AES-CCM Hardware Architecture using a shared SBox for home security

Selenge Tumurbaatar
Professor of Electronic Engineering Department, Huree University, Mongolia

Abstract

This work was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Assessment) and Yonsei University Institute of TMS Information Technology, a Brain Korea 21 program, Korea. CAD Tools were supported by IDEC.

1. Introduction

Today automation, safety, healthcare and entertainment are integrated into one home area network which is based on a low rate, low power and low cost wireless network based technology - ZigBee. IEEE 802.15.4 is a standard, which supports the protocol in PHY and MAC layer implementation of ZigBee system including its security. Because of its nature of the wireless network and high processing capacity, the efficient security processing is essential. A number of alternatives for the hardware implementation have been undertaken for the home security process depending on the required latency, memory limitation [1]. Since AES (advanced encryption standard)[2] is the computational core of the home security system, its design has a significant effect on the energy consumption and performance of the whole design. Current approaches for the AES implementation are the iterated, pipelined and loop-unrolled architectures. Some architectures suggest higher speed, shorter latency, but take more power and high cost. So we propose 8-bit data-width parallel AES architecture in order to achieve low power and reasonable area. The IEEE802.15.4 standard defines cryptographic security suites providing confidentiality and/or integrity for MAC packets; apart from a keyexchange protocol. The standard specification stipulates that the security implementation must support the AES-CCM-64 suit, which provides four security services: access control, message integrity, message confidentiality and replay protection. With the exception of acknowledge packets, all packets in the IEEE 802.15.4 standard support integrity and confidentiality protection in their data field optionally.

2. AES-CCM Security Suit Design and Analysis

Home security implementation in MAC level, the AES-CCM security suit implementation is mandatory. Since core function of this suit is AES, we focused on AES hardware architecture for efficient implementation.

2.1 AES design analysis

A design of the AES core is based on parallel architectures employing 8-bit data paths. AES functionality is presented in the AES Specification by NIST'99. Since in AES-CCM mode, AES cipher (AES encryption) can only be used for encryption in a transmitter and decryption in a receiver (no need of an inverse cipher), we implemented only AES cipher functionality.

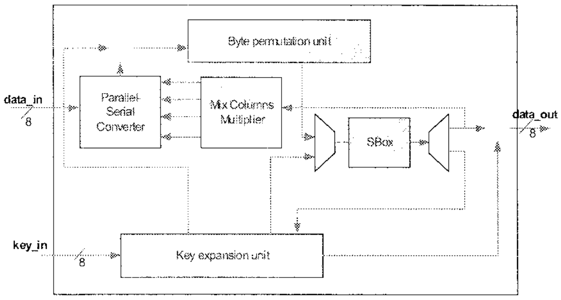

Figure 1. AES core top

Figure 1 shows a top module of AES core architecture. The core supports 128-bit keys and computes one round at a time in 16 clock cycles. It is composed of five components (Parallel- Serial Converter, Byte Permutation Unit, SBox /SubBytes/, MixColumns Multiplier and Key Expansion) in order to complete the AES functionality on 8-bit computation. A block plaintext and encryption key loaded to the core byte by byte at a time parallelly via the input ports data_in and key_in. The initial step, AddRoundKey function of the AES is performed during load state. After ten rounds of operation the ciphertext block can be unloaded byte by byte to the output port data_out. The final round operation (SubBytes, ShiftRows and AddRoundKey) is performed during the data unloading. Except SubBytes transformation unit, all units' operations are the same as [4].

2.2 SubBytes Transformation

Typical Subbytes implementation has three approaches: Look-Up Table (usually for FPGA implementation), GF inverter + affine transformation and direct mapping. Whilst the low power is a key for home security implementation, less power consumed architecture should be chosen. In the work of A.Satoh, IBM Tokyo Research Lab, the SBox implementation consumes the highest power consumption (75% of total power) among AES subfunctions with respect to 128-bit bus 11 round Loop Architecture. This paper proposes a shared SBox architecture for data ciphering and key expansion, while dividing one 16x16 LUT into subdivided LUTs (16x 4x 4) as shown. In [4] architecture, SBox initialized twice for data ciphering and key expansion while utilizing one LUT. Hence logic implementation area as well as power consumption can be reduced.

If the intermediate block data and key input to SBox are located in different sub-LUTs, it can access simultaneously and performs SubBytes operation in a cycle without any clock penalty. In the contrary, they are located the same LUTs, it should perform within two cycles.

This proposed architecture shall be more suitable for 128-bit computation per cycle, because it must initialize SBox 16 times at a time. Consequently, we can achieve more benefits with respect to power saving.

Two 2-bit multiplexers coordinate an access that input data accesses which a sub-LUT(SLUT) is invoked by input data.

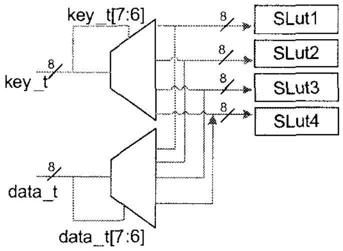

Figure 2. Data path of SBox unit

Generally a SLUT can be reorganized like 2, 4, or 16 subdivided LUTs; however in order to avoid collusion with SBox inputs from data and key, more numbers of LUTs are preferable. Probability of collision at an input port in the SBox module is estimated in Section C. Data path of SBox is shown in Figure 2

2.3 Collision Probability Analysis

In order to select the shared SBox architecture, the collision probability has been calculated. Let assume data to the SBox module is discrete random variables (1)

$$0 \le p \le 1$$

$$P_X(x) = \frac{n}{x} p^x (1-p)^{n-x}$$

2) Binomial (n,p) (1)

For a block data (16 Bytes), there are four data from key and plaintext may collide at input of SBox unit in a single round. Thus one block may collide 40 times in total 10 rounds of AES encryption process. Let calculate collision probability in four data inputs sequentially to the SBox input per block (2), where (3) has shown these four data non-collision probability. (4) is indicated the collision probability where collisions have sequentially occurred at a block in a round (it may occur at max. 4 collisions) and total 10 rounds as well.

Collision probability at a time

$$P_{40} = 40 * \frac{1}{4} = 10$$

$$P_1 = \frac{1}{4}$$

$$P_{X-4}(x) = \frac{4}{4}(\frac{3}{4})^4(1-\frac{3}{4})^{4-4} = \frac{81}{256} \qquad (2)$$

Non-collision probability

$$P_4 = 1 - \frac{81}{256} = \frac{175}{256} \qquad (3)$$

$$P_{40} = 10 * \frac{175}{256} = 6.8$$

Collision probability

AES core throughput calculation

There is also considered the AES core throughput.

Throughput =128/(average clock cycles to process a block x clk period) =128/(160~200) cycles*174MHz = 11~139Mbps

2.4 Timing Analysis

Depending on the home system application and its requirement, architecture of AES core should usually be selected. In order to design appropriate architecture, latency is analyzed to ensure maximum affordable time delay for security encryption in the system.

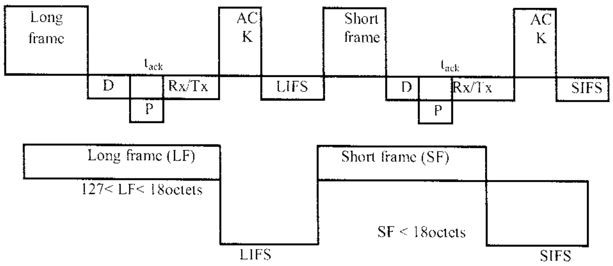

Figure 3. Acknowledged and Unacknowledged transmission

D- RFDelay
P- Propagation Delay
Rx/Tx- TurnAroundTime

Let us analyze whether latency is satisfied by the system requirement, since the architecture uses approx.160 clock latency for a block encryption. Here are system specifications: 250kb/s data rate, 62.5ksymbols/s symbol rate and 22.114MHz system clock (hence period is 45ns). The MAC sublayer needs a finite amount of time to process data received by the PHY. To allow this, transmitted frames shall be followed by an IFS period. The length of the IFS period is dependent on the size of the frame that has just been transmitted regardless the acknowledged or unacknowledged transmission as described in Figure 3.

Table 1. Mac sublayer parameters for timing analysis

MAC sublayer parameters	*Period/ Value*
aMinSIFSPeriod*	12symbols*62.5ksym/s=192μs
aMaxLIFSPeriod*	40symbols*62.5ksym/s =640μs
aMaxMACFrameSize*	aMaxPHYPacketSize-MaxFrameOverhead= 127-25=102 octets
aMaxSIFSFrameSize*	18 octets

Long frame must process within aMaxLIFSPeriod period, short frame process at aMinSIFSPeriod depicted in Table 1.

AES-CCM security latency computation

# of iterations of AES for MIC= (Mac frame + zeropad)/16+ 1 (for B0) , where zeropad satisfies (MAC frame size + zeropad) mod 16=0.

# of iterations of AES for CTR= (data payload size + zeropad)/16+1 (for MIC), where zeropad satisfies (data payload size + zeropad) mod 16=0.

The required number of AES usage for AES-CCM mode shall be calculated by adding of iterations MIC and CTR as shown Table 2. Encryption and decryption modes are identical. The total 10 rounds fluctuate between 160 and 200 clock cycles depending on collision at the shared SBox. Frame size for time latency has been chosen based on critical parameters, such as MaxSIFSFrameSize and aMaxLIFSFrameSize which specify time contraint for data transmission as well as reception.

Table 2. Latency estimation

	Number of iteration			Latency= 160		Latency= 200	
	MIC [times/blk]	CTR [times /blk]	CCM [times/blk]	Clk cycles	Period [μs]	Clk cycles	Period [μs]
18Octets (aMaxSIFSFrame Size*)	3	2	5	800	36	1000	45
127Octets (aMaxLIFSFram eSize)	9	9	18	2880	130	3600	162

The total required time to process frame at MAC sublayer equals to addition of MPDU encapsulation, security encryption, and PPDU encapsulation. In our current system, excluding encryption time the processing time are imperceptible. The AES core with a 160-latency or 200-latency has no time restriction regardless encryption decryption mode or a long or short frame applying into home security system requirement.

Table 3. AES core power consumption

Hierarchy	Switch power	Internal power	Leakage Power	Total power	%
AES	5.68e-04	1.89e-03	7.35e-06	2.46e-03	100
SBox	1.69e-04	2.12e-04	2.39e-06	3.83e-04	15.6
BytePer mu-tation	8.14e-05	4.31e-04	1.14e-06	5.14e-04	20.9
KeyExpan-sion	1.28e-04	6.21e-04	1.93e-06	7.50e-04	30.5
MixColumns	1.11e-04	2.73e-04	6.51e-07	3.85e-04	15.6
Parallel-Serial Converter	2.86e-05	1.36e-04	4.74e-07	1.66e-04	6.7

According to Table 3, the most switching power is consumed by the SBox unit, since every clock it switches for data and key substitution. The SBox unit also consumes the biggest leakage power, which is dependent on the voltage, temperature and state of transistors. KeyExpansion unit consumes the majority of the internal power of the core caused by the charging the internal loads as well as by-the short-circuit current between transistors of a gate.

3. Result Section

AES-CCM security hardware engine is synthesized in using 0.13um Donggbu standard cell library. It is simulated the operating conditions: 1.08V and 25°C. As a result from the synthesis, AES core has 3.9K and AES-CCM has 7.6K gate counts. Data arrival times are 5.74ns for AES core, 6.38ns for AES-CCM respectively. The most area of the AES- CCM implementation consumed for keeping state values in registers including 128-bit input, output and intermediate registers. Similarly, the registers and control multiplexers consume the majority of the area of the AES core.

As shown in Table 4, SBox and KeyExpansion units account for 60% of the total area. The KeyExpansion unit consumes the highest percentage of the power consumption with respect to other units. Furthermore, BytePermutation, SBox and MixColumns units consume the rest of power equally, since these entire units have switching activities caused by glitches of SBox collision.

Table 4: Area and power consumption distribution of the AES core

Components	Area (%)	Power (%)
SBox	28	16
BytePermutation	18	21
KeyExpansion	29	30
MixColumns	9	16
Parallel-Serial Converter	7	7
Control	9	10

Table 5 shows the switching power consumption of the proposed architecture with respect to the [4].

Table 5. Power consumption comparison table

Components	Panu.H et [4]		Proposed	
	uW/MHz	(%)	uW/MHz	(%)
SBox	13.2	35.7	4.2	16.0
BytePermutation	4.4	12.0	5.5	21.0
KeyExpansion	7.1	19.1	7.8	30.0
MixColumns	8.4	22.6	4.2	16.0
Parallel-Serial Converter	1.3	3.6	1.7	7.0
Control	2.6	7.0	2.6	10.0
Total	37.0	100.0	26.0	100.0

The SBox's switching power consumption has reduced twice, while conserving the power to access to the entire memory, it access only to a targeted SLUT in the shared SBox architecture. Due to some control logics caused by glitches from the collision at input of SBox unit, percentages of other unit's power distribution have increased despite its absolute values are still lower.

Table 6 illustrates AES hardware implementations' results. Crucial parameters for the system requirement such as gate counts, frequency, cycles per block as well as power are selected in terms of AES core design analysis.

Table 6. AES hardware implementations (enc)

No	Proposed	Panu.H et. [4]	Rijmen et. [5]
Gates [kgates]	3.9	3.4	3.4
Data bus	8	8	8
Max.freq [MHz]	174	152	80
Cycles/ block	160~200	160	1032
Throughput [Mbps]	111~139	121	10
Power [μW/MHz]	26	37	45
Process [μm]	0.13	0.13	0.13

The proposed architecture has less gate counts comparing with 32 or 128-bit data width architectures, however consumes more area than other 8-bit data width architectures. The proposed architecture has very reasonable throughput and the lowest power consumption with respect to all other referenced architectures. Throughput may vary from 111~139Mbps depending on data collision at the input of SBox unit. Cycles per block are still reasonable with respect to other architectures.

4. Conclusions

In order to find a proper architecture for comprehensive home security implementation, the required maximum latency period has been analyzed. According to the latency constraint, 8-bit computation based on AES architecture has been developed. The most effective current AES architecture proposed by H.Panu uses two SBox LUTs for data ciphering and key scheduling. However this paper proposes a shared SBox scheme for both data ciphering and key scheduling while maintaining sub-divided LUTs instead of one 16x16 LUT.

The proposed hardware design of AES-CCM security suit for the home security is implemented with Verilog HDL and synthesized in 0.13um Donggbu standard cell library. As a result from the synthesis, AES core has 3.9K and AES-CCM has 7.6K gate counts. The collision, timing and power analysis has completed. Design verification is performed using simulation-based verification. The proposed architecture saves 30% power than the referenced architecture with the cost of 17% of additional latency.

5. References

[1]IEEE. Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LRWPAN), IEEE Std. 802.15.4, 2003.

[2]Naven Sastry, David Wagner, "Security Considerations for IEEE 802.15.4 Networks" , ACM Press, New York, NY, USA, Pages: 32 – 42, ISBN:1-58113-925-X ,2004

[3]National Institute of Standards and Technology (NIST). Advanced Encryption Standard (AES), 2001. FIPS-197

[4]Panu Hamalainen, Timo Alho, Marko Hannikainen, and Timo D. Hamalainen. Efficeint Hardware Implementation of Security Processing of IEEE 802.15.4 Wireless Networks, 0-7803-9197-7/05, 2005, IEEE

[5]M. Feldhofer, J. Wolkerstorfer, and V. Rijmen. AES implementation on a grain of sand. IEEE Proc. Inf. Secur., 152(1):13–20, 2005