Design for High Throughput SHA-1 Hash Function on FPGA

Jae-woon Kim, Hu-ung Lee, Youjip Won
Dept. of Electronics and Computer Engineering, Hanyang University, Seoul, Republic of Korea
ragingwind@hanyang.ac.kr, oihtoto@hanyang.ac.kr, yjwon@hanyang.ac.kr

***Abstract*—In this paper, we propose SHA-1 architectures to achieve high-throughput hardware implementations. Two techniques such as loop unfolding and pre-processing were used for high-speed SHA-1 core design. The system is made of four sub-modules to increase throughput. Xilinx Virtex-6 FPGA is used for implementation. Implemented SHA-1 module achieves a throughput of 7.35 Gbps, and its behavior has been verified by connecting with Xilinx MicroBlaze soft processor.**

Keywords-SHA-1, hash function, hardware implementation

I. Introduction

Recently, the development of network systems has brought the explosion of digital data along with the emergence of new technologies such as the virtualization technology, cloud computing. The storage system that store and manage data is evolving with its importance on that account. Enlarged data within the storage system will have much the same content. To manage these redundant data, the data is divided into pieces of units called chunks or blocks. Each unit is using a hash algorithm to generate the fingerprint of a piece, in which proves the redundancy of data. Because, the values returned by a hash algorithm have unique values. As the system is larger and accelerated, the use of algorithm that requires large computational loads may be the bottleneck of the entire system. Hardware implementations present higher throughput than software and decrease CPU utilization, hence, it should contribute to improving overall system performance.

In this paper, SHA-1 algorithm, which is one of the most popular cryptographic hash functions, has implemented on the FPGA to apply to storage system being developed. Loop unfolding and pre-processing were used to perform two hash operations in a cycle and to reduce the critical path of hash operation, respectively. The system was configured as pipeline architectures based on the designed SHA-1 core to achieve high throughput and low latency of input data. The proposed SHA-1 architecture was implemented on Xilinx ML605 evaluation board that is based on the Virtex-6 FPGA.

II. SHA-1 Algorithm and related works

SHA-1[1] is a cryptographic hash function designed by the National Security Agency of the U.S.A. and published by NIST as a improved algorithm of SHA-0 in 1995. And it is used in many security protocols and applications, including TSL, SSL, SSH and IPsec. When a message smaller than 2^{64}-bit is input, the SHA-1 algorithm performs message padding dividing the message into 512-bit sub-messages and then, 80 SHA-1 operations are applied to each sub-message. The output value has a length of 160-bit from performing operations and is added to the previous hash code is the final hash code.

The basic SHA-1 algorithm is illustrated in Fig. 1.

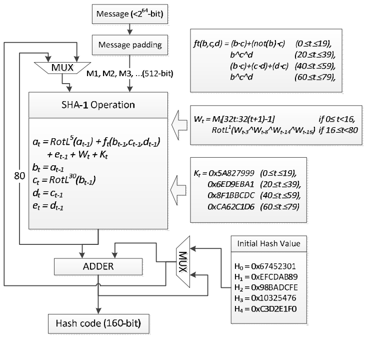

Figure 1. The basic SHA-1 algorithm.

For high-throughput SHA-1 design, the techniques such as loop unfolding, pre-processing[2, 3], multi-input adding based on a carry-save adder[4] and pipelining[5] have been proposed. In this paper, design methods that loop unfolding, pre-processing and pipelining were used.

III. Design

A. Base SHA-1 core

The proposed SHA-1 core design employs two-unfolding and pre-processing techniques as shown in Fig. 2., and it is designed on the basis of the Equation (1) The critical path in the base SHA-1 core has delay of 2 additions and function f_t as shown in the dotted line in Fig. 2. Six 32-bit registers were used to store temporary variables. The SHA-1 architecture requires 41 cycles to generate message digest.

$$\begin{aligned}
a_t &= RotL^5\{RotL^5(a_{t-2}) + l_{t-2}\} + f_t(a_{t-2}, RotL^{30}(b_{t-2}), c_{t-2}) + m_{t-2} \\
b_t &= RotL^5(a_{t-2}) + l_{t-2} \\
c_t &= RotL^{30}(a_{t-2}) \\
d_t &= RotL^{30}(b_{t-2}) \\
e_t &= c_{t-2} \\
l_t &= f_t(b_t, c_t, d_t) + e_t + W_t + K_t \\
m_t &= d_t + W_{t+1} + K_{t+1} \\
n_t &= W_{t+2} + K_{t+2}
\end{aligned} \tag{1}$$

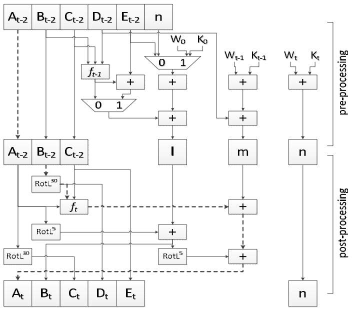

Figure 2. Base SHA-1 core architecture

B. Pipeline Architecture

It is shown in Fig. 3. that the pipeline architecture based on the SHA-1 core of the previous section. The SHA-1 core can run 80 operations in 41 cycles because it is applied loop unfolding and pre-processing, so the pipeline architecture can have up to 40 stages. In this paper, there are 4 stages and each stage operates during 10 cycles, but the first stage handles the additional operation of 1 cycle due to pre-processing.

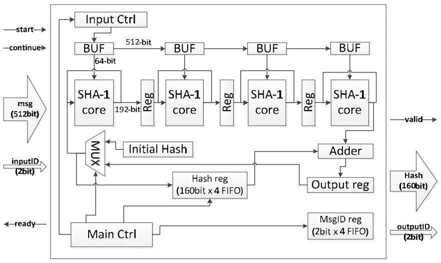

Figure 3. 4-stage pipeline SHA-1 architecture

IV. Implementation

Our designs implemented using a Xilinx ML605 evaluation board[6]. Based on the Virtex-6 LX240T FPGA[7], this evaluation board holds 512MB of DDR3 SDRAM. All the implementations were designed in Verilog HDL. Xilinx ISE 13.2 was used in design module, synthesis, placement & routing, programming and configuration. Simulation and verification was performed with ModelSim SE 6.6b and Xilinx Chipscope logic analyzer respectively.

The proposed SHA-1 design occupies 5,652 Slice LUTs, 3,818 Slice Registers and 1,649 Slices (Virtex-6 FPGA Slice comprises four LUTs and eight Registers[7]). To exhibit the benefits of applying the proposed design methodology, SHA-1 hash function was implemented following the steps of the proposed methodology and the results are shown in TABLE I.

TABLE I. RESULT OF HARDWARE IMPLEMENTATIONS

Design	Clock (Mhz)	Latency	Slice register	Slice LUT	Throughput (Mbps)
original	165.2	80	1,242	1,417	1,057
unfolding	142.6	40	1,270	1,647	1,825
Pre-processing	161.2	41	1,250	1,934	2,013
Pipeline(4p)	150.7	42	3,818	5,652	7,351

The throughput is defined with the following formula:

$$\text{Throughput} = (\text{block size} \cdot \text{frequency} \cdot \text{p}) \,/\, \text{latency} \quad (2)$$

Where block size is 512 bits, the p is the stages of pipeline. Calculated throughput of our design is 7.35 Gbps. The comparison with other publications is described in TABLE II. .

TABLE II. COMPARISON OF OTHER PUBLISHED IMPLEMENTATIONS

Design	Device	Clock (Mhz)	Latency	Slice	Throughput (Gbps)
[3]	Virtex-2	91.0	40	4,848	4.7
[2]	Virtex-2	118.0	80	2,894	5.9
proposed	Virtex-6	150.7	42	1,649	7.35

For verification of the proposed SHA-1 architecture, we have implemented Microblaze soft processor provided by Xilinx IP library on the FPGA and added it to our design. The interface of the SHA-1 core with Microblaze is based on the Xilinx Fast Simplex Link(FSL) stream bus. The synthesized soft core reaches up to 150 MHz on FPGA[6]. Xilinx Software Development Kit(SDK) tool was used as software testing environment. We used four 512byte message groups as the test set and verified the results running on the board by Xilinx Chipscope Logic Analyzer. According to the measurement results, computations of whole test set require 414 cycles, and padding operations require 48 cycles additionally.

V. Conclusion

In this paper, we proposed high-throughput SHA-1, which is a popular cryptographic hash function, hardware architecture and implemented it on Xilinx Virtex-6 FPGA. Several techniques, such as loop unfolding, pre-processing, pipelining, were used to achieve high throughputs for SHA-1. Proposed SHA-1 architecture requires 1,649 slices and achieves a throughput of 7.35 Gbps at 150.45 MHz. Finally, we added our design to Microblaze soft processor and interfaced it with FSL stream link for verification.

Acknowledgment

This work is sponsored by IT R&D program MKE/KEIT. [No.10035202, Large Scale hyper-MLC SSD Technology Development].

References

[1] F. PUB, "Secure hash standard," *Public Law,* vol. 100, p. 235, 1995.

[2] E. H. Lee, J. H. Lee, I. H. Park, and K. R. Cho, "Implementation of high-speed SHA-1 architecture," *IEICE Electronics Express,* vol. 6, pp. 1174-1179, 2009.

[3] H. E. Michail, A. P. Kakarountas, A. S. Milidonis, and C. E. Goutis, "A Top-Down Design Methodology for Ultrahigh-Performance Hashing Cores," *Dependable and Secure Computing, IEEE Transactions on,* vol. 6, pp. 255-268, 2009.

[4] E. G. Jung, D. Han, and J. G. Lee, "Low area and high speed SHA-1 implementation," in *SoC Design Conference (ISOCC)*, 2011, pp. 365-367.

[5] L. Jiang, Y. Wang, Q. Zhao, Y. Shao, and X. Zhao, "Ultra high throughput architectures for SHA-1 hash algorithm on FPGA," in *Computational Intelligence and Software Engineering(CiSE)*, 2009, pp. 1-4.

[6] D. CODEC, "ML605 Evaluation Board," *ML605 Hardware User Guide,* 2009.

[7] I. Xilinx. (2010, Virtex-6 Family overview. Available: http://www.xilinx.com/support/documentation/data_sheets/ds150.pdf