Hardware Evolution at Function Level

Masahiro Murakawa$^1$ Shuji Yoshizawa$^1$ Isamu Kajitani$^2$
Tatsumi Furuya$^3$ Masaya Iwata$^4$ Tetsuya Higuchi$^4$

$^1$ 7-3-1, Hongo, Bunkyo, Tokyo, Japan, University of Tokyo
e-mail: murakawa@bios.t.u-tokyo.ac.jp
$^2$ 1-1-1, Tennoudai, Tsukuba, Ibaraki, Japan, University of Tsukuba
$^3$ 2-2-1, Miyama, Funabashi, Chiba, Japan, Toho University
$^4$ 1-1-4, Umezono, Tsukuba, Ibaraki, Japan, Electrotechnical Laboratory

Abstract. This paper describes a function-level Evolvable Hardware (EHW). EHW is hardware which is built on programmable logic devices (e.g. PLD and FPGA) and whose architecture can be reconfigured by using a genetic learning to adapt to new unknown environments in real time. It is demonstrated that the function-level hardware evolution can attain much higher performances than the gate-level evolution, in neural network applications (e.g. two-spiral). VLSI architecture of the function-based FPGA dedicated to function level evolution is also described.

1 Introduction

Recently, the idea of evolving hardware itself has attracted a rapidly increasing interest of the researchers. Evolvable Hardware (EHW) is a hardware which is built on software-reconfigurable logic devices (e.g. PLD and FPGA) and whose architecture can be reconfigured by using a genetic learning to adapt to new unknown environments. If hardware errors occur, or if new hardware functions are required, EHW can alter its own hardware structure rapidly and accommodates such changes in real-time.

In order to design the conventional hardware, it is necessary to prepare all specifications of hardware functions in advance. In contrast to this, EHW can reconfigure itself without such specifications. In other words, EHW is most suitable for applications where no hardware specification can be given in advance. Applications solved by artificial neural network (ANN) are such examples because pattern classifier functions can not be obtained until learning is complete.

Research on EHW was initiated independently in Japan and in Switzerland around 1992 (for recent overviews, see [Higuchi94] and [Marchal94]). Since then, the interest is growing rapidly (e.g., EVOLVE95, the first international workshop on evolvable hardware was held in Lausanne in 1995).

Most of researches on EHW, however, have a common problem that the evolved circuit size is small. The hardware evolutions are based on primitive gates such as AND-gate and OR-gate; we call the evolution at this level gate-level evolution. The gate-level evolutions are not powerful for the use in industrial applications.
In order to solve this problem, we propose a new type of hardware evolution, function-level evolution, and a new FPGA architecture dedicated to the function-level evolution. We demonstrate that ANN applications such as two-spiral are solved by the function-level EHW. Actually EHW can synthesize a non-linear function genetically. This suggests that EHW may substitute ANN in industrial applications because EHW enables faster and more compact implementation than ANN.

In section 2, the basic idea of EHW is described. Section 3 introduces the functional-level evolution. Its advantage is demonstrated in section 4 by ANN applications. Section 5 proposes the FPGA architecture for the function-level evolution. Section 6 describes future EHW applications and concludes this paper.

2 Evolvable Hardware (EHW)

Function-level evolution, the theme of this paper, is a natural extension of the gate-level evolution. So, here we describe a basic idea of the gate-level EHW.

EHW changes its hardware structure to adapt itself to the environment in which it is embedded. To attain this goal, EHW utilizes PLDs and Genetic Algorithms (GAs) [Goldberg89]. PLDs are hardware devices whose architecture can be determined by downloading binary strings, called architecture bits. Architecture bits are the compilation result of higher level hardware description such as boolean functions and truth tables. On the other hand, GAs are robust search algorithms which use multiple chromosomes (usually represented as binary strings) and apply the natural selection-like operation to them to find out better solutions.

The basic idea of EHW is to regard the architecture bits of PLDs as chromosomes of GAs and to find out better hardware structure by GAs, as shown in Figure 1. Every chromosome is downloaded for evaluation while learning.

![Fig. 1. Evolvable Hardware (EHW)](image1)

![Fig. 2. The EHW-board](image2)