11.

A. Droop Detector

We adopted a circuit architecture with which two TDCs measure the delay difference between two same delay lines operating with different power supplies as a droop detector, as shown in Fig. 11. The regulator output voltage (V_{REG}) is set to a droop-threshold voltage and is supplied to the reference delay line. Meanwhile, the compared delay line operates with VDD. Thus, the delay difference between the two delay lines is derived from the voltage difference from the V_{REG} to VDD, and is converted into a droop-detection signal as 15-bit thermometer code. Droop detectors based on TDC architecture of [4]–[6] have the same circuit architecture as a critical path monitor. Therefore, they require calibration at every sampling frequency and temperature point in addition to voltage, whereas our two delay lines' delay-difference-monitor-architecture-based droop detector, by contrast, requires calibration at only one voltage point which is a droop-threshold voltage. This results from that the two delay lines' propagation delays comparably deviate due to temperature variation and the delay difference between them does not vary depending on its sampling frequency.

The timing resolution of a TDC determines the quantization error in droop detection and the quantization error increases the droop-detection-response latency, as described in Section III-A. The delay per stage determines the timing resolution of the TDC. Thus, to speed-up delay cells will result in quantization error reduction and faster droop-detection response. The two TDCs use a multipath delay line to speed up their delay cells, as shown in Fig. 12. The multiple-input

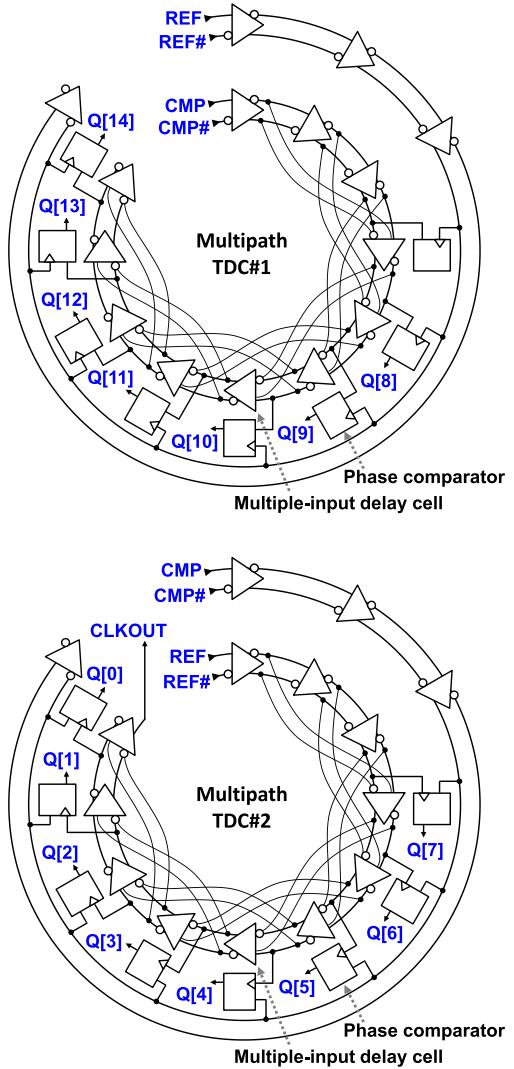


Fig. 12. Schematic of TDCs based on multipath delay line (multipath TDC).

structure taps several previous delay stages for input [9], [10]. In the example shown in Fig. 13, Z_0 is applied to the input. Thus, Z_3 already starts rising before Z_2 is applied to the input. Therefore, the multiple-input structure speeds up the transition time at the output compared with a single-input structure, reducing the gate delay by half. Accordingly, the

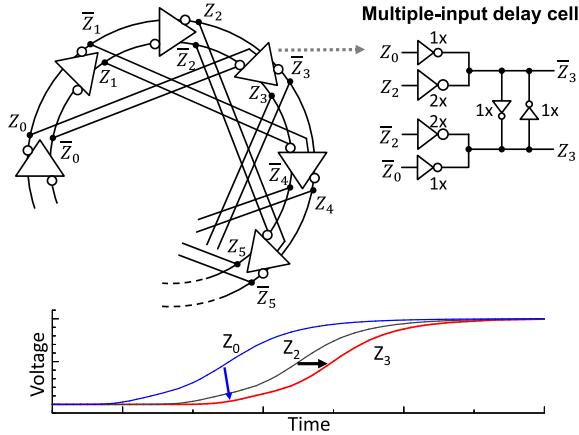


Fig. 13. Multipath delay line and multiple-input delay cell.

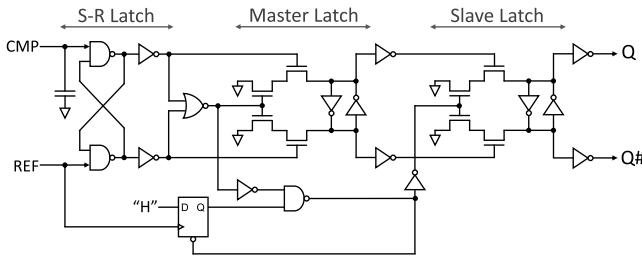


Fig. 14. Schematic of phase comparator.

multipath delay line provided 2 \times finer timing resolution, and then the quantization error in droop detection was reduced by half, which resulted in shortening the droop-detection-response latency by the equivalent of two clock cycles.

Fig. 14 shows the schematic of a phase comparator that compares the CMP with the reference phase (REF) at each delay stage in the TDCs shown in Fig. 12. The phase comparator outputs “1” if the CMP leads the REF, and outputs “0” if the CMP lags the REF. The dead-zone time of the phase comparator should be narrow enough to achieve a finer timing-resolution TDC. Thus, we adopted a set-reset latch (S–R latch) as a phase comparator. As is well known, the dead-zone time of an S–R latch is very narrow. We had found through circuit simulations and experiments that its dead-zone time was even less than 1 ps [11]. A narrow dead-zone time contributes to improving metastability in the phase comparison and achieving higher frequency operation of a TDC. A master and slave latch following the S–R latch shown in Fig. 14 holds and outputs a phase comparison result for one clock cycle.

B. Frequency Rebound Control Logic

The droop detector generates a 15-bit thermometer-coded signal (Q in Fig. 11). The “ $Q = 8$ ” indicates that the VDD is judged as equal to the droop-threshold voltage. The $Q[14:8]$ is used only when calibrating the two tunable delay lines (the compared delay line and the reference delay line in Fig. 11) to the same delay, and $Q[7:0]$ is delivered to the PLL after it is processed in the frequency-rebound control logic. The logic limits the increase rate of ΔF_{CODE} to one tap every 32 cycles, whereas it does not limit the decrease rate of ΔF_{CODE} .

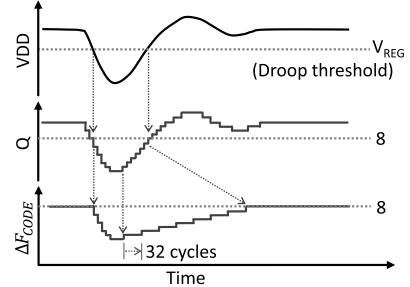


Fig. 15. Timing diagram of droop detector and frequency-rebound control logic.

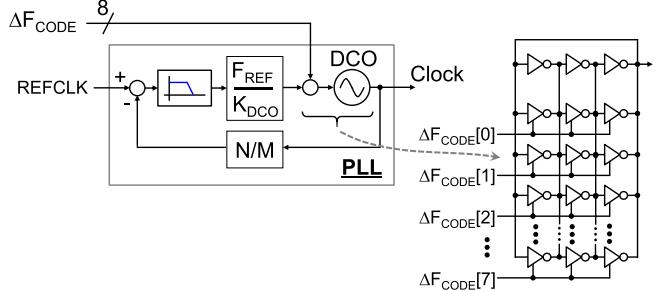


Fig. 16. PLL with direct frequency-reduction scheme.

as shown in Fig. 15. Rapidly increasing frequency causes a supply-current surge. Thus, this operation prevents the clock frequency from rapidly rebounding after a supply droop starts to settle, and then avoids a supply-current surge. Moreover, this operation masks the asynchronous droop-detection signal Q with the synchronized minimum droop-detection code so as to protect glitches occurring and incorrectly increasing frequency. As a result, the influence of glitches on frequency change is significantly mitigated, and thus synchronous transmission of ΔF_{CODE} is not necessarily required.

By processing the thermometer code as it is, priority encoder logic for binary conversion, decoder logic for thermometer conversion, and latches for synchronization can be eliminated. As a result, it does not cost any clock cycles to deliver $Q[7:0]$ to a PLL when ΔF_{CODE} decreases. It costs only a propagation delay equivalent to one clock cycle. This contributes to reducing the latency in data processing from 4 cycles to 1 cycle.

C. PLL With Direct Frequency-Reduction Mechanism

To respond to the first droop, the PLL should react as quickly as possible to the clock-modulation signal. Fig. 16 shows a block diagram of the PLL including a mechanism for direct frequency reduction. The 8-bit thermometer-coded clock-modulation signal (ΔF_{CODE}) is connected to the eight enable terminals of ring oscillators comprising the DCO, bypassing the PLL normal control path. When $\Delta F_{CODE}[n] = 0$, n th ring oscillator turns disabled and then the output clock frequency is reduced by one tap, as shown in Fig. 17. When $\Delta F_{CODE}[n] = 1$, the n th ring oscillator turns enabled and then the output clock frequency is rebounded toward the original frequency by one tap, as shown in Fig. 17. Because all wires of the thermometer-coded signal are the same weight unlike the wires of a binary-coded signal

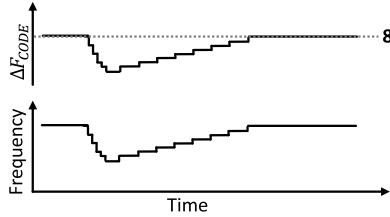


Fig. 17. Timing diagram of clock modulation at PLL.

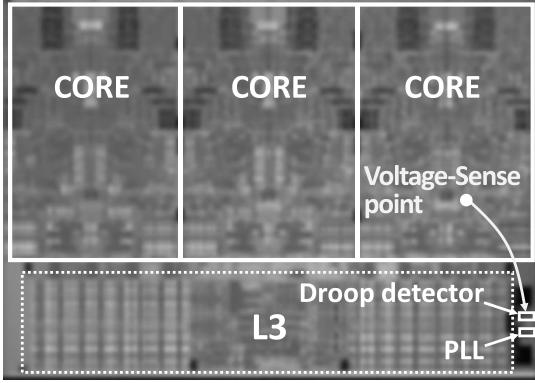


Fig. 18. Chip micrograph.

and the ΔF_{CODE} monotonically decreases or increases, the arrival time difference among the eight wires of the ΔF_{CODE} does not cause the number of enabled ring oscillators to temporally become incorrect, which eliminated the need for synchronization. In this way, the ΔF_{CODE} reaches the eight enable terminals of ring oscillators without synchronization and can directly change the number of enabled ring oscillators without any intervals. The PLL can quickly react to the clock-modulation instruction without any intervals.

V. EXPERIMENTAL RESULTS

A test chip including our adaptive-clocking-control circuit with SPARC processor cores was fabricated in a 20-nm CMOS process [8]. Adaptive-clocking control was applied to three SPARC processor cores, as shown in Fig. 18. If there was a sense point in each processor core, a droop detector would be installed for each sense-point and the minimum ΔF_{CODE} among them would be fed to the PLL and the frequency-decision logic. Two test programs were used to induce supply-voltage droops: LINPACK kernel (DGEMM) and our program that induces a rare worst case supply-voltage droop.

The droop detector was placed near the PLL to shorten routing and propagation delay from the droop detector to the PLL. Because an on-package trace has much lower impedance than an on-die metal wire and noise propagation in an on-package trace is much faster than in an on-die metal wire, a wide and low impedance on-package trace was used for supply-voltage sensing. Placing the droop detector near the PLL and using on-package trace for voltage sensing contributed to shortening the latency by the equivalent of four clock cycles compared with placing the droop detector near the voltage-sense point and routing far from the droop detector to the PLL. As a result, in addition to latency reduction

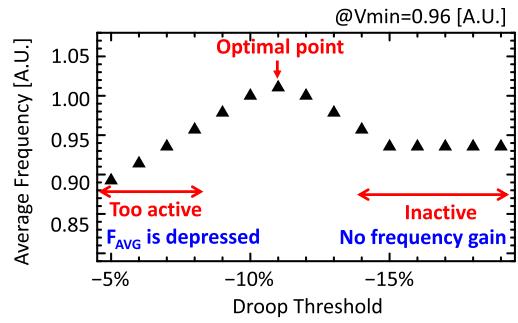


Fig. 19. Optimal droop-threshold voltage.

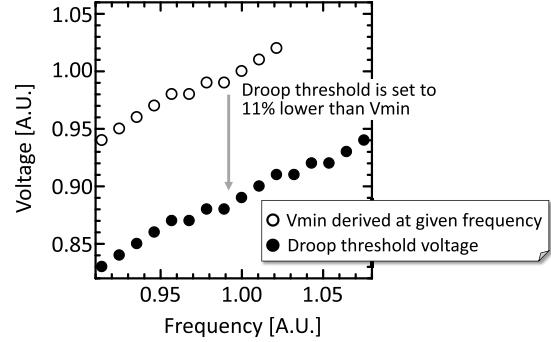


Fig. 20. Droop-threshold setting.

by our proposed adaptive-clocking-control circuit described in Section IV, the total adaptation-response latency from threshold crossing until decreasing frequency at the leaf nodes was reduced by 46% at the typical PVT condition in the chip design, and then fit within the target latency which was 8 \times faster than the first-droop-noise frequency of 50 MHz estimated from the simulated impedance profile.

A droop-threshold voltage is a key factor affecting adaptive-clocking control. When a droop-threshold voltage is too shallow from the VDD, the adaptive-clocking-control circuit will actively apply clock-frequency reduction, which will depress the average frequency. When a droop-threshold voltage is too deep from the VDD, the adaptive circuit will be almost out of action and will not apply clock-frequency reduction in which supply voltage is drooping nor provide frequency gain. We experimentally found that the optimal droop-threshold voltage was around 11% lower than the minimum VDD obtained at a given frequency, as shown in Fig. 19. The voltage of 11% lower than the minimum VDD was sufficiently higher than the threshold voltage of a transistor at every voltage-frequency point in this experiment, which indicates that every point was in the region where the maximum operating frequency changed approximately linearly with the supply voltage. Accordingly, every optimal droop-threshold voltage could be 11% lower than the minimum VDD obtained at a given frequency. Thus, as shown in Fig. 20, a minimum VDD was derived at each given frequency from a shmoop plot with adaptive-clocking disabled, and the droop threshold, which is the V_{REG} in Fig. 11, was set to 11% lower than the minimum VDD at each given frequency. Before the processor was started, propagation delays of the compared delay line

TABLE I
PERFORMANCE COMPARISON

	[1]	[4]	[6]	[7]	This work
Process	45nm	45nm	16nm	14nm	20nm
Application	PC	Server	Automotive	Server	Server
Method	Analog tracking	Critical path monitor based	Droop detector based	Droop detector based	Droop detector based
Frequency gain	5%	2% 80 MHz (3.78→3.86 GHz)	7.4%* 140 MHz (1.88→2.02 GHz) *using prediction	6% (4 GHz)	7.5% 350 MHz (4.65→5 GHz)

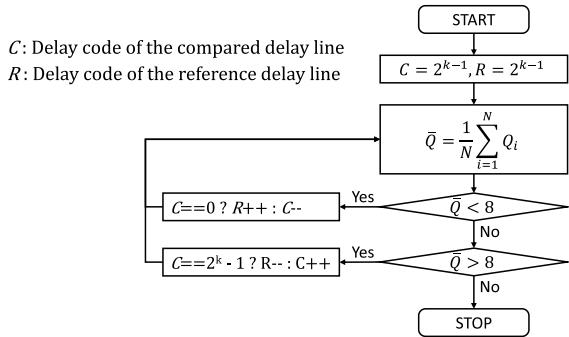


Fig. 21. Auto-calibration diagram.

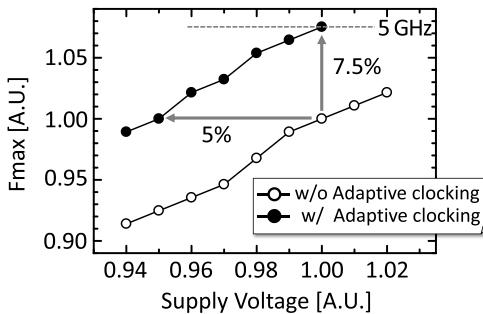


Fig. 22. Performance improvement by adaptive-clocking response to supply voltage. Average frequency depression attributed to adaptive clocking was less than 0.2%.

and the reference delay line in the droop detector shown in Fig. 11 were calibrated to the same delay under the condition that the VDD was equal to the V_{REG} so that the average of Q was 8. The compared and reference delay line were digitally controlled tunable delay lines. Thus, the delay difference between them could be adjusted to nearly zero by their delay codes, as shown in Fig. 21. When the average of Q was 8, the two delay lines' propagation delays were judged as equal, and calibration was finished. The VDD was set the same as V_{REG} while calibration was being performed, and was restored to the operating voltage after calibration was finished.

Fig. 22 shows a comparison of the maximum frequencies obtained at a given supply voltage between application and non-application of adaptive-clocking control. Overall, our adaptive-clocking-control circuit provided a 7.5% increase in the maximum frequency and achieved an operating frequency of 5 GHz. It also provided a supply-voltage decrease of 5%,

which indicates that the adaptive circuit successfully addressed both the first- and second-droop noise. If the adaptation had lagged far behind the first-droop or failed addressing the second-droop, the adaptive circuit would not have provided the voltage gain. The average frequency depression attributed to adaptive-clocking control was less than 0.2% because the duration of reducing frequency was even shorter than the running time of each test program. Moreover, because large droop events that supply droops crossed an 11% droop-threshold voltage did not occur in integer-arithmetic-dominant test programs such as SPECint, the adaptive-clocking-control circuit did not apply clock-frequency reduction under such test programs, and thus the average frequency depression due to adaptive-clocking control was not observed.

The experimental results indicate that our adaptive-clocking-control circuit succeeded in mitigating the impact of supply droops on SPARC processor performance, which demonstrated that our circuit is effective in shortening clock-frequency-adaptation latency and clock-frequency adaptation does not lag behind supply droop. Table I compares the adaptive-clocking-control performance between previous works and this paper. Compared to previous works, the largest frequency gain and highest operating frequency were obtained. Adaptive analog frequency/supply tracking proposed by Kurd *et al.* [1] is superior in terms of fast response to the first droop but cannot address the second and third droops. It is assumed that the second or third droop depresses its frequency gain. The reason our work obtained the largest frequency gain is that our response-latency-reduction schemes achieved the fastest adaptation response to supply droops compared with previous droop-detector-based adaptive-clocking techniques [4], [6], [7].

VI. CONCLUSION

An adaptive-clocking-control circuit was implemented in a 20-nm SPARC processor for supply-droop mitigation. The adaptive circuit contributed to shortening the clock-frequency-adaptation latency to supply droops for improving the effect of adaptive clocking on timing-margin recovery. A multipath TDC shortened the droop-detection-response latency by the equivalent of two clock cycles. Thermometer-code-based data processing reduced processing time by three clock cycles compared with binary-code-based data processing. Directly changing the number of ring oscillators comprising DCO enabled the PLL to quickly react to clock-modulation instruction without any intervals.

Adaptive-clocking control was applied to a SPARC processor, and it was found through experiments that our adaptive-clocking-control circuit effectively mitigates the adverse impact of the on-die supply droops on processor performance to reduce the frequency or voltage guardband. Silicon measurements indicate that the main benefit of the adaptive circuit was that it provided a state-of-the-art frequency gain of 7.5% and achieved an operating frequency of 5 GHz. The adaptive circuit also provided 5% minimum VDD reduction.

ACKNOWLEDGMENT

The authors would like to thank Y. Saito, T. Miura, H. Yamanaka, R. Kan, J. Yamada, A. Konmoto, and T. Iida for adaptive circuit implementation.

REFERENCES

- [1] N. Kurd, P. Mosalikanti, M. Neidengard, J. Douglas, and R. Kumar, "Next generation Intel Core micro-architecture (Nehalem) clocking," *IEEE J. Solid-State Circuits*, vol. 44, no. 4, pp. 1121–1129, Apr. 2009.
- [2] D. Jiao, B. Kim, and C. H. Kim, "Design, modeling, and test of a programmable adaptive phase-shifting PLL for enhancing clock data compensation," *IEEE J. Solid-State Circuits*, vol. 47, no. 10, pp. 2505–2516, Oct. 2012.
- [3] Y. YangGong *et al.*, "Asymmetric frequency locked loop (AFLL) for adaptive clock generation in a 28 nm SPARC M6 processor," in *Proc. IEEE Asian Solid-State Circuits Conf.*, Nov. 2014, pp. 373–376.
- [4] C. R. Lefurgy *et al.*, "Active management of timing guardband to save energy in POWER7," in *Proc. 44th Annu. IEEE/ACM Int. Symp. Microarchitecture (MICRO)*, Dec. 2011, pp. 1–11.
- [5] K. Wilcox *et al.*, "Steamroller module and adaptive clocking system in 28 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 50, no. 1, pp. 24–34, Jan. 2015.
- [6] C. Takahashi *et al.*, "A 16 nm FinFET heterogeneous nona-core SoC complying with ISO26262 ASIL-B: Achieving 10–7 random hardware failures per hour reliability," in *IEEE ISSCC Dig. Tech. Papers*, Jan. 2016, pp. 80–81.
- [7] M. S. Floyd *et al.*, "Adaptive clocking in the POWER9 processor for voltage droop protection," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2017, pp. 444–445.
- [8] T. Hashimoto *et al.*, "An adaptive clocking control circuit with 7.5% frequency gain for SPARC processors," in *Symp. VLSI Circuits Tech. Dig.*, Jun. 2017, pp. C112–C113.
- [9] S. S. Mohan, W. S. Chan, D. M. Colleran, S. F. Greenwood, J. E. Gamble, and I. G. Kouznetsov, "Differential ring oscillators with multipath delay stages," in *Proc. IEEE Custom Integr. Circuits Conf.*, Sep. 2005, pp. 503–506.
- [10] M. Z. Straayer and M. H. Perrott, "A multi-path gated ring oscillator TDC with first-order noise shaping," *IEEE J. Solid-State Circuits*, vol. 44, no. 4, pp. 1089–1098, Apr. 2008.
- [11] T. Hashimoto, H. Yamazaki, A. Muramatsu, T. Sato, and A. Inoue, "Time-to-digital converter with Vernier delay mismatch compensation for high resolution on-die clock jitter measurement," in *Symp. VLSI Circuits Tech. Dig.*, Jan. 2008, pp. 166–167.



Tetsutaro Hashimoto received the B.E. degree in electronic engineering and the M.E. degree in communications and computer engineering from Kyoto University, Kyoto, Japan, in 1998 and 2000, respectively.

In 2000, he joined Fujitsu Laboratories Limited, Kawasaki, Japan, where he had been involved in signal and power supply integrity analysis and related design methodologies. Since 2007, he has been involved in the circuit designs of on-die measurement macros and on-die LDO regulators for power supply noise suppression in digital integrated systems. His current research interests include power supply noise mitigation by adaptive-clocking control for high-performance processors.



Yukihito Kawabe received the M.E. degree in electronic engineering from the Tokyo Institute of Technology, Japan, in 1993.

In 1993, he joined Fujitsu Laboratories Limited, Atsugi, Japan, where he had been involved in the development of media processors and low-power design techniques. His current research interests include power management for high-performance processors.



Michiharu Hara joined Fujitsu Limited, Kawasaki, Japan, in 1987, where he had been involved in the architecture and circuit design of instruction units in UNIX, HPC, and mainframe processors. He is currently involved in the development of power management system for SPARC processors.



Yasushi Kakimura received the B.E. degree in electronic engineering from Chiba University, Chiba, Japan, in 1990.

In 1990, he joined Fujitsu Limited, Kawasaki, Japan, where he had been involved in the development of standard cells for UNIX and HPC processors. He is currently involved in the development of custom macros and ADPLL for SPARC processors.



Kunihiko Tajiri received the B.E. degree in electronic and information engineering from the University of Tokyo, Tokyo, Japan, in 1995.

In 1995, he joined Fujitsu Limited, Kawasaki, Japan, where he is currently a Manager of the Technology Development Division. He is currently involved in the development and circuit design of custom macros in execution unit and ADPLL for SPARC processors.



Shinichiro Shirota received the B.E. and M.E. degrees in applied physics from Hokkaido University, Sapporo, Japan, in 2006 and 2008, respectively.

In 2008, he joined Fujitsu Limited, Kawasaki, Japan, where he has been involved in high-speed interconnect design and analog circuit design for enterprise server systems.



Ryuichi Nishiyama received the B.S. and M.S. degrees in electronic engineering from the University of Electro-Communications, Tokyo, Japan, in 1995 and 1997, respectively.

In 1997, he joined Fujitsu Limited, Kawasaki, Japan, where he has been involved in analog circuit design. He is currently involved in the development of high-speed serial link.



Yasumoto Tomita received the B.S., M.S., and Ph.D. degrees in electrical engineering from Keio University, Yokohama, Japan, in 2002, 2004, and 2007, respectively.

He was with Fujitsu Laboratories Limited, Kawasaki, Japan, where he has been a Research Manager of the Computer Systems Laboratory and involved in the circuit design of high-speed CMOS I/O and the development of artificial intelligent computing system.

Dr. Tomita served as a Technical Program Committee Member for A-SSCC and VLSI Symposium on Circuits.



Hitoshi Sakurai received the B.E. degree in electronic and information engineering from Yokohama National University, Yokohama, Japan, in 1990.

In 1990, he joined Fujitsu Limited, Kawasaki, Japan, where he is currently a Manager of the Processor Development Division. He had been involved in the architecture and circuit design of the second Cache Unit in mainframe and UNIX processors. He is currently involved in the chip logic design and chip implementation of mainframe and SPARC processors.



Sugio Satoh received the B.E. and M.E. degrees in electronic engineering from Nihon University, Tokyo, Japan, in 1985 and 1987, respectively.

In 1987, he joined Fujitsu Limited, Kawasaki, Japan, where he is currently an Expert of the Technology Development Division. He had been involved in the architecture and circuit design of vector processing unit in minisupercomputer processors, and the development of mainframe processors. Since 1997, he has been involved in the development of custom macros and related design methodologies for enterprise server processors.



Hiroshi Okano received the B.E. and M.E. degrees in electrical engineering from Hiroshima University, Hiroshima, Japan, in 1990 and 1992, respectively.

In 1992, he joined Fujitsu Laboratories Limited, Atsugi, Japan, where he had been involved in high-performance techniques for microprocessors under power constraints. Since 2016, he has been a Director with the AI Platform Division, Fujitsu Limited, Kawasaki, Japan. His current research interests include service-oriented technologies for microprocessors.



Hideo Yamashita received the bachelor's degree from the Department of Physics, Faculty of Education, Kobe University, Kobe, Japan, in 1989.

In 1989, he joined Fujitsu Limited, Kawasaki, Japan, where he is currently the Director of the Processor Development Division. He is involved in the development of mainframe computer hardware, and designed execution unit of the CPU, and then he has also developed the SPARC processors and supercomputer processors.