A Hierarchical Context Organization of a Coarse-Grained Reconfigurable Architecture for Block Ciphers

Jinjiang Yang[1, a], Peng Cao[1,b], Jianbing Hu[1,c] and Le Chang[2,d]

[1]National ASIC System Engineering Research Center,Southeast University, Nanjiing 210096, China

[2]Institute of Microelectronics of Tsinghua University, Beijing, 100084, China

[a]yangjinjiang@seu.edu.cn, [b]caopeng@seu.edu.cn, [c]220121230@seu.edu.cn
[d]changle1901@gmail.com

Keywords: CGRA, Block Ciphers, Hierarchical Context

Abstract. A Coarse-Grained Reconfigurable Architecture targeted on block cipher is designed in this paper, which can meet the needs of various cipher supporting in the security applications and enhance the performance of these ciphers. By analyzing the character of the execution and data structure of block ciphers, the hierarchical context organization is adopt. AES and DES are mapped to the architecture and the result shows that the configuration time of them are reduced 42% and 39% respectively. Compared with other similar architectures, the proposed one in this paper has the performance advantage.

Introduction

According to the development of the network communication, the security of data becomes more and more important. Many cryptographic protocols are proposed to ensure the security. As the foundation of these protocols, ciphers can affect the whole system by performance. The cryptographic protocols often have various ciphers to support, while they also have stringent performance requirement because of the increasing network bandwidth. Take IPSec[1] and SSL[2] for example, AES and DES can be chosen for block data encryption by users, and IPSec protocols in network communications needs to support Gbps throughput.

As a general solution, GPP (General Purpose Processor) [3] is widely used for its flexibility rich programmability. However, due to the sequential execution mode, GPP is hardly to meet the increasing requirement on performance, which is driven by the consumer demands. On the other hand, ASIC (Application Specific Intergrade Circuits)[4] can provide the best performance for the specific application, since the data flow and the function unit are optimized specially for the target the application. Meanwhile, these delicate function unit and fixed data paths make it incapable of adapting new requirement of applications or update in cryptographic protocols.

New philosophy on design of the architecture is required and reconfigurable architecture (RA)[5] is the satisfactory tradeoff between ASIC and CPU. Computing task can be directly mapped onto the resources of RAs to avoid the software execution overheads. The RAs even have the post fabric flexibility so that it can reconfigure themselves for new application requirement or protocol update after implemented. FPGA (Field Programmable Gate Array) is known as the traditional fine-grained RAs, which are widely used in cipher implementation [6]. However, CGRA (Coarse-grained Reconfigurable Architecture) represents another class of reconfigurable architecture, which replaces the LUT (Look-up Table) in FPGA with coarser computational blocks and simplify the interconnection pattern of FPGA. Many ciphers have been implemented onto CGRAs, such as RCPA/RCBA [7]. However, large amount of the configuration in CGRA becomes the bottleneck of performance. A Coarse-grained Reconfigurable Architecture is designed for block cipher. By analyzing the character of the execution and data structure of block ciphers, the hierarchical context organization is adopt to reduce the configuration time and enhance the performance of block cipher.

Coarse-Grained Reconfigurable Architecture for Block Cipher

Architecture Overview.

The Coarse-Grained Architecture for block cipher is shown in Fig.1. It consists of three key modules: a Reconfigurable Computing Array (RCA) to speed up computing-intensive tasks, a Context Controller (CC) to configure the RCA, and a data buffer to store the temporary data during the runtime of the RCA. The RCA will process the computations by mapping and setting the relevant configurations (also called the context), which is managed by the Context Controller. There is also two FIFOs provide a high bandwidth data path for plaintext and cipher text.

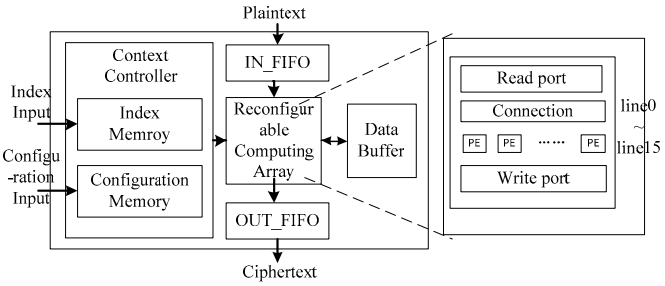

Figure.1:Overview of the coarse-grained architecture for block cipher

Reconfigurable Computing Array(RCA).

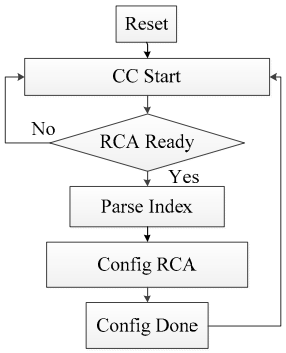

Figure.2: The executing process of CC

Hierarchal Context Organization

As shown in Fig.1, all the modules in the RCA can be configured, such as read port, write port, PEs and connections. This means a lot of configuration is needed for the whole architecture [8], and it will increase the time cost to configure the RCA. As it is depicted in Eq.1, the total time(Ttotla), of the RCA can be divvied into two parts, configuration time(Tconf) and calculation time(Tcalc).

$$Ttotal = Tconf + Tcalc \tag{1}$$

Furthermore, Tconf can be divided into a sequence t_conf_i, which means the confiuration time of the ith DFG of the algorithm. When the whole RCA need to be configured, t_conf_i reaches the biggest value t_max. Fig.3 shows that, when every DFG needs to configure the whole RCA, the total time(Ttotal) will be significant long.

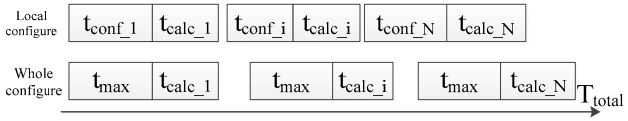

Figure.3: The total time of RCA

According to the analysis of block cipher, the excution and data structure of block ciphers are similar, which leads to that different DFGs share lots of the same operations. Therefore, the

organization of the context can be optimized by separating the configuration in terms of the frequency of ultilization. The detail of the hierarchical context organization is shown in Fig.4. The context can be divided into three kinds, which are context of PE, context of connection, context of read/write. However, the frequency of ultilization with the three kinds of context are not the same. Before optimization, the three kinds of context in each line are updated together, which makes the configuration time of every DFG of the algorithm to be t_max. After optimization, the three kinds of context are separated. The context in every line will be updated partially, when this kind of context in the DFG is changed. Take context of connection for example, for the connection is complex, this kind of context is significant big. However, the frequency of ultization of this kind of context is much lower than other kinds. Thus, after optimization, the configuration time of the block cipher, such as AES and DES, can achieve a considerable reduction.

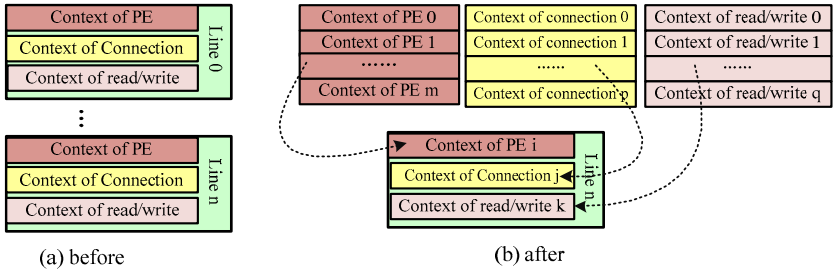

Figure4. The optimization of the organizaiton of contex

Result and Comparison

As the representative block cipher, AES and DES are mapped to the coarse-grained reconfigurable architecture designed in this paper. The result are shown in Fig.5 and Fig.6.

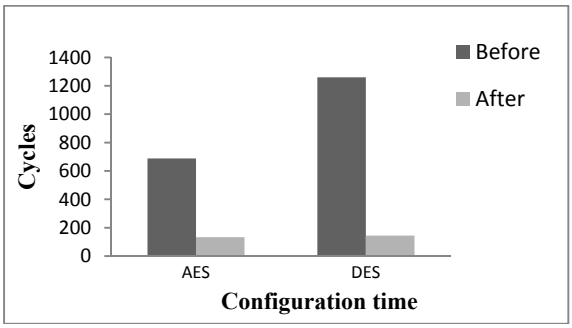

Figure.5.:The configuration time reduction

As it is shown in Fig.5, after the optimization mentioned above, the configuration time reduction is considerable, which means the total time of the algorithm is reduced according to Eq.1. The AES configuration time is reduced from 688 cycles to 133 cycles, while the DES configuration time is reduced from 1260 cycles to 145 cycles. Fig.6 depicts that the percentages of Tconf/Ttotal of AES and DES reduce to 48% and 56%, respectively

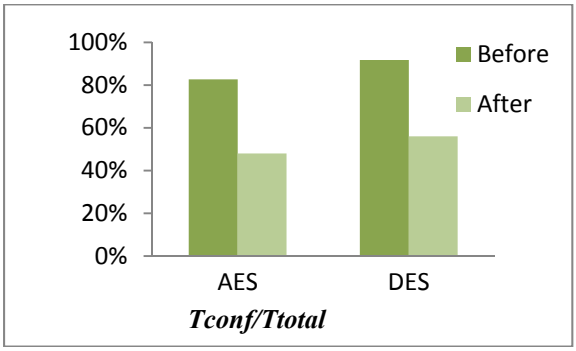

Figure.6: The percentage of Tconf/Ttotal

Table 1 shows the comparasion result with other simillar architectures. The coarse-grained

reconfigurabe architecture desigend in this paper is synthesized under SMIC 55nm CMOS technology. The result showns that our architecture achieves higher throughput and area efficiency than the simillar architectures.

Table 1: Comparasion with other simillar architecures

	Technology (nm)	Frequency (MHz)	Algorithm	Performance (Mbps)	Area (Kgate)	Area Efficiency (Kbps/gate)
RCPA[7]	180	180	AES DES	794 460.8	1490	0.53 0.31
COBRA[8]	350	130	AES	1451	6692	0.22
Tihs paper	55	300	AES DES	4430 3500	5010	0.87 0.69

Conclusion

A Coarse-grained Reconfigurable Architecture is designed for block cipher in this paper. To achieve high performance and area efficiency, the hierarchical contex organization is adopt to reduce the configuration time. The comparison result shows that the block algorithm implementation on the architecture designed in this paper outperforms others implemented on the similar architectures.

References

[1] R. Atkinson., Security architecture for the internet protocol. IETF Draft Architecture ipsec-arch-sec00[R]. 1996.

[2] http://datatracker.ietf.org/wg/tls/documents/.

[3] L. Bin and B. M. Baas. Parallel AES Encryption Engines for Many-Core Processor Arrays[J]. IEEE Transactions on Computers, vol. 2013,62:536-547,.

[4] S. K. Mathew, F. Sheikh, M. Kounavis, S. Gueron, A. Agarwal, S. K. Hsu, et al. 53 Gbps Native GF(2^4)^2 Composite-Field AES-Encrypt/Decrypt Accelerator for Content-Protection in 45 nm High-Performance Microprocessors[J]. IEEE Journal of Solid-State Circuits, 2011, 46:767-776,.

[5] R. Hartenstein. A decade of reconfigurable computing: a visionary retrospective[C]. the Proceedings of the conference on Design, automation and test in Europe, Munich, Germany, 2001.

[6] L. Qiang, X. Zhenyu, and Y. Ye. A 66.1 Gbps single-pipeline AES on FPGA[C]. 2013 International Field-Programmable Technology (FPT), Conference, 2013: 378-381.

[7] D. Zi-bin, Y. Xiao-hui, R. Qiao, and Y. Xue-rong. The research and design of reconfigurable cipher processing architecture targeted at block cipher[C]. ASICON '07. 2007:814-817.

[8] A. J. Elbirt and C. Paar. An instruction-level distributed processor for symmetric-key cryptography[J]. IEEE Transactions on Parallel and Distributed Systems, 2005 16:468-480.