JOURNAL OF SEMICONDUCTOR TECHNOLOGY AND SCIENCE, VOL.15, NO.2, APRIL, 2015
http://dx.doi.org/10.5573/JSTS.2015.15.2.307

ISSN(Print) 1598-1657
ISSN(Online) 2233-4866

Efficient Fault-Recovery Technique for CGRA-based Multi-Core Architecture

Yoonjin Kim* and Seungyeon Sohn

***Abstract*—In this paper, we propose an efficient fault-recovery technique for CGRA (Coarse-Grained Reconfigurable Architecture) based multi-core architecture. The proposed technique is intra/inter-CGRA co-reconfiguration technique based on a ring-based sharing fabric (RSF) and it enables exploiting the inherent redundancy and reconfigurability of the multi-CGRA for fault-recovery. Experimental results show that the proposed approaches achieve up to 73% fault recoverability when compared with completely connected fabric (CCF).**

***Index Terms*—Embedded systems, coarse-grained reconfigurable architecture (CGRA), multi-core, fault tolerant computing**

I. INTRODUCTION

To guarantee reliability in the embedded system, the computing engines such as embedded processors or accelerators should be tolerant to prevent decommissioning the entire systems from a few failures. Coarse-Grained Reconfigurable Architecture (CGRA) based multi-core architecture can be considered as a suitable solution for the fault-tolerant computing engine with the dynamic redundancy [1] because of its inherent redundancy and reconfigurability. Fig. 1 shows such an example of the CGRA-based multi-core architecture – it is composed of four CGRAs and on-chip communication architecture like networks-on-chip (NoC) or on-chip bus which couples them. However, until now, there have been a few research projects [2-4] based on fault-tolerant CGRA without exploiting such strengths of CGRA as well as their works are limited to single CGRA. Therefore, in this paper, we propose an efficient fault-recovery technique that enables exploiting the inherent redundancy and reconfigurability of the multi-CGRA.

Manuscript received Aug. 24, 2014; accepted Mar. 30, 2015
Dept. of Computer Science, Sookmyung Women's University, Korea
E-mail : {ykim, kjkj35}@sookmyung.ac.kr
Yoonjin Kim is the corresponding author on this paper.

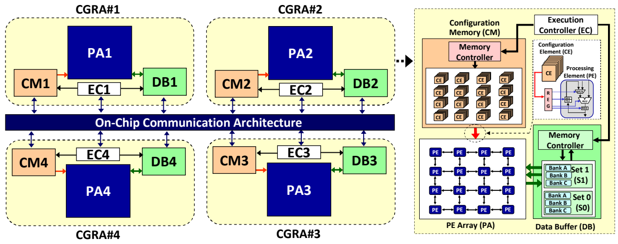


Fig. 1. CGRA-based multi-core architecture.

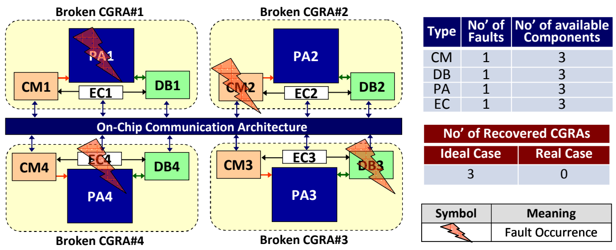


Fig. 2. Example#1: homogeneous 4 CGRAs have 4 faults.

II. MOTIVATION

Fig. 2 shows Example#1 that every CGRA is broken because a dysfunctional component stops each CGRA from working - we assume that the dysfunction components have permanent faults. However, we can consider recovering the broken CGRAs by replacing the dysfunctional components with available ones in other CGRAs because the available components are enough to revive up to 3 broken CGRAs. However, the conventional multi-CGRA organization as Example#1 does not allow such a recovery process because the available components in different CGRAs cannot be recombined to form working CGRAs by only on-chip communication architecture. It means that such a monotonous aggregation of several CGRAs does not enable exploiting the inherent redundancy and reconfigurability of the multi-CGRA when faults occur. Therefore, in the next section, we propose an efficient reconfiguration technique on ring-based sharing fabric (RSF) for supporting efficient recombination for fault-recovery.

III. Fault-Recovery Technique on Ring-Based Sharing Fabric (RSF)

1. Ring-based sharing fabric (RSF)

We hypothesize that a multi-CGRA can support component-level inter-CGRA reconfiguration that means use of each component is not limited to a CGRA – the components means Configuration Memory (CM), Data Buffer (DB), or PE Array (PA). Then, the available components in different CGRAs can be recombined to form working CGRAs for fault-recovery. Therefore, we can easily consider a highly flexible fabric for inter-CGRA reconfiguration as completely connected fabric (CCF) that enable any combination of mapping between all of the components. However, such a full connectivity causes significant area and power overhead with increasing the number of CGRAs. Therefore, we propose ring-based sharing fabric (RSF) as shown in Fig. 3(b). The RSF connects all of the PAs through single-cycle interconnections and a DB (or a CM) is shared by two adjacent PAs on the RSF. Such connectivity shows minimal interconnection overhead even though the number of CGRAs increases. In addition, Fig. 3(b) represents fault-recovery process on RSF for Example#1. It illustrates the RSF that is configured for fault-recovery by inter-CGRA reconfiguration – it replaces the dysfunctional components with available ones in other CGRAs. In this case, the available components are recombined to form 3 recovered CGRAs – it is ideal fault-recovery case for Example#1.

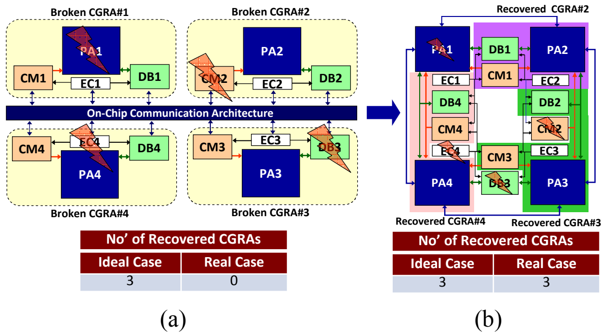


Fig. 3. Fault-recovery on RSF for Example#1 (a) Example#1: homogeneous 4 CGRAs have 4 faults, (b) Inter-CGRA reconfiguration on RSF for recovering the faults.

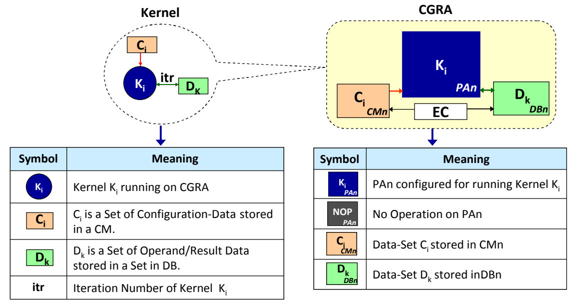


Fig. 4. Symbolic representation of kernel running on multi-CGRA.

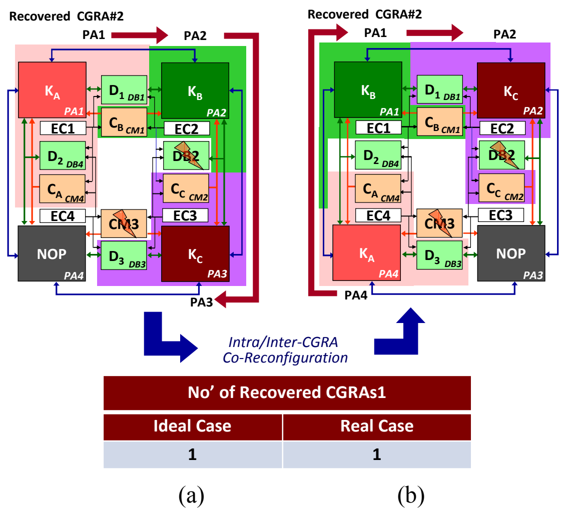
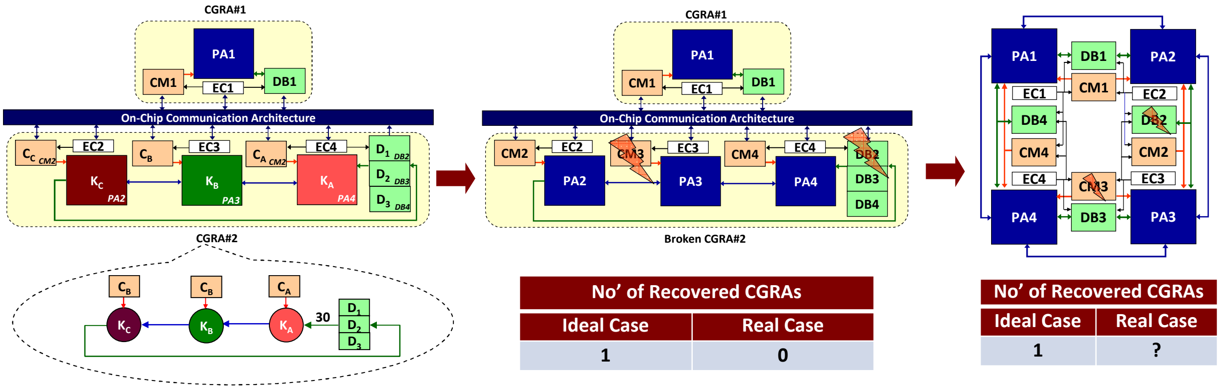


Fig. 5. Intra/inter-CGRA co-reconfiguration on RSF for fault-recovery of Example#2 (a) Configuration#1, (b) Configuration#2.

2. Intra/inter-CGRA co-reconfiguration

The RSF structurally allows that a PA can use up to two CMs or two DBs. However, if a PA requires three



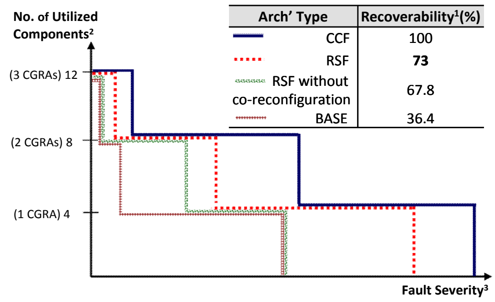
(a) (b) (c)

Fig. 6. Example#2: no interconnection among available DBs and PAs on RSF (a) A kernel-stream runs on CGRA#2, (b) CGRA#2 is broken with 2 faults, (c) The projection of the same fault-occurrence on RSF.

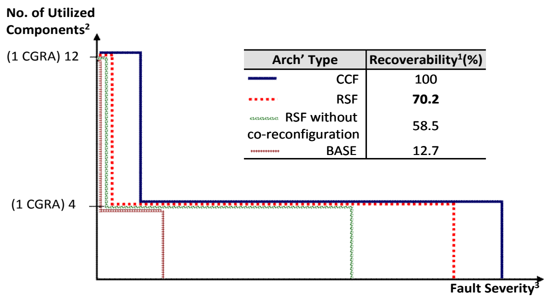
CMs/DBs and over, the RSF cannot utilize these CMs and DBs. Therefore, our proposed intra/inter-CGRA co-reconfiguration technique that alleviates such a structural limitation of the RSF. Fig. 6 shows Example#2 that a working PA is not connected with an available DB on RSF when faults occur. We assume that the pipelining of kernel-stream requires three DBs and it iteratively runs on CGRA#2 at 30 times as Fig. 6(a) - the meanings of the symbols for kernel/CGRA are defined in Fig. 4. In this case, we can consider that CGRA#2 is broken with 2 defective components as Fig. 6(b) and we can project the same fault-occurrence on RSF as Fig. 6(c). However, fault-recovery on the RSF is not structurally possible because of no connectivity between DB3 and PA1. However, we can alleviate such a structural limitation by shifting configuration of the kernel-stream on the RSF. Fig. 5 illustrates how to exploit intra/inter-CGRA co-reconfiguration in order to achieve the shifting configuration. Before all, Fig. 5(a) shows initial configuration of the kernel-stream that PA1 utilizes DB1 (D_1) and DB4 (D_2) for the running of 20 iterations. Then the RSF can be configured as Fig. 5(b) that shows the utilization of one more DB (DB3) for the remaining 10 iterations. The utilization of DB3 (D_3) can be achieved by shifting the configurations of the PAs from to 'PA4 →PA1→PA2'. Therefore, the RSF operates as if a PA is connected with three DBs. In this case, the intra-CGRA reconfiguration means that PA1 and PA2 are reconfigured twice in order to perform K_A/K_B and K_B/K_C. On the other hand, the inter-CGRA reconfiguration enables that three CGRAs are configured with different number of DBs and connected through the direct interconnections. Such a co-reconfiguration can start immediately because each CM is shared by two adjacent PAs that are dynamically reconfigurable. It means that the pipelining of the kernel-stream continually runs on the RSF without stall.

IV. Fault Recoverability Evaluation

We suggest a quantitative evaluation method of fault recoverability of a multi-CGRA based on two criterions. First criterion is fault severity that should reflect the seriousness degree of the fault-occurrence with considering recovery potential. The second criterion is the number of utilized components that are working after recovering faults - it means the recovery degree. It is much more accurate than the number of recovered CGRAs because heterogeneous multi-CGRA can show different number of utilized components despite of the same number of the recovered CGRAs. If we consider two such criterions together, we can see an inverse relationship between them. Therefore, we can plot the relationship on the graph according to 4 types of multi-CGRA as Fig. 7 - CCF, RSF, RSF without co-reconfiguration and BASE. In this figure, 2 graphs show the relationship for the previous examples (Example#1 and Example#2). The fault recoverability of 4 types of multi-CGRA can be quantitatively represented as the rate of the area of the region in the horizontal/vertical-plane bounded by the graph – it is relative rate to CCF (100%) and shown in the right upper table on each graph. We



(a)



(b)

Recoverability¹: 100 X area of graph RSF (or BASE) / area of graph CCF
No. of Utilized Components² : No. of ECs, PAs, DBs and CMs on the recovered CGRAs
Fault Severity³: Increasing severity means decreasing recoverable CGRAs with more faults. 3366 cases of fault-occurrences are listed in the severity order on the horizontal axis.

Fig. 7. Fault recoverability comparison (a) Example#1 as Fig. 3: BASE type – homogeneous 4 CGRAs, (b) Example#2 as Fig. 6: BASE type – heterogeneous 2 CGRAs.

have implemented such a fault recoverability evaluator with C language. The evaluator automatically increases fault severity and estimates corresponding number of utilized components according to 4 types of multi-CGRA – we have run the implemented evaluator on a PC that is composed of an Intel Xeon X5690 operating at 3.46 GHz and 4GB-memory. In the graphs, the RSFs show 73% and 70.2% of the fault recoverability. In addition, we can see the effectiveness of the co-reconfiguration technique – it enhances 5.2% and 11.7% of the recoverability of the RSFs compared with the RSFs without it. Even though the recoverability is not enhanced considerably by the co-reconfiguration technique, it is well worth enough because only the reconfiguration scheme can extend the system lifetime without additional hardware resources.

V. Conclusion

There have been only a few researches based on fault-tolerant CGRA without exploiting the strengths of CGRAs.

Therefore, in this paper, we propose an efficient fault-recovery technique on ring-based sharing fabric (RSF) that enables exploiting the inherent redundancy and reconfigurability of the multi-CGRA. Compared with completely connected fabric (CCF), the proposed approach achieves up to 73% faulty recoverability.

Acknowledgments

This Research was supported by the Sookmyung Women's University Research Grants 1-1203-0070.

References

[1] Daniel P. Siewiorek, "Architecture of fault tolerant computers," *IEEE Computer*, vol. 17, no. 8, pp. 9-18, Aug. 1984.

[2] S. M. A. H. Jafri, S. J. Piestrak, Olivier Sentieys, and Sebastien Pillement, "Error recovery technique for coarse-grained reconfigurable architectures," in *Proc. of IEEE International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS)*, pp. 441-446, Apr. 2011.

[3] D. Alnajiar, Hiroaki Konoura, Younghun Ko, Yukio Mitsuyama, Masanori Hashimoto and Takao Onoye, "Implementing Flexible reliability in a coarse-grained reconfigurable architecture," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 12, pp. 2165-2178, Dec. 2013.

[4] Ganghee Lee and Kiyoung Choi, "Thermal-aware fault-tolerant system design with coarse-grained reconfigurable array architecture," in *Proc. of NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, pp. 272–279, Jun. 2010.

Yoonjin Kim received the B.S. degree in information and communication engineering from Sungkyunkwan University, Seoul, South Korea, in 2003, the M.S. degree in electrical engineering and computer science from Seoul National University, Seoul, South Korea, in 2005, and the Ph.D. degree in computer engineering from Texas A&M University,

College Station, in 2009. From 2009 to 2010, he was a Senior R&D Staff Member with the Samsung Advanced Institute of Technology (SAIT), Gyeonggi, South Korea. In 2010, he joined the faculty of the Department of Computer Science at Sookmyung Women's University in Seoul, South Korea. His research interests include embedded systems, computer architecture, VLSI/system-on-chip design, and hardware/software co-design.

Seungyeon Sohn received the B.S. degree and the M.S. degree in computer science from Sookmyung Women's University, Seoul, South Korea, in 2013 and 2015. Her research interests include embedded systems and reconfigurable computing.