CMOS Annealing Machine: A Domain-Specific Architecture for Combinatorial Optimization Problem

Chihiro Yoshimura*, Masato Hayashi*, Takashi Takemoto†, and Masanao Yamaoka*

*Center for Technology Innovation–Electronics
Research and Development Group, Hitachi, Ltd.
Tokyo, Japan 185-8601

†Center for Exploratory Research
Research and Development Group, Hitachi, Ltd.
Hokkaido, Japan 001-0021

E-mail: {chihiro.yoshimura.ak, masato.hayashi.fo, takashi.takemoto.tj, masanao.yamaoka.ns}@hitachi.com

Abstract—Domain-specific architectures are being studied to improve computer performance beyond the end of Moore's Law. Here, we propose a new computing architecture, the CMOS annealing machine, which provides a fast means of solving combinatorial optimization problems. Our architecture is based on in-memory computing architecture through utilizing the locality of interactions in the Ising model. The prototype presented in 2019 has two processors on a business-card-sized board and solves problems 55 times faster than conventional computers.

I. Introduction

Advances in semiconductor manufacturing processes have improved performance in computers based on the widely used von Neumann architecture [1]. Denard scaling, which was the driving force, has already broken down, and Moore's Law has come to an end. New computer architectures that further improve computer performance in the post-Moore era are being studied. Among the various architectures proposed, one common term is that the next generation architecture will be a "domain-specific architecture" [2].

Domain-specific architectures are for a specific class of applications and use the knowledge of the application to realize efficiency. For example, parallelization can be performed using the nature of the problem to be solved, efficient use of the memory hierarchy, and selection of appropriate calculation accuracy. On the other hand, the biggest advantage of the conventional architecture, versatility, is reduced. Therefore, these architectures are positioned as accelerators for processors based on conventional architectures.

The combinatorial optimization problem has applications such as scheduling, resource allocation, and route searches. Finding the global optimization of combinatorial optimization problems is often an NP-hard problem. The Ising model can express the behavior of magnetic spins, and it consists of n binary spins $\{\sigma_1, \dots, \sigma_n\}$, interactions $\{J_{ij}\}$ between spins, and external magnetic fields $\{h_i\}$, as shown in Fig. 1 (a) [3]. The Ising model's energy is defined as

$$H(\sigma_1, \cdots, \sigma_n) = -\sum_{i<j} J_{ij}\sigma_i\sigma_j - \sum_i h_i\sigma_i. \tag{1}$$

The energy varies with the spin configurations, and it can be visualized as an energy landscape, as shown in Fig. 1 (b). Finding a spin configuration that minimizes the energy of the Ising model with a nonplanar graph topology is an NP-hard problem [4]. The combinatorial optimization problems can be mapped into the Ising model [5]. The Ising models' energy corresponds to the objective function of combinatorial optimization problems, meaning the spin configuration with the lowest energy represents the global optimum solution of the original problem.

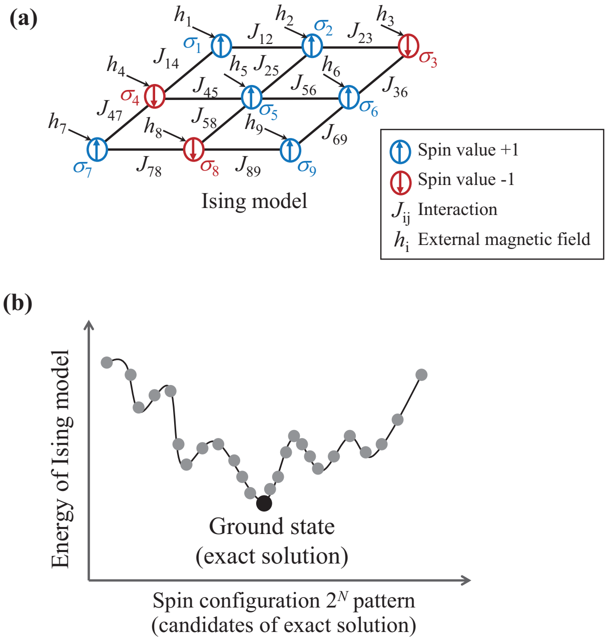


Fig. 1. (a) Ising model and (b) its energy landscape.

Finding the global optimum solution to the NP-hard problem generally requires exponential time. Approximation algorithms can find relatively better local optimum solutions and are used to solve the problem in practical time. Simulated annealing is a probabilistic algorithm inspired from annealing in metallurgy and is widely used to approximate global optimization [6]. Many architectures that use the Ising model have been proposed as domain-specific architectures for combinatorial optimization problems, most being dedicated hardware that efficiently implements simulated annealing and its variants to find the spin configuration of the Ising model.

We developed a domain-specific computer CMOS annealing machine for the combinatorial optimization problem by annealing the Ising model implemented as the widely used

978-1-7281-4123-7/20/$31.00 © 2020 IEEE

CMOS integrated circuit.

II. CMOS Annealing Machine

A. Basic Architecture

The CMOS annealing machine is dedicated hardware whose basic architecture we proposed in 2013 [7]; it simulates the structure of the Ising model and searches for $\{\sigma_1, \ldots, \sigma_n\}$ that minimizes H in Eq. 1 by annealing. The spin and accompanying coefficients are grouped into a spin unit, as shown in Fig. 2. Each spin unit has a memory cell array to represent the spin and the coefficients. The next state of spin is determined by a digital or analog operator. The spin units are connected according to the topology of the desired Ising model. The coefficient values that represent the optimization problem are written in memory cells to solve the problem using the processor. Thus, the interaction between spin units is executed repeatedly by updating the spin value. The operator takes neighboring spin values and corresponding coefficients to determine the next state of spin to minimize the energy locally. An update means that the latest output of the operator is stored in the memory cell that represents the spin, then the output can be observed from neighboring spin units.

Figure 3 shows the flow to solve the combinatorial optimization problem by use of CMOS annealing machine. This processor acts like an accelerator attached to a conventional computer system (e.g., personal computer) for the combinatorial optimization problem. In general, a quadratic unconstrained binary optimization (QUBO) problem can be expressed using an Ising model. First, QUBO's cost function is formulated as an Ising model. The components of the Ising model are the interaction coefficient $\{J_{ij}\}$ between spins and the external magnetic field coefficient $\{h_i\}$ for each spin. Therefore, these two types of coefficients are input into the annealing machine. The CMOS annealing machine searches the spin configuration $\{\sigma_1, \ldots, \sigma_n\}$ that corresponds to the lower energy state under the given coefficients. The spin configuration corresponds to the solution of the original optimization problem which is the source that formulates the Ising model. Users can know the solution of the combinatorial optimization problem by reading the spin values on the CMOS annealing machine after finishing the annealing process.

B. First-Generation Prototype Chip

We implemented a prototype based on this architecture and presented it at the ISSCC 2015 [8]. The prototype chip is fabricated in the 65-nm process shown in Fig. 4 (a). The chip size is $4\text{mm} \times 3\text{mm} = 12\text{mm}^2$, and it contains 20480 spins three-dimensional lattice Ising model $(128 \times 80 \times 2)$. The chip design is based on the processing-in-memory approach, in which logic circuits and SRAMs are tightly coupled. Each spin unit, shown in Fig. 4 (b), has 1-bit memory cell for storage of the spin, 13-bits memory cell for the storage of coefficients, and an operator to determine the next state of the spin. Arithmetical operation is realized as analog behavior of the CMOS circuit. The whole chip is designed by repetition of spin units. Each of them needs data transfer between host system. This prototype chip has bit lines and word lines as same as SRAM chip to realize random access for entire memory cells in the chip.

To evaluate this prototype's performance, we solved the randomly generated maximum cut problem with the prototype and a conventional computer. The maximum cut problem is an NP-hard combinatorial optimization problem. The prototype solved the approximate solution of the maximum cut problem of a graph that consisted of 20480 vertices in 1.5 ms. The approximation algorithm for the maximum cut problem on the conventional computer found an approximate solution of the same quality in 15 ms. The prototype was 1800 times more energy efficient than the conventional computer in solving the same problem with the same quality [9].

In solving the optimization problem by annealing, random numbers are used as equivalent to heat. In other words, when the annealing processor is designed, the quantity of the random number generator becomes dominant. To increase scalability, we reduce the amount of random number generators, implementing a method of supplying random numbers required by the spin unit of the whole chip with a small number of random number generators by utilizing asynchronous wirings that extend over the whole chip [10].

In the future, we may consider using the uncertain behavior of the device (caused by semiconductor miniaturization) as the randomness required for annealing. For this purpose, we also used this prototype to show the possibility of solving optimization by using uncertainty in devices [11].

III. Prototype for Edge Computing

At ISSCC2019, we unveiled an improved second-generation prototype chip with an edge-ready business-card-sized node, shown in Fig. 5 [12]. The prototype chip is fabricated in the 40-nm process shown in Fig. 5 (b). In the second-generation prototype, the rules for updating the spin value were changed to improve the accuracy of the solution over the first generation. In addition, the topology of the Ising model supported by the prototype is called the "King's graph," a topology in which a diagonal line is added to a two-dimensional grid. One chip supports 30976 (176×176) spins.

The biggest difference from the first-generation prototype is the ability to connect multiple chips to support a larger Ising model. For this purpose, each chip has an LVDS interface that connects the chips, shown as Fig. 5 (b). With this interface, two chips were connected to support 61952 (352×176) spins in the node shown in Fig. 6. In the post-Moore era, an increased number of transistors per unit area cannot be expected. To improve performance beyond this limitation, we must provide scalable interconnect to use multiple chips together, and the second-generation prototype showed the scalability by use of the implemented interconnect.

A node with two of these chips is a small footprint $(91\text{mm}\times 55\text{mm})$ that can be applied to the edge of an IoT system and is the same size as a business card, as shown in Fig. 5 (a). This node is connected to a host personal computer via USB and exchanges Ising model coefficients and spin values. The node operates through a USB power supply. The only components mounted on the board are two prototype chips, a power supply circuit, a FPGA-based controller, and a USB communication circuit.

The second-generation prototype is 455 times faster than the first-generation prototype. The prototype's energy efficiency

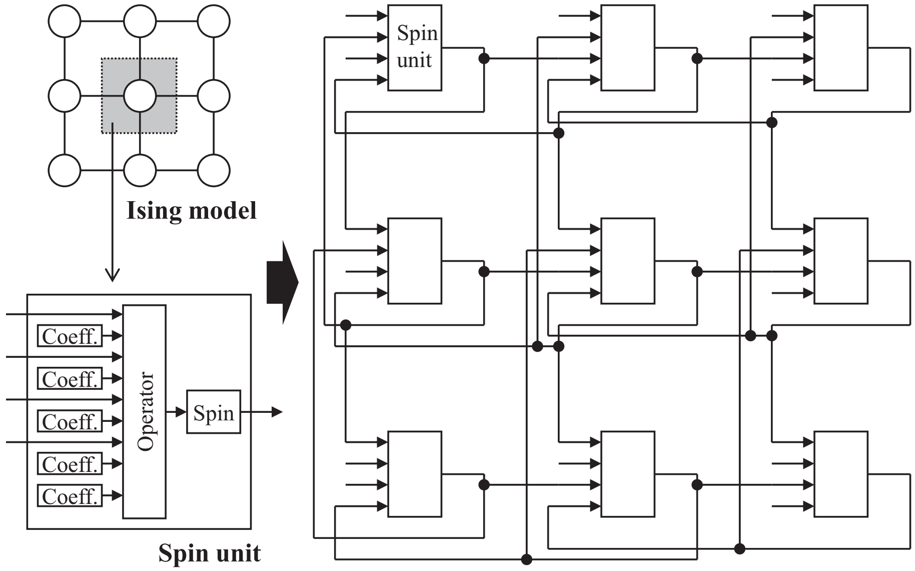


Fig. 2. Basic architecture of the CMOS annealing machine. The spin of the Ising model and the associated coefficient are configured as one spin unit. A large number of spin units are arranged and wired to reproduce the topology of the Ising model to be simulated.

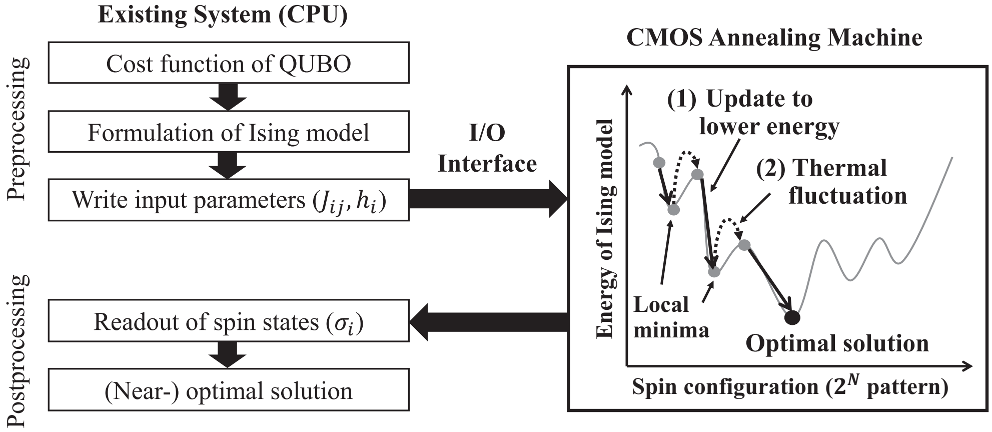


Fig. 3. Flow of solving the combinatorial optimization problem using the CMOS annealing machine. The CMOS annealing machine functions as an accelerator specialized for combinatorial optimization problems. A conventional computer controls the entire flow.

is 1.75×10^5 times better than a conventional computer in finding the approximate solution of the maximum cut problem. Aside from the maximum cut problem, we also evaluated the performance of the minimum vertex cover problem, which is an NP-hard combinatorial optimization problem. Similar to the performance evaluation for the maximum cut problem, a heuristic algorithm specialized for the minimum vertex cover problem was executed on a conventional computer and used for comparison. Although the relative solution accuracy varied, the second-generation prototype achieved a 3–7% better solution accuracy over the conventional computer. The prototype solved the largest problem, which contained 61952 vertices in $206\mu\mathrm{sec}$, at 55 times the conventional speed [13].

The I/O time is a large bottleneck. The CMOS annealing machine's input—the host computer writing coefficients into the annealing machine—took 22.6 msec. The machine's output—the host computer reading spin values from annealing machine—took 1.5 msec. The total I/O time was 24.1 msec, relatively larger than the computation time. To avoid this overhead, the CPU in the conventional computer and the annealing machine should be tightly coupled rather than a current I/O interconnect such as USB and PCI Express.

IV. FPGA-based Prototype and its Cloud Operation

The CMOS annealing machine can also be configured as a digital circuit on the FPGA. Therefore, we verified the architecture using an FPGA in the process from the first-generation prototype to the second-generation prototype [14]. Design space exploration in the development of CMOS annealing machines is mainly determining the topology and coefficient accuracy of the Ising model to be supported. Although the fully-connected topology and the coefficient width of double or more precision floating point numbers is ideal, it is in a trade-off relationship with the number of spins and power consumption. The co-design of application and architecture is needed to determine the most efficient configuration for the

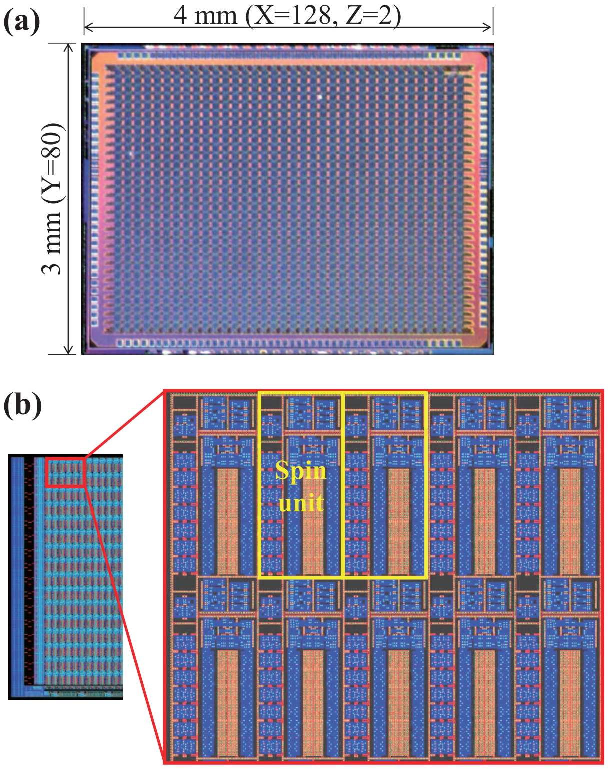


Fig. 4. (a) Die photograph of the first-generation CMOS annealing machine and (b) its detailed structure. The spin unit processing element handles single spins and accomplished interactions, and the entire chip is designed by repetition of spin units like a tiled structure.

architecture. We developed the FPGA-based cloud environment to collaborate with customers as shown in Fig. 7.

The FPGA-based prototype we developed contains 25 Xilinx Virtex UltraScale XCVU095 FPGAs. The 25 FPGAs are connected by a 5×5 two-dimensional torus topology interconnect as shown in Fig. 8. This interconnect is responsible for the transfer of spin values and the data transfer necessary for overall control, such as register settings in the FPGA. These FPGAs behave like one large FPGA with this interconnect. One of the 25 FPGAs is connected to the host server via PCI Express. Viewed from the host server, it appears as one PCI Express device consisting of 25 FPGAs.

This cloud environment is installed in our data center and provided to customers via the internet, providing a job management system that shares a single machine with multiple users and can be used via REST API. Users can use both this cloud environment and edge-ready nodes in a Python-based development environment.

V. Related Work

Many special-purpose computers have been proposed for the Ising model. These computers can be categorized by two aspects: device technology and purpose.

As for device technology, some studies aim to utilize novel devices to further improve performance over conventional electronics. A quantum annealing machine that uses superconducting devices to solve combinatorial optimization problems using quantum mechanical superposition has also been proposed [15]. However, maintaining the quantum state during the computing process is difficult. Recent studies have used the stochastic behavior of magnetic tunnel junctions instead of qubits in a quantum annealing machine [16].

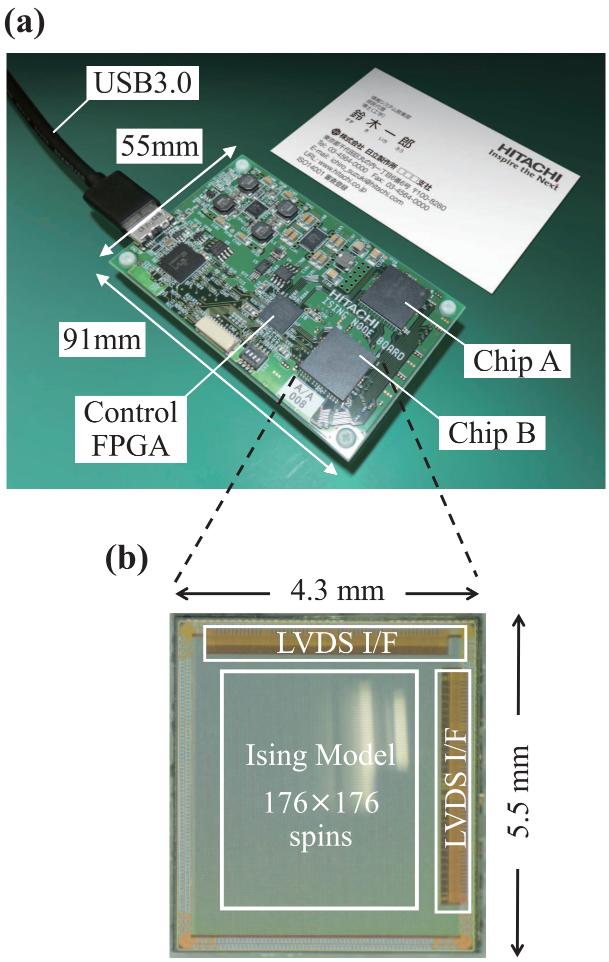


Fig. 5. (a) Photograph of a business-card–sized edge-ready CMOS annealing machine node implementing two chips. The node is connected to the host computer via a USB cable and powered via USB. (b) Die photograph of a second-generation CMOS annealing machine containing the processing elements and LVDS interfaces that connect multiple chips to enhance the capability of a larger problem.

As for purpose, these computers can be divided into two categories: Monte Carlo simulation of the spin system [17], [18], [19] and solving combinatorial optimization problems.

Our study can be categorized as a CMOS-based device and combinatorial optimization applications. Many extensions to our architecture and similar architectures have been proposed [20], [21], [22], [23]. These studies commonly use CMOS digital circuits and memory cells. That is, the memory cell holds the value of the spin, and the arithmetic circuit realizes the interaction between the spins. On the other hand, a method based on an analog circuit that expresses an Ising model with a collection of LC oscillators with MOS transistors has also been proposed [24], [25]. In the analog circuit system, the spin value is expressed by the phase of the oscillator. Injection locking realizes the interaction between spins.

VI. Conclusion

We developed prototype CMOS annealing machines and confirmed that the combinatorial optimization problems could

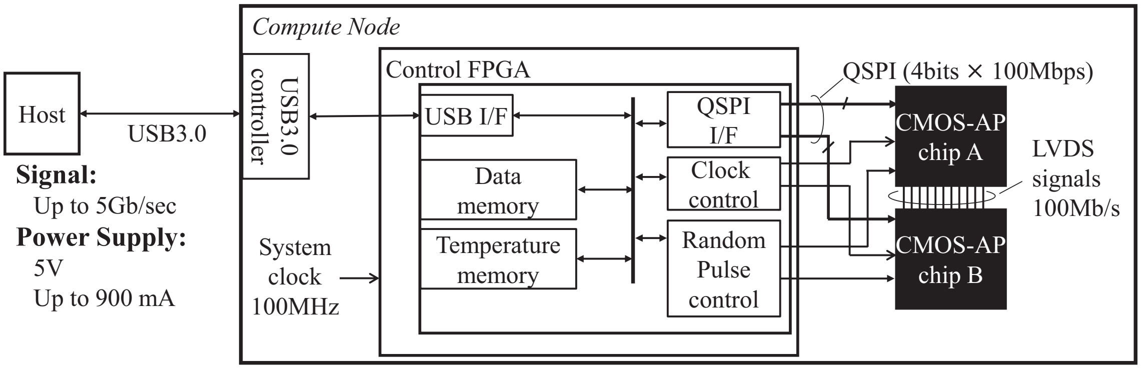
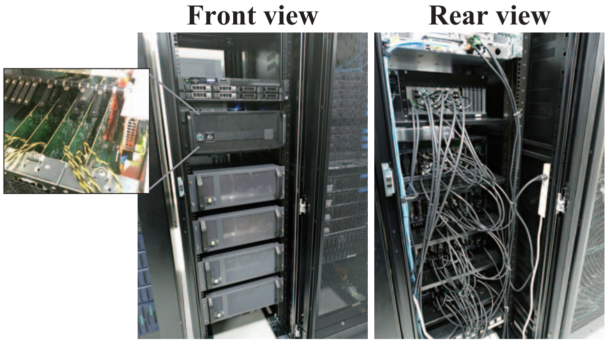


Fig. 6. Block diagram of a business-card-sized edge-ready CMOS annealing machine node.


Front view

Rear view

Fig. 7. Photograph of the FPGA-based large-scale prototype machine installed in our datacenter. Multiple users can share this prototype via the internet.

be solved in practice. The first-generation prototype manufactured by the 65-nm process supports 20480 spins and is 1800 times more energy efficient than conventional computers in solving the maximum cut problem.

The second-generation prototype manufactured by the 40-nm process integrated 30976 spins and connected multiple chips to support larger problems. A business-card–sized board with two chips supported 61952 spins. The energy efficiency is even better than the first-generation prototype and is 1.75×10^5 times higher in the maximum cut problem. Performance evaluation depends on the problem to be compared and the selection of algorithms for it. For example, the second-generation prototype solves the minimum vertex cover problem 55 times faster than the heuristic algorithm specialized for the minimum vertex cover problem running on a conventional computer.

In addition, we developed a large-scale FPGA cloud environment that uses 25 FPGAs, and we promote co-creation with customers by providing a cloud environment of CMOS annealing machines configured on the FPGA. Diverting all software stacks cultivated in conventional computers as they are is difficult for both CMOS annealing machines and non-Neumann computers, and a new software stack must be constructed. Advances in conventional computers have been realized by combining large systems (such as those in the cloud and data centers) and small systems (that can be incorporated into devices such as microcontrollers). We will continue to develop CMOS annealing machines for both cloud and edge environments.

Acknowledgments

Part of this study is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

References

[1] K. Rupp, “42 years of microprocessor trend data,” 2018. [Online]. Available: https://github.com/karlrupp/microprocessor-trend-data.

[2] J. L. Hennessy and D. A. Patterson, “A new golden age for computer architecture,” *Communications of the ACM*, vol. 62, no. 2, pp. 48–60, 2019.

[3] S. B. Brush, “History of the Lenz-Ising model,” *Reviews of Modern Physics*, vol. 39, no. 4, pp. 883–893, 1967.

[4] F. Barahona, “On the computational complexity of Ising spin glass models,” *Journal of Physics A: Mathematical and General*, vol. 15, no. 10, pp. 3241–3253, 1982.

[5] A. Lucas, “Ising formulations of many NP problems,” *Frontiers in Physics*, vol. 2, no. 5, pp. 1–15, 2014.

[6] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, “Optimization by simulated annealing,” *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[7] C. Yoshimura, M. Yamaoka, H. Aoki, and H. Mizuno, “Spatial computing architecture using randomness of memory cell stability under voltage control,” 2013 European Conference on Circuit Theory and Design (ECCTD), 2013.

[8] M. Yamaoka, C. Yoshimura, M. Hayashi, T. Okuyama, H. Aoki, and H. Mizuno, “20k-spin Ising chip for combinational optimization problem with CMOS annealing,” 2015 IEEE International Solid-State Circuits Conference (ISSCC) Digest of Technical Papers, 2015.

[9] M. Yamaoka, C. Yoshimura, M. Hayashi, T. Okuyama, H. Aoki, and H. Mizuno, “A 20k-spin Ising chip to solve combinatorial optimization problems with CMOS annealing,” *IEEE Journal of Solid-State Circuits*, vol. 51, no. 1, pp. 303–309, 2016.

[10] M. Hayashi, M. Yamaoka, C. Yoshimura, T. Okuyama, H. Aoki, and H. Mizuno, “Accelerator chip for ground-state searches of Ising model with asynchronous random pulse distribution,” *International Journal of Networking and Computing*, vol. 6, no. 2, pp. 195–211, 2016.

[11] C. Yoshimura, M. Yamaoka, M. Hayashi, T. Okuyama, H. Aoki, K. Kawarabayashi, and H. Mizuno, “Uncertain behaviours of integrated circuits improve computational performance,” *Scientific Reports*, vol. 5, no. 16213, 2015.

[12] T. Takemoto, M. Hayashi, C. Yoshimura, and M. Yamaoka, “A 2 × 30k-spin multichip scalable annealing processor based on a processing-in-memory approach for solving large-scale combinatorial optimization problems,” 2019 IEEE International Solid- State Circuits Conference (ISSCC), pp. 52–54, 2019.

[13] M. Hayashi, T. Takemoto, C. Yoshimura, and M. Yamaoka, “A Cloud-ready scalable annealing processor for solving large-scale combinatorial optimization problems,” 2019 Symposium on VLSI Technology, pp. C148–C149, 2019.

[14] C. Yoshimura, M. Hayashi, T. Okuyama, and M. Yamaoka, “Implementation and evaluation of FPGA-based annealing processor for Ising model by use of resource sharing,” *International Journal of Networking and Computing*, vol. 7, no. 2, pp. 154–172, 2017.

[15] M. W. Johnson et al., “Quantum annealing with manufactured spins,” *Nature*, vol. 473, pp. 194–198, 2011.

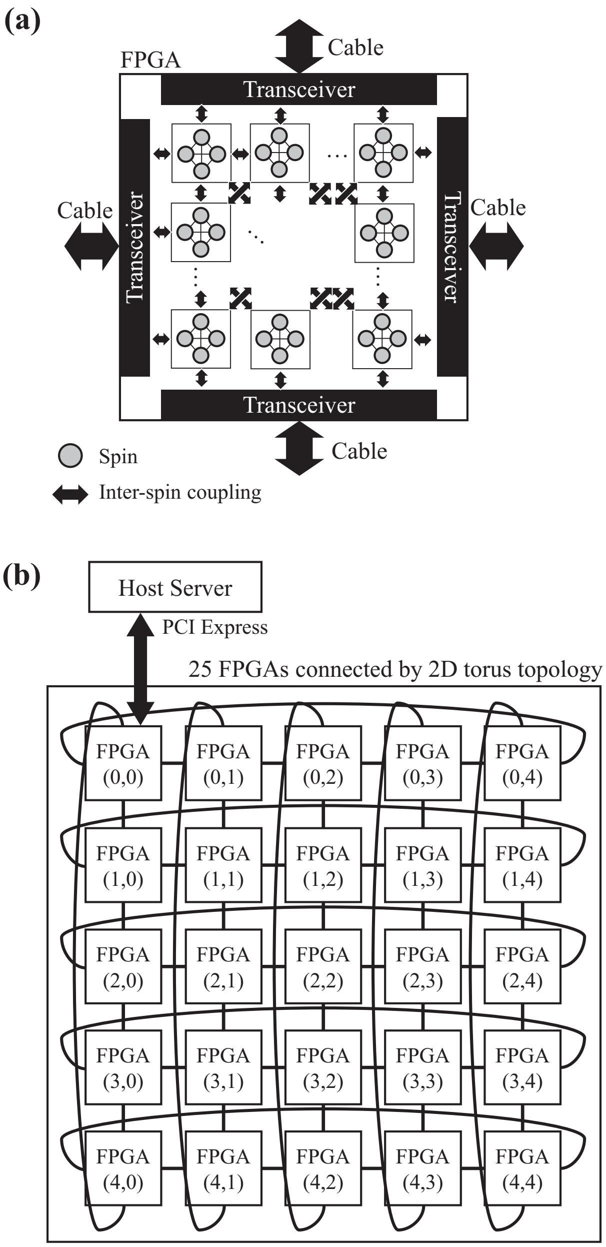


Fig. 8. The detailed structure of the FPGA-based large-scale prototype machine. (a) Diagram of the FPGA and (b) topology of connection between FPGAs.

[16] W. A. Borders, A. Z. Pervaiz, S. Fukami, K. Y. Camsari, H. Ohno, and S. Datta, “Integer factorization using stochastic magnetic tunnel junctions,” *Nature*, vol. 573, pp. 390–393, 2019.

[17] F. Belletti et al., “Janus: an FPGA-based system for high-performance scientific computing,” *Computing in Science Engineering*, vol. 11, no. 1, pp. 48–58, 2009.

[18] A. Gilman, A. Leist, and K. A. Hawick, “3D lattice Monte Carlo simulations on FPGAs,” Proceedings of the International Conference on Computer Design (CDES), 2013.

[19] Y. Lin, F. Wang, X. Zheng, H. Gao, and L. Zhang, “Monte Carlo simulation of the Ising model on FPGA,” *Journal of Computational Physics*, vol. 237, pp. 224–234, 2013.

[20] K. Someya, R. Ono, and T. Kawahara, “Novel Ising model using dimension-control for high-speed solver for Ising machines,” 2016 14th IEEE International New Circuits and Systems Conference (NEWCAS), 2016.

[21] J. Zhang, S. Chen, and Y. Wang, “Advancing CMOS-type Ising arithmetic unit into the domain of real-world applications,” *IEEE Transactions on Computers*, vol. 67, no. 5, pp. 604–616, 2018.

[22] H. Gyoten, M. Hiromoto, and T. Sato, “Enhancing the solution quality of hardware Ising-model solver via parallel tempering,” in Proceedings of IEEE/ACM International Conference on Computer-Aided Design (ICCAD), 2018.

[23] H. Gyoten, M. Hiromoto, and T. Sato, “Area efficient annealing processor for Ising model without random number generator,” *IEICE Transactions on Information and Systems*, vol. E101.D, no. 2, pp. 314–323, 2018.

[24] T. Wang and J. Roychowdhury, “Oscillator-based Ising machine,” arXiv:1709.08102, 2017.

[25] J. Chou, S. Bramhavar, S. Ghosh, and William Herzog, “Analog coupled oscillator based weighted Ising machine,” *Scientific Reports*, vol. 9, no. 14786, 2019.