
Autonomous, dynamic, and suitable for a variety of real-world applications.

EVOLVABLE HARDWARE CHIPS *for* INDUSTRIAL APPLICATIONS

TETSUYA HIGUCHI AND NOBUKI KAJIHARA

In contrast to conventional hardware, in which the structure is irreversibly fixed in the design process, evolvable hardware is designed to adapt, as the chameleon changes its color to blend in with the environment, to changes in task requirements or in the environment, through its ability to reconfigure its own hardware structure dynamically and autonomously. This capacity for adaptation, achieved by employing efficient search algorithms known as genetic algorithms, has great potential for the development of innovative industrial applications. Although the concept of EHW is relatively new, some EHW chips are already being evaluated for their commercial value. In this article, we introduce four



EHW chips currently being developed as part of MITI's Real-World Computing Project; an analog EHW chip for cellular phones, a neural network EHW chip capable of autonomous reconfiguration, a data compression EHW chip for electrophotographic printers, and an adaptive control EHW chip for use in prosthetic hands and robot navigation.

Basic Concepts

Evolvable hardware is based on the idea of combining reconfigurable hardware devices with genetic algorithms to execute reconfiguration autonomously [2, 8]. The structure of a reconfigurable hardware device can be changed any number of times by downloading into the device a software bit string called configuration bits. Field-programmable gate array (FPGA) and programmable logic devices (PLD) are typical examples of reconfigurable hardware devices, for which there is already a market worth more than \$2 billion and growing at 23% per year. However, it should be noted that the reconfiguration must still be executed manually by hardware designers.

A genetic algorithm (GA) is a robust search algorithm loosely based on population genetics [1]. It effectively seeks solutions from a vast search space at reasonable computation costs. Before a GA starts, a set of candidate solutions, represented as binary bit strings, is prepared. This set is referred to as a population, and each candidate solution within the set as a chromosome. A fitness function is also defined that represents the problem to be solved in terms of criteria to be optimized. The chromosomes then undergo a process of evaluation, selection, and reproduction. In the evaluation stage, the chromosomes are tested according to fitness function. The results of this evaluation are then used to weight the random selection of chromosomes for the final stage of reproduction in favor of the fitter ones. In this final stage, a new generation of the chromosomes are "evolved" through genetic operations that attempt to pass desirable characteristics to the next generation. Through this process, which can be repeated as many times as required, less-fit chromosomes are gradually expelled from a population and the fitter chromosomes become more likely to emerge.

The basic concept behind the combination of these two elements in EHW is to regard the configuration bits for reconfigurable hardware devices as

chromosomes for genetic algorithms (See Figure 1). If a fitness function is properly designed for a task, then the genetic algorithms can autonomously find the best hardware configuration in terms of chromosomes (that is, configuration bits).

For example, in data compression with EHW we use a prediction function. Optimal prediction functions vary greatly according to the different kinds of data to be compressed. It is, therefore, not possible to design a prediction hardware function in advance. Instead of specifying a detailed hardware design, we define a fitness function. In the case of data compression, the data compression rate is used as a fitness function. Accordingly, a circuit of prediction function with a higher data compression rate is likely to remain in a population. When a good chromosome is obtained, it is immediately downloaded into the reconfigurable device.

If the prediction performance of a given EHW is reduced due to changes in the nature of the data to be compressed, then the GA process is invoked and the search for a better prediction hardware structure is initiated. In this way, EHW is capable of both autonomous and dynamic hardware reconfigurations. In the remainder of this article we show how this characteristic of EHW is utilized in four EHW chips and their applications.

Analog EHW Chip for Cellular Phone

Due to the remarkable advances in recent CPU and digital signal processor (DSP) technology, applications with analog circuits are rapidly being replaced by digital computing. However, there are still many applications that require high-speed analog circuits. Communication is one such application.

However, an inherent problem in implementing analog circuits is that the values of the manufactured analog circuit components, such as resistors and capacitors, will often differ from the precise design specifications. Such discrepancies cause serious problems for high-end analog circuit applications. For example, in intermediate frequency (IF) filters, which are widely used in cellular phones, even a 1% discrepancy from the center frequency is unacceptable. It is therefore necessary to carefully examine the analog circuits and to discard any that do not meet the specifications.

The analog EHW chip for IF filters can correct these variations in the analog circuits values by

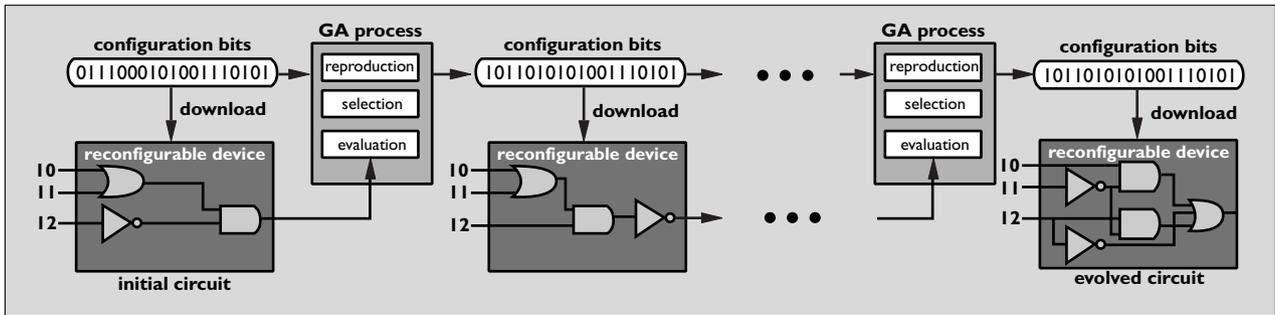


Figure 1. Basic concept of evolvable hardware.

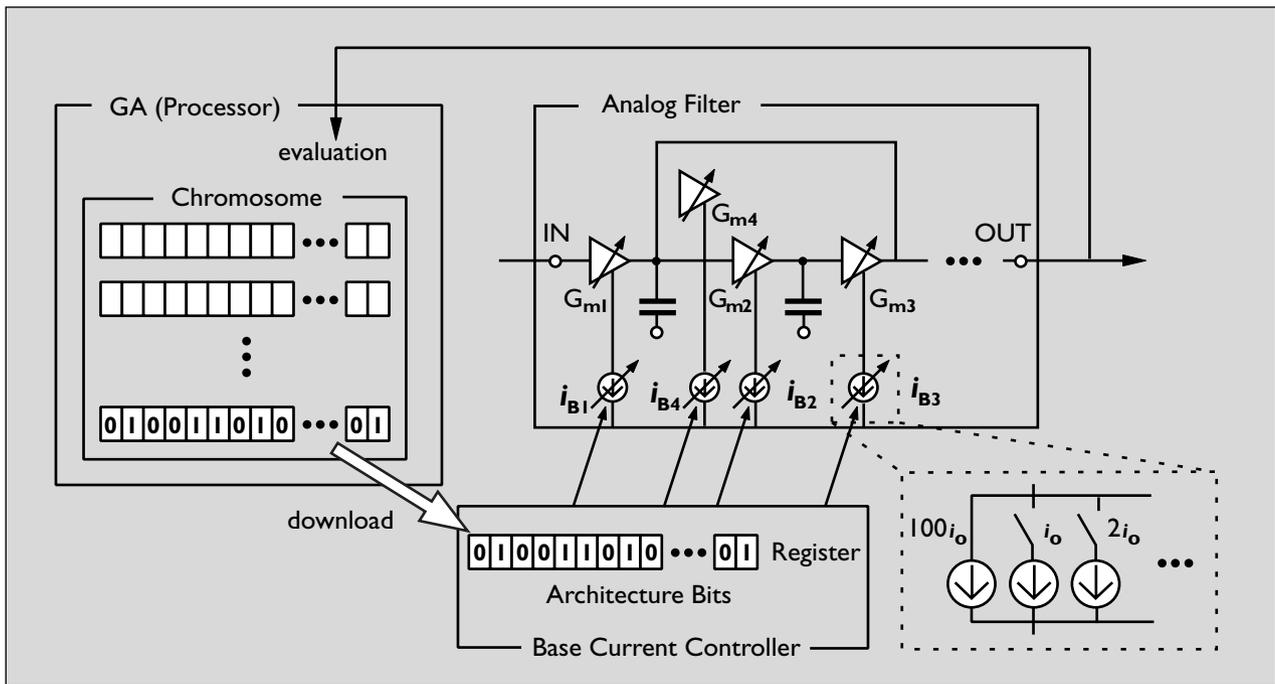


Figure 2. Basic idea of the analog EHW chip for IF filters.

genetic algorithms [6]. Using this chip provides us with two advantages:

- **Improved yield rate.** When analog EHW chips are shipped that do not satisfy specifications, then the GA can be executed to alter the defective analog circuit components to conform to specifications. The GA is supposed to be executed in the LSI tester.
- **Smaller circuits.** One way to increase the precision of component values in analog LSIs has been to use large valued analog components. However, this involves larger circuits, and accordingly higher manufacturing costs and greater power consumption. With the EHW chip, however, the analog circuits can be made smaller. Obviously, smaller IF filters are particularly welcome in cellular phones, but similar considerations exist in a wide variety of

applications in which analog circuits are used.

Figure 2 illustrates the analog EHW chip. The chip includes 39 G_m components (transconductance amplifiers) whose values can be set genetically. The values, which actually control the base current of the CMOS, are coded as configuration bits. Each G_m element value may differ from the target value up to a maximum of 20%. Initial simulations have shown that 95% of the chips can be corrected to satisfy the IF filter's specifications. The chip was manufactured in September 1998.

Data Compression Chip for Printing

Electrophotographic (EP) printing is the latest generation technology in the printing and publishing industry, which makes it possible to print books with high-precision photo quality. Data compression devices are essential for the design of EP print-

The structure of a reconfigurable hardware device can be changed any number of times by downloading into the device a software bit string called configuration bits.

ers, which handle large amounts of data very quickly. For example, one 1200-dpi A4-size EP image requires 70MB for storage, and EP printers processes hundreds of different pages at a speed of 100 pages/min. (Typical color copiers can print less than 10 pages/min.) This means that to print a book with 100 pages, 7GB of image data must be transferred to the printer at a speed of 1800MB/min. Unfortunately, the data transfer rate of typical hard drives is only 300MByte/min. EP printers, therefore, have to employ data compression techniques (1) to compress image data efficiently, and (2) to reconstruct the compressed data

mented by EHW. Using the GA, an optimal prediction function is found and the hardware prediction mechanism is reconfigured accordingly. This leads not only to improved compression rates but also to higher decompression speed, because decompression is also carried out by the EHW hardware. The chip includes a 32-bit RISC processor hardware that executes the genetic algorithm to obtain the optimal hardware prediction.

Table 1 is the comparison with two major international standards for data compression; Lempel-Ziv (“compress” command of Unix) and JBIG (Joint Bi-level Image coding experts Group), both are available on LSI chips. The EHW chip attained almost twice the compression rates for printer images—this compression mechanism will be used in a commercial EP printer.

	Printer Image	Fax Image
Lempel-Ziv	3.34	8.41
JBIG	3.35	14.67
EHW	6.52	19.82

Table 1. Comparison of data compression rates.

very quickly. However, traditional data compression techniques are insufficient in both the compression rates and decompression speeds.

The EHW data compression chip can solve these two problems by a precise prediction mechanism using reconfigurable hardware [7]. Image data consists of values for many pixels. Because the value of each pixel tends to be closely related to its neighboring pixels, it is possible to predict the value of a given pixel based on the values of its neighboring pixels. If the value can be correctly predicted, it is not necessary to store it separately, which represents a saving in the size of the image data. This means that compression rates greatly depend on the precision of predictions. In order to increase the compression rate, it is necessary to continually reselect the most suitable prediction mechanism for the varying patterns within an image.

The prediction mechanism in the chip is imple-

Neural Network EHW Chip

The Genetic Reconfiguration of DSPs (GRD) chip, manufactured in April 1998, is evolvable hardware designed for neural network applications [5]. Both the topology and the hidden layer node functions of a neural network mapped on GRD chips are dynamically reconfigured using genetic algorithms.

In neural network applications, optimal performance for a given problem is obtained by creating a neural network with the most suitable topology and the most appropriate node functions (for example, sigmoid function or Gaussian). Furthermore, in order to meet the time constraint imposed by real-time applications, neural network hardware systems need to be “tailored” to an ideal network size for a problem. In general, it is very difficult to design an optimal neural network and process it with scalable parallel hardware.

With the GRD chip, the GA software on the RISC processor continues to reconfigure the neural network topology and node functions in order to maintain the optimal performance. The GRD chip consists of a 32-bit RISC processor and a binary-tree network of 15 DSPs. Each DSP can execute one

node function. Using the binary-tree network, multiple GRD chips can be easily connected to configure scalable neural network hardware.

Because a RISC processor is incorporated within the GRD chip, it does not need the host machine control for these tasks. This is desirable for embedded systems in practical industrial applications, together with the fast *online* learning capability.

The results on simulating an adaptive equalizer in digital mobile communication have showed that execution with a single GRD chip took 2.51 seconds, whereas execution on a Sun Ultra2 200MHz chip takes 36.87 seconds. The planned use of the GRD chip includes applications whose environments vary over time and have real-time constraints, such as CATV modems.

Digital EHW Chip for Adaptive Control

This chip was developed in April 1998 to serve as an off-the-shelf device for implementing adaptive con-

trol logic [3]. Currently, this chip has been utilized for two applications: an autonomous mobile robot and a myoelectric artificial hand.

In most research on EHW, GAs are executed in software on personal computers or workstations. This makes it difficult to use EHW in situations that need circuits to be as small and light as possible. For example, a prosthetic hand should be the same size as a human hand and weigh less than 700 grams. Similar restrictions exist for autonomous mobile robots with EHW controllers. One answer to these problems is to integrate both the GA hardware and the reconfigurable logic into a single LSI chip.

This has been done with the digital EHW chip for adaptive control, which consists of three components, (1) the programmable logic array (PLA), (2) the GA hardware, and (3) a 16-bit CPU core (which is a NEC V30). Arbitrary logic circuits can be dynamically reconfigured on the PLA component according to the chromosomes obtained by the GA

Hardware Evolution System AdAM

**Tomofumi Hikage, Hitoshi Hemmi,
and Katsunori Shimohara**

The Adaptive Architecture Methodology (AdAM) [3] is a platform utilized in a hardware evolution experiment. We constructed the hardware evolution system AdAM based on Hardware Description Language (HDL). This system automatically generates HDL and can evolve HDL—we briefly explain the operation of the system here.

The population of chromosomes is generated according to the HDL production rule. Next, the fitness of each individual is evaluated by the operation simulation at the HDL level. The individual that remains in the next generation is selected according to the fitness evaluation. A genetic operation is given to the chromosome of the selected individual, and the next generation's chromosome population is made. Such evolutionary processes eventually result in individuals that have the desired hard-

ware behaviors. Various experiments are done based on this system. We implemented a diploid chromosome and a progressive evolution model based on this system—the outline of the diploid chromosome and the progressive evolution model are explained here.

Diploid chromosome. The chromosome of the basic AdAM system is a haploid chromosome. We extended the chromosome of the basic system to a diploid

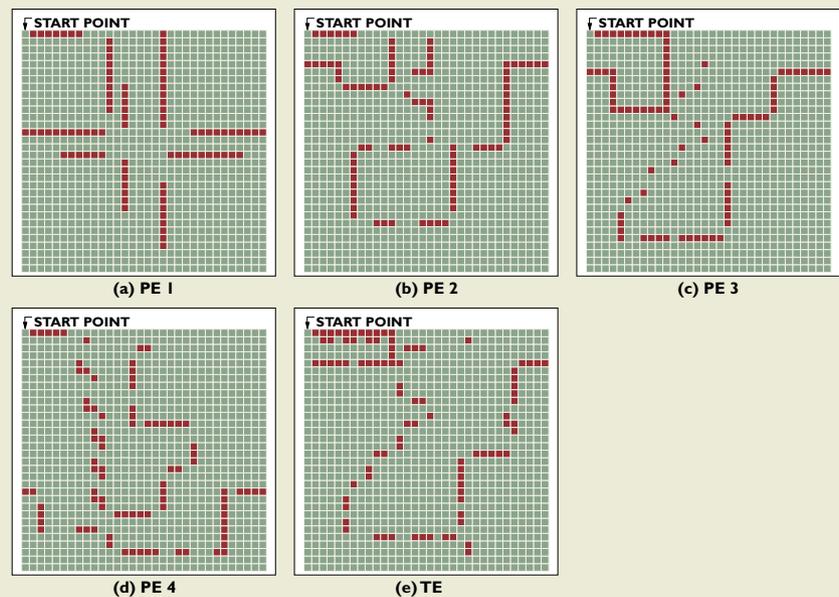


Figure 1. Progressive environments and target environment.

Providing opportunities for the development of new and exciting applications would be impossible without the autonomously adaptive function of evolvable hardware.

hardware. The CPU core interfaces with the chip's environment, as well as supporting fitness calculations when necessary. The size of the GA hardware is almost one-tenth of a 32-bit CPU in gate size. However, genetic operations by this chip are 62 times faster than the Sun Ultra2's operations.

The chip is being used for a control circuit in a myoelectric artificial hand (see Figure 3). The hand can be controlled by myoelectric signals, which are

the muscular control signals. However, myoelectric signals vary from person to person. Accordingly, anyone who has wanted to use a conventional myoelectric hand has to adapt to it through a long period of training (almost one month). To overcome this problem, research is carried out on controllers that can adapt themselves to the characteristics of an individual person's myoelectric signals. Most of this research is conducted using neural networks with

chromosome [1]. The extension was necessary to give the system enough genetic diversity to evolve under environments that change with time.

Most multicellular organisms have diploidy, and it is thought that this maintains population diversity and improves the survival of species in a fluctuating environment. Since the dominant-recessive heredity shown by an organism with diploid chromosomes is important, we extended the chromosome of the basic system to a diploid chromosome for a system mechanism that maintains diversity. Genetic operations were also extended, and meiosis and fertilization were added.

Progressive evolution model. The basic idea of the progressive evolution model [2] is that organisms evolve while acquiring new functions to match environmental changes. We therefore believe that environmental changes drive evolution, so our idea actively uses environmental changes to accelerate evolution. In the progressive evolution model, evolution occurs in environments that change in a stepwise manner toward the final target environment (TE).

The purpose of the model is to divide a large "hurdle" into a series of small steps so that the evolutionary process can easily handle the hurdle. We have extended the operation simulation of the basic system. The environment has been progressively changed from an easy one to a more difficult one. We call the intermediate easy environments Progressive Environments (PE). This model is explained as an example of an artificial ant problem as shown in Figure 1.

The black square in Figure 1 shows the food of an artificial ant. In our problem, artificial ants adapt to a certain environment so that they can quickly and effectively gather food. The total time required for the ant evolution in a progressively changing environment could be shortened more than the ant evolution in the target environment.

We implemented the diploid chromosome and the progressive evolution model based on the AdAM system. In the future, we will consider the construction of a system that replaces the operation simulation of the basic AdAM system with actual hardware. **G**

REFERENCES

1. Hikage, T., Hemmi, H., and Shimohara, K. Hardware evolution system introducing dominant and recessive heredity. In *Proceedings of ICES '96* (Tsukuba, Japan, Oct. 7–8). LNCS 1259, Springer, 1996, 423–436.
2. Hikage, T., Hemmi, H., and Shimohara, K. Progressive evolution model using a hardware evolution system. In *Proceedings of AROB '97* (Oita, Japan, Feb. 18–20), 1997, 18–21.
3. Mizoguchi, J., Hemmi, H., and Shimohara, K. Production genetic algorithms for automated hardware design through an evolutionary process. In *Proceedings of ICEC '94* (Orlando, Fla. June 27–29), 1994, 661–664.

TOMOFUMI HIKAGE (hikage@cslab.kecl.ntt.co.jp) is a research scientist at NTT Communication Science Laboratories in Kyoto, Japan. **HITOSHI HEMMI** (hemmi@cslab.kecl.ntt.co.jp) is a senior research scientist at NTT Communication Science Laboratories in Kyoto, Japan. **KATSUNORI SHIMOHARA** (katsu@cslab.kecl.ntt.co.jp) is an executive manager at NTT Communication Science Laboratories in Kyoto, Japan.

© 1999 ACM 0002-0782/99/0400 \$5.00

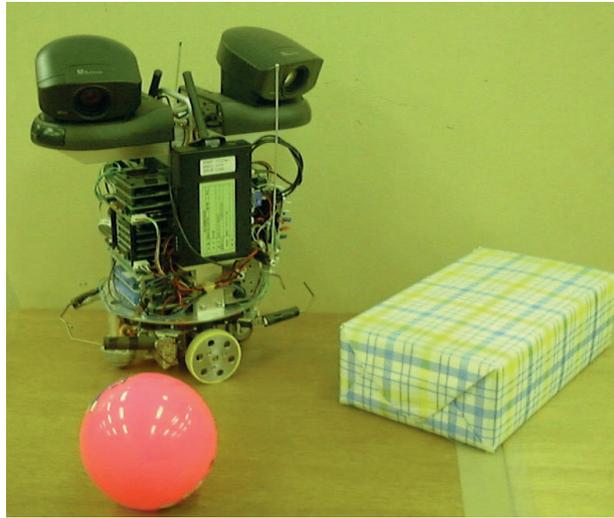


Figure 3.
(left)
Myoelectric
prosthetic hand.
Figure 4.
(right)
Autonomous
mobile robot
Evolver.

back propagation (BP) learning. However, learning with back propagation requires a great deal of time. Because the EHW chip can adapt itself quickly, the learning time can be reduced to a few minutes.

Another application is an adaptive navigation task for a real-world mobile robot that must track a moving colored ball while avoiding obstacles (see Figure 4) [4]. Because the robot moves in an unknown and unpredictable environment, the robot is required to change its behavior adaptively. The robot, called *Evolver*, has two camera eyes and sensors (collision and proximity), but no a priori knowledge of the shapes and positions of obstacles. The control logic implemented on the EHW is continuously reconfigured toward improved behavior. For example, even if one of the sensors becomes broken, the robot autonomously reconfigures its control logic on the EHW momentarily to continue the tracking using other functioning sensors. This adaptation speed is two orders of magnitude faster than that of classical approaches.

Conclusion

As the four applications of EHW presented here illustrate, EHW is a key technology providing opportunities for the development of new and exciting applications, which would be impossible without the autonomously adaptive function of EHW. Although FPGA technology is spreading rapidly and the usefulness of reconfigurable hardware is becoming more widely recognized, this technology is severely limited by the fact that reconfiguration of its hardware is not autonomous. The dynamic and autonomous nature of EHW makes it ideally suited for applications that are time-variant in nature, or for those that suffer from real-time constraints, but as the versatility of the applications described here amply demonstrate, the EHW concept is so generic

that it can be applied to a wide variety of applications. **G**

REFERENCES

1. Goldberg, D.E. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.
2. Higuchi, T., Iwata, M., and Liu, W. Eds. In *Proceedings of First International Conference on Evolvable Systems, Lecture Notes on Computer Science, 1259*, Springer Verlag, 1996.
3. Kajitani, I., Hoshino, T., Nishikawa, D., Yokoi, H., Nakaya, S., Yamauchi, T., Inuo, T., Kajihara, N., Iwata, M., Keymeulen, D., and Higuchi, T. A gate-level EHW chip: Implementing GA operations and reconfigurable hardware on a single LSI. In *Proceedings of the International Conference on Evolvable Systems*, (Sept. 1998), 1–12.
4. Keymeulen, D., Iwata, M., Kuniyoshi, Y. and Higuchi, T. Comparison between an off-line model-free and an on-line model-based evolution applied to a robotics navigation system using evolvable hardware. In *Artificial Life VI: Proceedings of the Sixth International Conference on Artificial Life*, C. Adami, R. K. Belew, H. Kitano and C.E. Taylor, Eds., MIT Press, 1998, 109–209.
5. Murakawa, M., Yoshizawa, S., Kajitani, I., Yao, X., Kajihara, N., Iwata, M., and Higuchi, T. The GRD chip: Genetic reconfiguration of DSPs for neural network processing. *IEEE Transactions on Computers* (June 1999). To appear.
6. Murakawa, M., Yoshizawa, T., Adachi, T., Suzuki, S., Takasuka, K., Keymeulen, D., and Higuchi, T. Analogue EHW chip for Intermediate Frequency Filter. In *Proceedings of the International Conference on Evolvable Systems*, (Sept. 1998), 134–143.
7. Sakanashi, H. Salami, M. Iwata, M. Nakaya, S. Yamauchi, T., Inuo, T., Kajihara, N., and Higuchi, T. Evolvable hardware chip for high precision printer image compression In *Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98)*, (Madison, Wisconsin, July 1998), 486–491.
8. Yao, X. and Higuchi, T. Promises and challenges of evolvable hardware. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, 28, 4 (Nov. 1998).

TETSUYA HIGUCHI (higuchi@etl.go.jp) is the leader of the Evolvable Systems Laboratory at Electrotechnical Laboratory, MITI, in Japan.

NOBUKI KAJIHARA (kajihara@ccm.cl.nec.co.jp) is the leader of RWCP Adaptive Devices NEC Laboratory in Japan.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.