17p-A10-6

Demonstration of Dynamic Reconfiguration in a Crystalline IGZO-based Multi-Context FPGA

°**Nora Björklund**[1,2], **Yuki Okamoto**[1], **Takeshi Aoki**[1], **Munehiro Kozuma**[1], **Yoshiyuki Kurokawa**[1], **Takayuki Ikeda**[1], **Naoto Yamade**[1], **Yutaka Okazaki**[1], **Hidekazu Miyairi**[1], **and Shunpei Yamazaki**[1]
[1]**Semiconductor Energy Laboratory Co., Ltd.,** [2]**Linköping University, Sweden,**
E-mail: nbi014@sel.co.jp

I. Introduction

Field programmable gate arrays (FPGAs) are user programmable hardware mainly consisting of programmable logic elements (LEs) and programmable routing switches (RSs). The LEs, each performing a simpler logic function, are connected to each other by RSs to create complex logic circuits. How the RSs connect the LEs, and also what logic function each LE executes is decided by data written to configuration memories (CMs) in the RSs and the LEs respectively. Since logic operations are executed in the LEs, the conventional way to attain greater circuit design freedom is to increase the number of LEs. However, in most cases all LEs are not used simultaneously, which for FPGAs with a great amount of LEs results in several LEs not executing for periods of time. In 1997 Trimberger et al. described a time-multiplexed FPGA (in this paper called multi-context FPGA (MC-FPGA)) that uses configuration contexts to extend the FPGA's computation capacity temporally and thus avoiding additional LEs [1]. To create an additional context, CMs are added in every RS and LE to create "memory planes" that the FPGA can switch between.

However, implementing contexts can be expensive when considering power consumption and interconnect complexity, and the need for faster, power- and area-efficient MC-FPGAs has given rise to implementations with new and innovative materials. In previous work, we have presented a fine-grained MC-FPGA design [2] using the *c*-axis-aligned-crystalline In-Ga-Zn-O (CAAC-IGZO) field effect transistors (FETs) to decrease both area and power consumption. In this paper we demonstrate that an MC-FPGA using CAAC-IGZO FETs safely can perform dynamic reconfiguration (DR) of an inactive context while executing another context's configuration.

II. DR in an MC-FPGA Using Crystalline IGZO

Fig.1 shows a conceptual design of a conventional FPGA (a), and our proposed MC-FPGA (b). To create temporal function space the MC-FPGA can load configurations to an inactive context from an external source while in use. Our proposed MC-FPGA have been realized with two contexts and is using the LE described in [3] and the RS in [4].

To demonstrate DR in our MC-FPGA, a SPICE simulation was performed where the FPGA could load three different programs:

Task 1: An incremental shifter that shifts a signal from output[0] to output[19].

Task 2: A decremental shifter that shifts a signal from output[19] to output[0].

Task 3: A divider where the frequency of output[*i*] is half the frequency of output[*i*-1].

The results of the simulation can be seen in fig. 2, in which we clearly can see that the contexts are switching, and also how context 1 changes from its initial shifter configuration to a divider while the FPGA executes context 0.

During the coming presentation, we will discuss the details of how our MC-FPGA is constructed, how we enable DR and what benefits we gain from using crystalline IGZO technology in the circuitry.

References

[1] S. Trimberger et al., *Proc. IEEE Symp. FPGA-Based Custom Computing Machines*,1997, p. 22.
[2] M. Kozuma et al., *JJAP*, **53**: 2014. p. 04EE12.
[3] M.Ikeda et al., *JSAP 2013 autumn* 18p-D3-7.
[4] T. Nakagawa et al., *JSAP 2014 spring* 20p-E8-7.

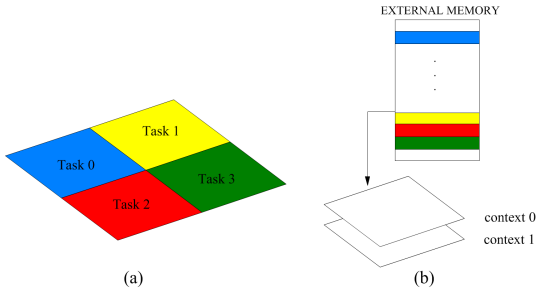

Fig. 1. (a) A conventional FPGA where all tasks need to be configured at once, (b) a dynamically reconfigurable MC-FPGA with two contexts where the context configuration data is fetched from an external memory and can therefore emulate a larger logic design in time.

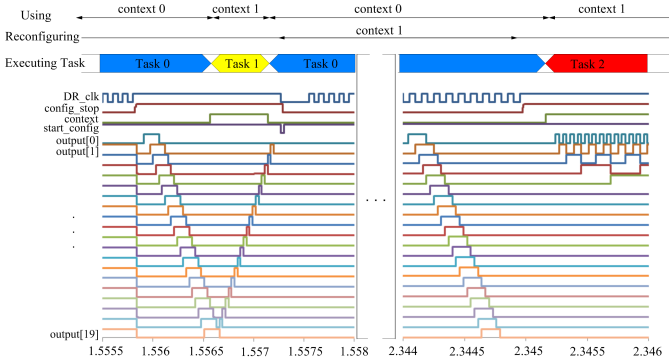

Fig. 2. SPICE simulation of DR in a MC-FPGA using CAAC-IGZO FETs.