FPGA based highly efficient MISTY1 architecture

Yasir[1a)], Ning Wu[1], Xin Chen[1], Muhammad Rehan Yahya[1], and Xiaoqiang Zhang[2]

[1] *College of Electronic and Information Engineering,*
Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

[2] *College of Electrical Engineering, Anhui Polytechnic University,*
Wuhu 241000, China

a) *khizaryasir@yahoo.com*

Abstract: This letter proposes highly efficient MISTY1 8-rounds pipelined architecture for wireless networks. A novel methodology is adopted for implementation of MISTY1 substitution functions by optimizing S9 and S7 LUTs (Look-Up Tables) to minimize the silicon area. Besides, a key module FI function is compliant to double edge-trigger the optimized S9 LUTs. This leads to substantial reduction in the pipeline requirements for the proposed hardware architecture. For path delay reduction, logic modifications are made in FI and FO functions realizing efficient and high-speed MISTY1 implementation. FPGA implementation on Xilinx FPGA, Virtex 7 xc7vx690t yielded a throughput value of 16.3 Gbps covering area of 1265 CLB slices.

Keywords: MISTY1, RAM, FPGA, LUTs

Classification: Integrated circuits

© IEICE 2017
DOI: 10.1587/elex.14.20170841
Received August 9, 2017
Accepted August 28, 2017
Publicized September 8, 2017
Copyedited September 25, 2017

References

[1] M. Matsui: “New block encryption algorithm MISTY,” Springer FSE LNCS **1267** (1997) 54 (DOI: 10.1007/BFb0052334).

[2] Yasir, *et al.*: “Highly optimised reconfigurable h/w arch. of 64 bit block ciphers MISTY1 and KASUMI,” Electron. Lett. **53** (2017) 10 (DOI: 10.1049/el.2016.3982).

[3] D. Yamamoto, *et al.*: “Comp. arch. for ASIC implementation of MISTY1 block cipher,” IEICE Trans. Electron. **E93-A** (2010) 3 (DOI: 10.1587/transfun.E93.A.3).

[4] Yasir, *et al.*: “Compact hardware implementations of MISTY1 block cipher,” J. Circuits Syst. Comput. (2017) 27 (DOI: 10.1142/S0218126618500378).

[5] A. Rjoub and E. M. Ghabashneh: “Low power/high speed optimization approaches of MISTY algorithm,” IEEE 5th Intl. Conf. on Elect. Devices, Syst. and Applications (2016) 4 (DOI: 10.1109/ICEDSA.2016.7818520).

[6] D. Yamamoto, *et al.*: “Compact arch. for ASIC and FPGA imp. of KASUMI block cipher,” IEICE Trans. Electron. **E94-A** (2011) 2628 (DOI: 10.1587/transfun.E94.A.2628).

[7] P. Kitsos, *et al.*: “Arch. and FPGA implementation of 64 bit MISTY1 block cipher,” J. Circuits Syst. Comput. **15** (2006) 817 (DOI: 10.1142/S0218126606003362).

[8] A. F. Martínez-Herrera, *et al.*: “GCM implementations of Camellia-128 and

SMS4 by optimizing the polynomial multiplier," Microprocess. Microsyst. **45** (2016) 129 (DOI: 10.1016/j.micpro.2016.04.006).

[9] A. Soltani and S. Sharifian: "An ultra-high throughput and fully pipelined implementation of AES algorithm on FPGA," Microprocess. Microsyst. **39** (2015) 480 (DOI: 10.1016/j.micpro.2015.07.005).

[10] Q. Liu, *et al.*: "High throughput and secure AES on fpga with fine pipelining and enhanced key expansion," IET Comput. Digit. Tech. **9** (2015) 175 (DOI: 10.1049/iet-cdt.2014.0101).

[11] G. Rouvroy, *et al.*: "Efficient FPGA imp. of block cipher MISTY1," IEEE Proc. 17th Int. Parallel and Dist. Processing Symposium (2003) 7 (DOI: 10.1109/IPDPS.2003.1213343).

1 Introduction

MISTY1 is a NESSIE approved 64-bit block cipher developed by Mitsubishi Electric [1]. Standardized by ISO/IEC, MISTY1 falls in a third security level called as "normal-legacy" designed for smaller data blocks of 64-bits or less e.g. payments with 8-byte passwords. It is proven to be secure against "Linear and Differential Cryptanalysis" having probability value of 2^-56. Therefore, MISTY1 block cipher is widely used for wireless sensor networks, mobile communications, online transactions and ATMs.

The design and optimization of cryptographic algorithms have been studied in detail keeping in view the application requirements for low area, high speed or achieving a trade-off between area and speed [2, 3, 4, 5, 6, 7, 8, 9, 10, 11]. For low area design, the commonly adopted methods include re-utilization of logic blocks and s-boxes optimization [2, 3, 4, 5, 6, 7]. The area-efficient implementation techniques have widely been adapted to a feistel-like MISTY1 structure using a single FI/FO function for embedded applications [2, 3, 4]. The compact MISTY1, however are highly unsuitable for high speed applications having low throughput values. Contrary to area-efficient design schemes, encryption algorithms including AES, KASUMI, CAMELLIA and MISTY1 employ RAMs/LUTs/combinational logic to substitute s-boxes using pipe-lined architecture for high speed implementations [7, 8, 9, 10, 11]. It is found that the non-optimized high-speed architectures implementing straight-forward pipelines require large area thus reducing the efficiency [7, 11]. In this regard, our study is mainly focused on the efficient implementation of MISTY1 having salient features as under:

- Area optimization of S9 and S7 s-boxes.
- Efficient implementation of FI function by employing a double edge-triggered technique in the data path of S9 substitution function.
- Design and implementation of MISTY1 architecture with logic modifications in FI and FO functions for reduction in path delay.

2 Optimized S9 and S7 s-boxes

A comprehensive analysis on MISTY1 s-boxes revealed that the algebraic expressions of S9 and S7 can be decomposed into branched LUT structure such that each expression y_i of S9 and S7 is formulated as 5-bit, 4-bit or 3-bit input LUTs as

DOI: 10.1587/elex.14.20170841
Received August 9, 2017
Accepted August 28, 2017
Publicized September 8, 2017
Copyedited September 25, 2017

described in Table I. The output y_i is obtained by 'XORING' the LUTs output as given by eq. (1).

$$y_i = LUT1 \oplus LUT2 \oplus LUT3 \tag{1}$$

The primary advantage of transforming S9/S7 mathematical expressions into 3× LUTs is that it does not affect the path delay of FI function (described in detail in Section 3). Moreover, by reducing the depth using maximum 5-bit input LUTs, the hardware area is reduced considerably. Table II shows the area reduction of 48.39% with the proposed LUTs as compared to 9-bit and 7-bit LUTs for S9 and S7 respectively mentioned in MISTY1 specifications [1].

Table I. LUTs of S9 and S7 substitution functions

Sub Ftn	yi	LUT1	LUT2	LUT3
S9	y_9	$x_9x_5 \oplus x_9x_4 \oplus x_8x_4$	$x_8x_3 \oplus x_7x_3 \oplus x_7x_2 \oplus 1$	$x_6x_2 \oplus x_6x_1 \oplus x_5x_1$
	y_8	$x_9x_7 \oplus x_6 \oplus x_8x_6 \oplus x_7x_6$	$x_9x_3 \oplus x_7x_3 \oplus x_9x_1 \oplus 1$	$x_6x_5 \oplus x_5x_4 \oplus x_2 \oplus x_6x_1 \oplus x_4x_1$
	y_7	$x_9x_8 \oplus x_8x_6 \oplus x_5 \oplus x_9x_5 \oplus x_7x_5 \oplus x_6x_5$	$x_9x_3 \oplus x_5x_4 \oplus x_4x_3$	$x_8x_2 \oplus x_6x_2 \oplus x_1$
	y_6	$x_9 \oplus x_8x_7 \oplus x_7x_5 \oplus x_7x_1 \oplus x_5x_1$	$x_8x_4 \oplus x_6x_4 \oplus x_5x_4 \oplus x_4$	$x_8x_2 \oplus x_4x_3 \oplus x_3x_2$
	y_5	$x_9x_6 \oplus x_7x_6 \oplus x_9x_4 \oplus x_3 \oplus x_6x_4 \oplus x_4x_3 \oplus x_7x_3$	$x_8 \oplus x_7x_1$	$x_5x_3 \oplus x_3x_2 \oplus x_2x_1$
	y_4	$x_9x_6 \oplus x_8x_5 \oplus x_8x_3 \oplus x_6x_5 \oplus x_5x_3$	$x_9x_1 \oplus x_6x_2 \oplus x_2 \oplus x_2x_1$	$x_7 \oplus x_4x_2 \oplus x_3x_2$
	y_3	$x_9x_8 \oplus x_8x_5 \oplus x_9x_1 \oplus x_5x_1 \oplus x_1$	$x_6 \oplus x_3x_1 \oplus x_2x_1 \oplus 1$	$x_7x_4 \oplus x_7x_2 \oplus x_5x_4 \oplus x_4x_2$
	y_2	$x_9x_8 \oplus x_8 \oplus x_8x_7 \oplus x_7x_6$	$x_9x_5 \oplus x_9x_2 \oplus x_5x_2 \oplus x_3x_2$	$x_8x_3 \oplus x_8x_1 \oplus x_6x_3 \oplus x_4 \oplus 1$
	y_1	$x_9 \oplus x_9x_8 \oplus x_8x_7 \oplus x_9x_4 \oplus x_7x_4 \oplus x_4x_3$	$x_9x_2 \oplus x_6x_1 \oplus x_9x_1 \oplus 1$	$x_6x_3 \oplus x_5 \oplus x_3x_1$
S7	y_7	$x_7 \oplus x_6x_4 \oplus x_6x_2 \oplus x_7x_6x_1 \oplus x_7x_2x_1 \oplus x_4x_2x_1$	$x_7x_5x_2 \oplus x_5x_1 \oplus 1$	$x_7x_4x_3 \oplus x_3x_2$
	y_6	$x_7x_5 \oplus x_7x_3 \oplus x_7x_1 \oplus x_5x_3x_2 \oplus x_7x_2x_1 \oplus x_1$	$x_4x_3 \oplus x_4x_1 \oplus x_5x_4x_1$	$x_6x_3x_1 \oplus x_6x_2 \oplus 1$
	y_5	$x_6x_5 \oplus x_7x_5x_4 \oplus x_3 \oplus x_6x_3 \oplus x_7x_6x_3$	$x_7x_3x_2 \oplus x_4x_3x_2 \oplus x_7x_2 \oplus x_4x_1 \oplus x_7x_4x_1$	$x_6x_1 \oplus x_3x_1 \oplus x_5x_3x_1$
	y_4	$x_7 \oplus x_6 \oplus x_7x_6x_5 \oplus x_7x_4 \oplus x_5x_1 \oplus x_6x_4x_1$	$x_5x_3 \oplus x_6x_3x_2$	$x_7x_3x_1 \oplus x_2x_1 \oplus 1$
	y_3	$x_5x_4 \oplus x_2 \oplus x_5x_2 \oplus x_7x_4x_2$	$x_7x_3 \oplus x_6x_4x_3 \oplus 1$	$x_6x_5x_2 \oplus x_6x_1 \oplus x_6x_2x_1 \oplus x_3x_2x_1$
	y_2	$x_7 \oplus x_6 \oplus x_5 \oplus x_7x_6x_5 \oplus x_7x_4 \oplus x_6x_5x_4 \oplus x_6x_3 \oplus x_7x_5x_3$	$x_7x_2 \oplus x_7x_6x_2 \oplus x_4x_2$	$x_7x_1 \oplus x_5x_2x_1$
	y_1	$x_7x_6 \oplus x_4 \oplus x_7x_4 \oplus x_7x_2 \oplus x_4x_2 \oplus x_6x_4x_2 \oplus x_6x_1 \oplus x_7x_4x_1$	$x_5x_2 \oplus x_6x_5x_1 \oplus x_5x_2x_1$	$x_5x_4x_3 \oplus x_3x_1$

Table II. Reduction in area (occupied slices) implemented on Xilinx Virtex 7

Function	Methodology	LUTs						Area (S9+S7)	Reduction (%)
		As per Table I			As per Eq. (1)	Ref. [1]			
		3-1	4-1	5-1	3-1	7-7	9-9		
S9	Straight - Forward	-	-	-	-	-	1	31	48.39
S7		-	-	-	-	1	-		
S9	Proposed	1	19	7	9	-	-	16	
S7		-	13	8	7	-	-		

© IEICE 2017
DOI: 10.1587/elex.14.20170841
Received August 9, 2017
Accepted August 28, 2017
Publicized September 8, 2017
Copyedited September 25, 2017

3 FI function implementation

MISTY1 straight-forward FI function and the proposed FI function employing optimized s-boxes are depicted in Fig. 1a and 1b respectively. It is evident that the proposed FI function is executed on a single clock cycle triggering adjacent (upper and lower) S9 LUTs on positive and negative clock-edges respectively. This methodology differs from old implementations consisting of only positive-edge triggered pipe-lines thereby requiring multiple clock cycles [7, 11]. Furthermore, KI_{IJR} XOR is performed after zero extension (Z) for path delay reduction [3]. In order to maintain logic equivalency, KI_{IJR} XOR is also performed on the right most 7-bits after S7 LUTs execution. The symbol 'T' in Figs. 1a and 1b denotes the truncate operation of 2× MSB bits. Thus, the optimized S9/S7 LUTs in concatenation with KI_{IJR} XOR modification results in an efficient FI function implementation. The path delay of FI function can be expressed as eq. (2):

$$\text{Path Delay (FI)} = T/2 = T_{C2Q\ (S9)} + 2T_{P\ (XOR)} + T_{P\ (S9)} + T_{Setup\ (S9)} \tag{2}$$

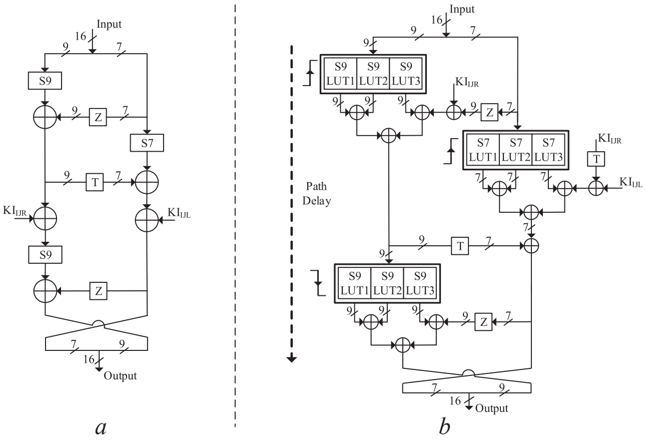


Fig. 1. FI Function

4 Optimized implementation of FO function

The FO function and its sub functions FI_1, FI_2 and FI_3 are illustrated in Figs. 2a, 2b, 2c and 2d respectively. The FO function is configured to perform parallel operations such that FI_1/FI_2 are performed in first clock cycle and FI_3/32-bit XOR are carried out in second clock cycle. The design comprehends low area consisting of 2 × 16-bit pipe-lines in FO and 1 × 7-bit pipeline in FI_1, FI_2 and FI_3 functions. FI_1 function consists of XOR operation preceded by FI whereas FI_2 is a simple FI function. Lastly, FI_3 assembles FI function, XOR operation after FI and 16-bit LSBs XOR operation i.e. the last operation of each round. A 32-bit XOR implementation within FO function formulizes the path delay given as eq. (3):

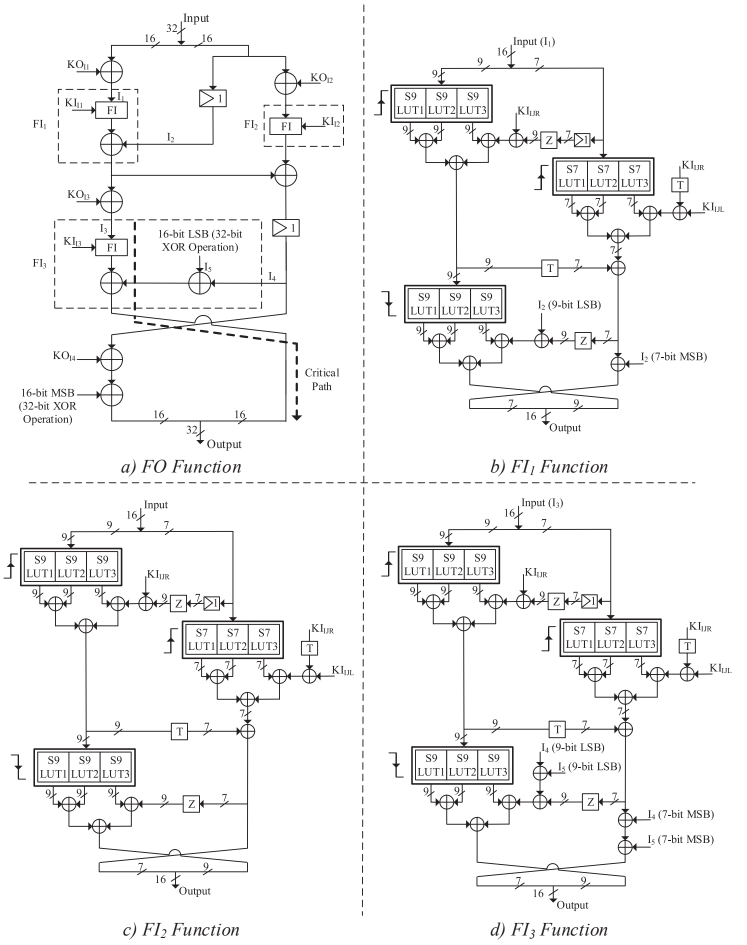
$$\begin{aligned}\text{Path Delay (FO)} &= \text{Path Delay (FI)} = T/2 \\ &= T_{C2Q\ (S9)} + 2T_{P\ (XOR)} + T_{P\ (S9)} + T_{Setup\ (S9)}\end{aligned} \tag{3}$$

5 The proposed MISTY1 architecture

MISTY1 8-rounds pipe-lined architecture is shown in Fig. 3 generating 64-bit cipher text after every clock cycle. A 2-stage pipe line register is inserted in 1st

© IEICE 2017
DOI: 10.1587/elex.14.20170841
Received August 9, 2017
Accepted August 28, 2017
Publicized September 8, 2017
Copyedited September 25, 2017

round followed by 4-stage and 5-stage pipe-line registers in even and odd rounds respectively for synchronizing LSBs/MSBs. A highly efficient MISTY1 is thus obtained with the optimized FO and FI functions implementations.



a) FO Function

b) FI_1 Function

c) FI_2 Function

d) FI_3 Function

Fig. 2. Proposed FO Function and FI_1, FI_2 and FI_3 Functions

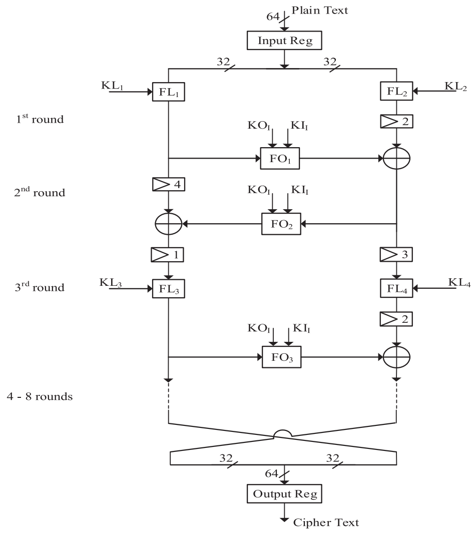


Fig. 3. MSITY1 8-rounds Pipe-lined Architecture

DOI: 10.1587/elex.14.20170841
Received August 9, 2017
Accepted August 28, 2017
Publicized September 8, 2017
Copyedited September 25, 2017

Table III. Results and analysis (*: Published Results; $^{\#}$: Implemented by Authors)

Ref.	Algorithm	FPGA Device	Area (Slices)	Throughput (Gbps)	Freq. (MHz)	Efficiency (Mbps/Slices)
[6]	KASUMI	xcv300e-8bg432*	332	0.044	39.4	0.134
[9]	AES	xc6vlx240t*	35,328	260	508	7.36
[10]	AES	xc7vx690t*	4339	75.92	593	17.50
[8]	CAMELLIA	xc5vlx330*	2805	28.4	221.6	10.12
[2]	MISTY1	xc7vx690t*	487	0.21	248.6	0.43
[7]	MISTY1	xcv400ebg432-8*	1865	0.56	79	0.3
	MISTY1	xcv1000bg560-6*	4732	7.2	96	1.52
		xc7vx690t$^{\#}$	2920	21.9	342	7.5
	MISTY1	xcvII3000bf957-6*	4039	12.6	168	3.12
		xc7vx690t$^{\#}$	2920	21.9	342	7.5
[11]	MISTY1	xcv1000bg560-6*	6322	10.18	159	1.61
[11]	MISTY1	xcvII2000bg575-6*	6322	19.4	303	3.07
Prop.	MISTY1	xc7vx690t$^{\#}$	1265	16.3	254.5	12.9

6 Results and comparison

Table III summarizes the performance analysis of proposed architecture implemented on Xilinx Virtex 7, xc7vx690t.

A throughput value of 16.3 Gbps was obtained with CLB slices of 1265 achieving efficiency of 12.9 Mbps/slices. The remarkable results are the outcome of decomposed s-boxes and FI/FO function optimizations with fine placement of pipe lines. We also evaluated our circuit design with AES, KASUMI, and CAMELLIA and found that our design has the 2nd highest efficiency. Furthermore, we implemented ref. [7] (i.e. MISTY1 architecture claimed as the most efficient) with our FPGA device under the same environment for fair comparison. The parametric values of area, throughput and efficiency were obtained as 2920 CLB Slices, 21.9 Gbps and 7.5 Mbps/slices respectively thus proving our proposed design to be highly efficient and the 3rd fastest MISTY1 architecture till date.

7 Conclusion

This letter presents MISTY1 pipe-lined architecture characterizing efficient implementation. A double edge-triggered methodology employing optimized LUTs for S9/S7 enabled a single clock cycle operation of FI function thereby reducing the area. The logic modifications for path delay reduction resulted in high throughput implementation of MISTY1. A highly efficient MISTY1 architecture is well-suited for wireless networks and mobile computing.

Acknowledgments

Supported by National Natural Science Foundation of China (Nos. 61774086, 61376025, 61404087), Natural Science Foundation of Jiangsu Province (No. BK20160806) & Aeronautical Science Foundation of China (No. 20152052025).

DOI: 10.1587/elex.14.20170841
Received August 9, 2017
Accepted August 28, 2017
Publicized September 8, 2017
Copyedited September 25, 2017