IEICE Electronics Express, Vol.6, No.4, 180–186

Implementation of HIGHT cryptic circuit for RFID tag

Young-Il Lim[1], **Je-Hoon Lee**[2a)], **Younggap You**[2], **and Kyoung-Rok Cho**[2]

[1] *DRAM Design Team, Samsung Electronics Co.*
San 16, Banweol-dong, Hwasung, Gyeonggi 445–701, Rep. of Korea
[2] *BK21 Chungbuk Information Tech. Center, Chungbuk Nat'l University,*
San 12, Gaeshin-dong, Heungduk-ku, Cheongju 361–763, Rep. of Korea
a) *leejh@hbt.cbnu.ac.kr, krcho@cbu.ac.kr*

Abstract: This paper presented a simplified hardware architecture of the block cryptographic algorithm, HIGHT, for wireless applications like a RFID system. We have modified the original HIGHT algorithm that reduced the critical path in the key scheduler and dismissed redundant logics sharing encryption and decryption datapathes, and thereby yield a smaller silicon area. The proposed HIGHT supporting both encryption and decryption had 2,608 gates, 13% smaller than the original HIGHT design excluding decryption block. It consumes the average power 10.8 μW at 2.5 V for 100 kHz. It can be applicable to passive RFID tag without serious difficulty in size and power. Also, the maximum clock frequency of 125 MHz allows a data throughput rate of 235 Mbps that can support cryptography of high-speed multimedia data.

Keywords: block cipher, RFID, security, cryptic circuit, ubiquitous
Classification: Integrated circuits

References

[1] K. Finkenzeller, *RFID-Handbook 2nd Edition*, 2003.
[2] National Inst. Of Standard and Technology (NIST), *FIPS-197:Advanced Encryption Standard*, 2001.
[3] M. Feldhofer, et al., "AES implementation on a grain of sand," *IEE Proc. Information Security*, vol. 152, no. 1, pp. 13–20, Oct. 2005.
[4] D. Hong, et al., "HIGHT: a new block cipher suitable for low-resource device," *Proc. CHES2006 LNCS 4249*, pp. 46–59, Oct. 2006.
[5] R. Dennard, et al., "Design of Ion-implanted MOSFETs with very small physical dimensions," *J. of SSCC*, vol. 9, no. 5, pp. 256–268, Oct. 1974.

1 Introduction

The battery life and the privacy between users become critical issues as wireless devices become more popular. Recent information security addresses ubiquitous sensor network (USN) environment, which emphasizes communi-

DOI: 10.1587/elex.6.180
Received January 05, 2009
Accepted January 19, 2009
Published February 25, 2009

cation availability without limitation of time and location. The wireless system employing radio frequency identification (RFID) system is representative technique in USN [1]. There is a security risk in the data communication between a RFID tag and a reader. The block cipher, AES (advanced encryption standard) and HIGHT are employed to make a strong security [2, 3, 4]. Power consumption and die-size are major design concerns for cryptographic component integrated in the mobile system. M. Feldhofer et al. presented AES that has 3,400 gates and average power consumption is 4.5 μW with 100 kHz clock and 1.5 V [3]. There is another block cipher, HIGHT, which does not only consist of simple operations to be ultra-light but also has enough security as a good encryption algorithm [4]. It requires only 3,048 gates.

This paper proposed a new hardware architecture of HIGHT, aiming at small size, high speed, and low power that will be useful for RFID systems. The proposed architecture shorten the length of the critical path and shares the common datapath for encryption and decryption processes. The implementation results of the proposed HIGHT show the fulfillment of mandatory requirements of a passive RFID tag application.

2 HIGHT algorithm and hardware architecture

A HIGHT has four major functional blocks of key schedule, initial transform, 32 iterative rounds, and the final transform as shown in Fig. 1a. The Key schedule is generating 128 sub-keys from SK_0 to SK_{127} for one encryption process based on a master key as shown in Fig. 1b. The initial transform converts a plaintext to the input of the first round operation using four whitening-key bytes from WK_0 to WK_3, The round transform performs 32 iterative round functions and then, the final transform is responsible for changing the 64-bit output of the last round function to the 64-bit ciphertext using the four whitening-key bytes from WK_4 to WK_7. A whitening key is used to hide the input data of a round function. HIGHT has a simple structure to perform encryption and decryption. It uses simple word operations such as addition mod 2^8, subtraction mod 2^8, XOR, and left bitwise rotation.

There are some differences between an encryption and a decryption of the HIGHT although they have the same transformation processes as shown in Fig. 1c. The decryption process is an inverse operation of encryption. The final transform in encryption eliminates swaps among bytes in the 32nd round function. It transforms the output of the 32nd round into the ciphertext. The byte-swap for decryption is performed in the opposite direction to that in the encryption. The sequence of sub-keys for transform operations reverses to get a decrypted plaintext. Thus, the initial and final transform of the decryption replaces the final and initial transform of the encryption, respectively. The addition operation in the initial and final transforms of encryption replaces the subtraction in decryption. Also, the round function in decryption uses the subtraction instead of addition. These differences can be alleviated by unifying the same datapaths for encryption and decryption and controlling them alternately. It is possible to design more small HIGHT by sharing the

DOI: 10.1587/elex.6.180
Received January 05, 2009
Accepted January 19, 2009
Published February 25, 2009

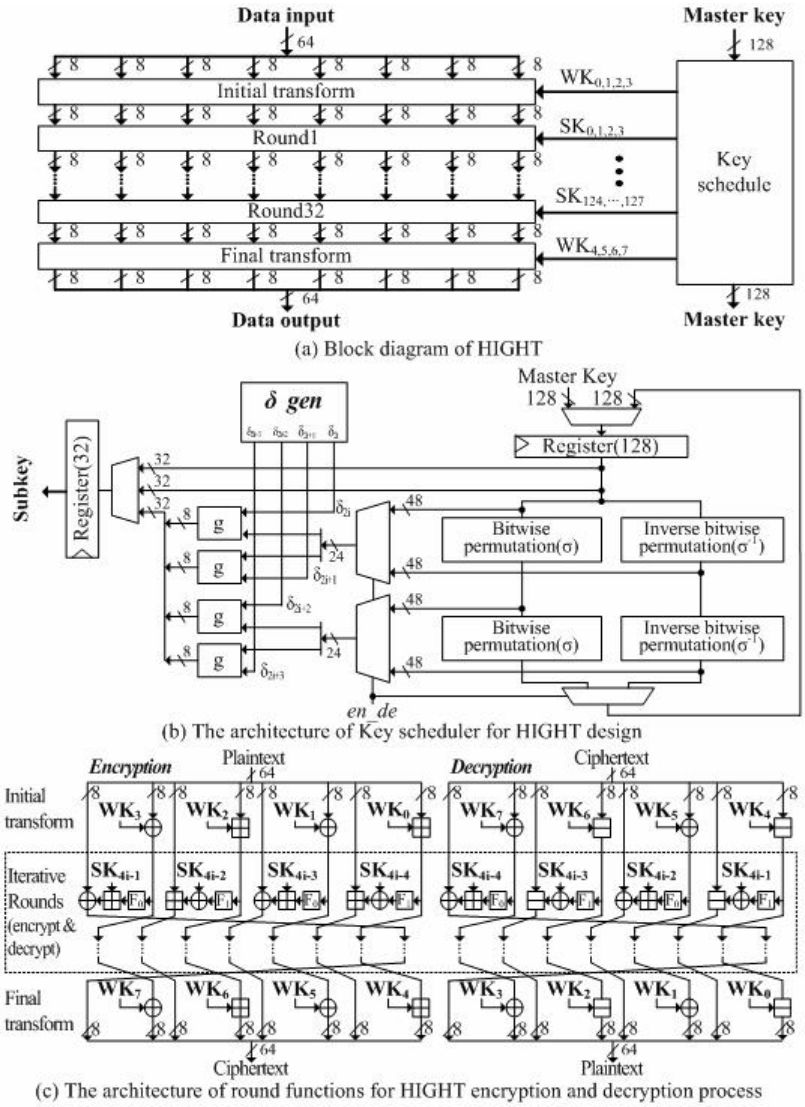


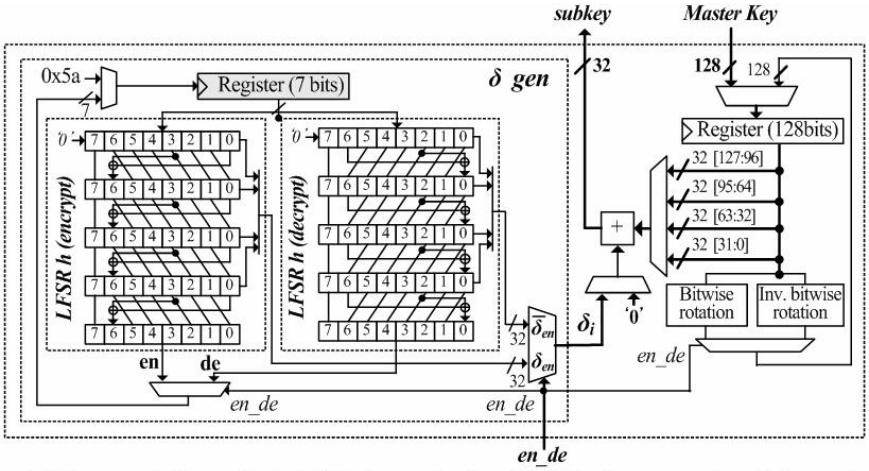
Fig. 1. Block diagram and cryptic algorithm of HIGHT.

common logics between encryption and decryption.

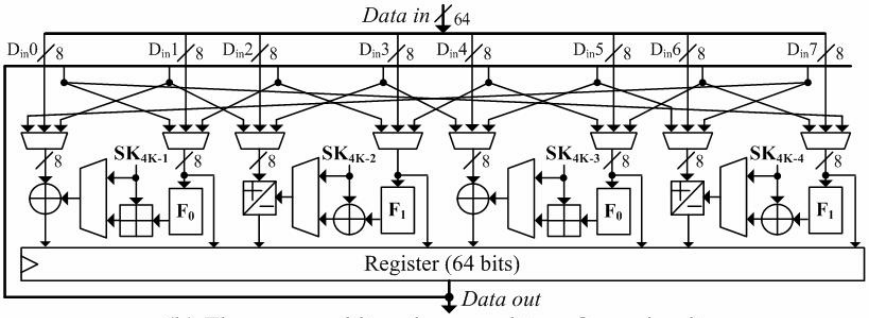
3 The proposed hardware architecture of HIGHT

There are two key design issues in our implementation. We shorten the critical path of the key scheduler. The output of the proposed key scheduler goes through 2-stages, *MUX→adder*, while the output of the key scheduler in the conventional HIGHT goes through 4-stages, *MUX→g function→MUX→ register*. Also, we achieve the area saving by replacing the bitwise permutation and g-function in a conventional HIGHT [4] as shown in Fig. 1b to bitwise rotation and addition operation as shown in Fig. 2a, respectively. In addition, we unify the round function both an encryption and a decryption by sharing the common datapahts in both of them. The proposed design can perform the encryption and decryption alternatively. The select signal, *en_de* chooses the operation of encryption or decryption. Thus, these key design issues bring us gate size reducing and higher operation frequency comparing to the conventional HIGHT design [4].

DOI: 10.1587/elex.6.180
Received January 05, 2009
Accepted January 19, 2009
Published February 25, 2009



(a) Proposed Key schedule block employing *LFSR h for encrypt and decrypt*



(b) The proposed iterative round transform circuit

Fig. 2. The proposed HIGHT architecture.

The proposed HIGHT keeps operation the same with the conventional HIGHT. The initial transform uses four words of the master key. A 32-bit keys passes through the 4-to-1 multiplexer at a time. The proposed HIGHT performs the bitwise rotation for encryption and the reverse bitwise rotation for decryption alternately because it employs both rotation blocks. The key generator produces sub-keys as a master key without any change during the initial and the final transform. However, the sub-keys generated in the round transforms are the results of add operations with 32-bits constant, δ_i obtained from the δ generator. The sub-key used in the initial and the final transform is generated by performing addition operations with '0', not 32-bits constant, δ_i. It can save the addition of MUX logic by using the part of a master key as a sub-key without any changes. During the iterative round transforms, the sub-key can be generated by δ_i obtained from *LFSR h* and 32-bit data among the 128 bits caused by the bitwise rotation of a master key. Each of them performs the addition in 8-bit at a time and it generates a 32-bit sub key by combining four 8 bits. Thus, the sub key used in the 32 round transform can be obtained from the 4 bitwise rotations.

We construct the δ generation circuit having two *LFSR h* circuits for both encryption and decryption separately as shown in Fig. 2a. δ_0 is fixed as 0x5a16, which becomes the initial state of 7-bit *LFSR h*. The output MUX can choose the output δ_i among the output of encrypt *LFSR h*, δ_{en} and the output of decrypt *LFSR h*, δ_{de} according to the selection signal, *en_de*. Each

DOI: 10.1587/elex.6.180
Received January 05, 2009
Accepted January 19, 2009
Published February 25, 2009

number in *LFSR h* represents the bit position of δ_i. It is constructed with a simple wiring with four XOR operations. For performing addition operations within the key scheduler, we generate 32-bit constants by combining four 8-bit outputs. The *LFSR h* circuit for the decryption should operate reversely with that for the encryption.

The proposed architecture of the round transform is depicted in Fig. 2b. It reduces the circuit size of the frequently used round transform among the encryption and decryption algorithms. The round circuit can be applied to the initial and the final transform, too. The 64-bit input data can be divided into 8-bit blocks and represented as follows.

$$Data_{in} = D_{in}7\|D_{in}6\|D_{in}5\|D_{in}4\|D_{in}3\|D_{in}2\|D_{in}1\|D_{in}0 \quad (1)$$

Here, let define $D_{in}7$ and $D_{in}0$ as MSB and LSB, respectively. After input data from $D_{in}7$ to $D_{in}0$ are shuffled alternately as shown in Fig. 2b, the initial transform will be started. And then, the output is transferred to the iterative round transform. The 3-to-1 MUXs in front of the round transform are responsible for untwisting the swap of the round function by selecting the input data to the round transform. There are two differences between encrypt and decrypt. The one is that the addition in encryption replaces subtraction in decryption. The other is that the byte-swap for decryption is performed the opposite direction to that in the encryption. The original master key bit pattern is recovered after the 32 rounds master key scheduling. The four words of the master key are sent to the output for the final transform. The total number of clock cycles for encryption and decryption is 34 clocks.

4 Performance analysis

The silicon implementation of the proposed HIGHT uses MagnaChips 0.35-μm CMOS ASIC process. We choose area optimization option at logic synthesis. We have used NanoSim tool from Synopsys and Spice parameters to obtain the power consumption. Note that the proposed HIGHT meets every mandatory requirement for a RFID tags. Due to the low-power restrictions, the clock frequency of the RFID tag must be divided from 13.56 MHz to 100 kHz. In addition, the circuit size should be under 5,000 gates and the current consumption is under 15 μA, and the encryption and decryption time is under 15 ms. Our implementation supporting both encryption and decryption process comprises 2,608 equivalent gates. It consumes only an average current of 4.3 μA when operated at 100 kHz and 2.5 V supply. It spends 680 μs for encrypting and decrypting 64-bit data. Thus, these results mean that our design is applicable to a passive RFID tag.

Table I shows the comparison results with the other counterpart cipher chips. The AES [3] and the proposed HIGHT were fabricated using 0.35-μm technology. However, the conventional HIGHT [4] was fabricated using 0.25-μm technology. For fair comparison, we evaluate three types of block ciphers on 0.35-μm technology. Thus, we reestimate the performance of the conventional HIGHT on 0.35-μm technology using linear scaling method [5].

DOI: 10.1587/elex.6.180
Received January 05, 2009
Accepted January 19, 2009
Published February 25, 2009

In terms of circuit area, the AES [3], the conventional HIGHT [4], and the proposed HIGHT have 3,400, 3,048, and 2,608 gates, respectively. Our design achieves 13% area reduction comparing to the conventional HIGHT design even though it supports both encryption and decryption processes. It is obtained from the key scheduler optimization and unifying the round logics. In the conventional HIGHT, the key scheduler requires 1,648 gates and the round and others require 1,400 gates. However, the key scheduler in our implementation has 1,591 gates. In the key scheduler design, we added only one $LFSR\ h$ for decryption and control logics, while we reduce the logic area by replacing four bitwise permutations and g-functions to two bitwise rotations and an addition operation. In addition, we prevent the area increase in the round logics by sharing the common datapath for encryption and decryption and by simplifying control logic.

The proposed HIGHT shows outstanding results in power and performance. The AES and the proposed HIGHT implementation have an average current consumption of 8.2 μA and 4.3 μA at the 100 kHz clock frequency, respectively. It notes that the current consumption of the proposed HIGHT design also outperforms twice than that of recently proposed low-resource hardware implementation of AES. In addition, the maximum throughput of the conventional HIGHT is 108 Mbps at the 57 MHz clock frequency. On the other hand, our implementation works flawlessly at 125 MHz clock, thereby the maximum performance of 235 Mbps is achieved. It is due to the shorten the critical path of the key scheduler.

Table I. Comparison results with the other block chipers.

Algorithm	*Function*	*Area (EG)*	*Max. throughput*	*Max. frequency*	*Current (μA@100 kHz)*
AES [3] (0.35 μm)	encrypt	3,400	9.9 Mbps	80 MHz	8.2-μA
HIGHT [4] (0.25 μm) HIGHT [4] (0.35 μm)	encrypt	3,048	151 Mbps 108 Mbps	80 MHz 57 MHz	N/A
Proposed HIGHT (0.35 μm)	encrypt decrypt	2,608	235 Mbps	125 MHz	4.3-μA

5 Conclusions

Portable RFID devices run under severe power limitation. This paper proposes an implementation of the ultra-light block cipher algorithm, HIGHT for a passive RFID tag. The proposed HIGHT design eliminates the redundant logics by sharing the datapath for encryption and decryption. Also, it gives more efficient key schedule design. The proposed design meets the mandatory requirements for RFID device such as power consumption, gate count, and throughput. Our design consumes the average power under 10.8 μW at 100 kHz clock frequency and 2.5 V supply. It has only 2,608 equivalent gates. In addition, it shows the maximum throughput of 235 Mbps. It can be applicable to other application such as multimedia data on a high speed network.

DOI: 10.1587/elex.6.180
Received January 05, 2009
Accepted January 19, 2009
Published February 25, 2009

Acknowledgments

This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD) (The Regional Research University Program/Chungbuk BIT Research-Oriented University Consortium)

DOI: 10.1587/elex.6.180
Received January 05, 2009
Accepted January 19, 2009
Published February 25, 2009