IMPLEMENTATION OF PERMUTATION FUNCTION OF ASCON CIPHER USING VERILOG

Bibhore Goswami
Department of Electronic System Engineering
Indian Institute of Science
Bengaluru, India
bibhoreg@iisc.ac.in

***Abstract*—This report discusses permutation function of the ASCON cipher based encryption algorithm and the power timing analysis of its hardware implementation. Different Architectural Implementation shows the trade off between area, power and time which will be discussed in the subsequent sections.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Ascon uses a duplex-sponge-based mode of operation for authenticated encryption. The recommended key, tag and nonce length is 128 bits. The sponge operates on a state of 320 bits, with message blocks of 64 or 128 bits. The encryption process is split into four sections- initialization, Associated data processing, plaintext processing, finalization. In each stage, specified inputs are fed to a p function with a particular no. of iteration. This gives an absolute random looking output for each stage , as shown in 9.

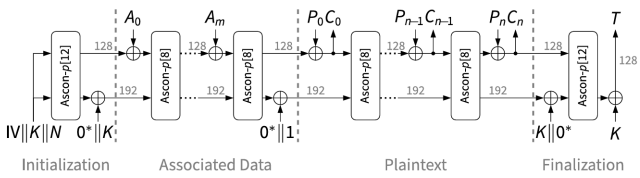


Fig. 1. ASCON cipher implementation

II. Permutation function

The heart of the ASCON cipher encryption algorithm is Permutation function or P function. It takes a 320 bit input and processes it to generate a random looking 320 bit output. The application of the P function can be divided into three sections- (a) round constant addition (b) substitution layer with 5 bit s-box (c) linear layer with 64 bit diffusion. Althogh the verilog code, discussed in this paper has a little different approach towards the implementation of the P function.

Identify applicable funding agency here. If none, delete this.

A. Input verilog module

The design begins with the input verilog module, which takes a 320-bit input and splits it into five 64-bit segments, assigning each to separate outputs. These outputs form the initial state of the data before the transformation process begins.

B. Column diffusion module

Following this, the Column diffusion module processes the five 64-bit inputs by combining corresponding bits from each input to form 5-bit vectors. A pre-defined non-linear mapping function is then applied to each of these 5-bit vectors. One of the inputs also undergoes an XOR operation with an 8-bit round constant before being processed, adding a layer of variability to the transformation.

Table 5: Ascon's 5-bit S-box $\mathcal{S}$ as a lookup table.

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f	10	11	12	13	14	15	16	17	18	19	1a	1b	1c	1d	1e	1f
$\mathcal{S}(x)$	4	b	1f	14	1a	15	9	2	1b	5	8	12	1d	3	6	1c	1e	13	7	e	0	d	11	18	10	c	1	19	16	a	f	17

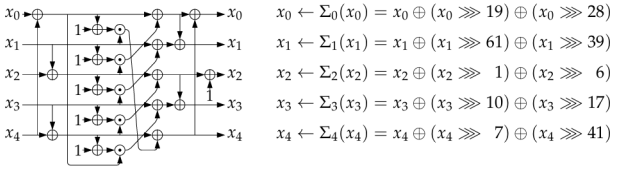
Fig. 2. Column diffusion layer

C. row diffusion module

Next, the row diff module performs a bit wise permutation on each 64-bit input using rotations and XOR operations. This step further scrambles the data to enhance diffusion, ensuring that small changes in the input propagate widely across the output. The output of this module is a 320-bit vector formed by concatenating the transformed 64-bit segments.

D. P function module

The P function module connects these three components (input verilog, Column diffusion, and row diff) into a single stage of transformation. It takes a 320-bit input, splits it into segments, applies column and row diffusion, and outputs a new 320-bit vector. This comprehensive transformation ensures that each bit of the input influences multiple bits of the output, creating a highly non-linear mapping.



(a) Ascon's 5-bit S-box $\mathcal{S}(x)$ (b) Ascon's linear layer with 64-bit functions $\Sigma_i(x_i)$

Fig. 3. Row diffusion layer

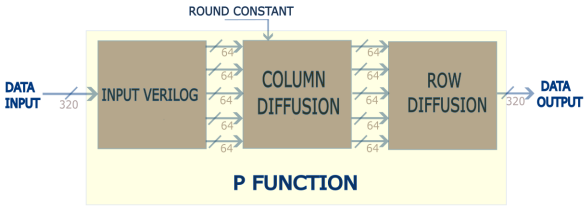


Fig. 4. Block diagram of P function module

E. Iterative implementation of P function module

The final output of the iterative p function module is a 320-bit vector that has undergone multiple rounds of nonlinear transformation, providing strong diffusion and confusion properties suitable for cryptographic applications.

III. Comparative analysis of different architectural implementation

No. of clock cycles	Area (micro meter square)
1	4655.266
2	9259.99
4	18457.474
6	27638.2
12	55214.15 .28

Fig. 5. Area for different architecture

From the results it's clear that the area increases as the area increases although the power remains more or less same at [0.209 miliwatt]. The clock frequency for the operation is 606 MHz.

IV. Outputs of simulation

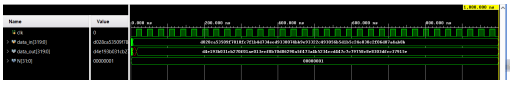


Fig. 6. 12 rounds per cycle

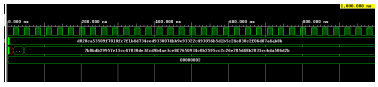


Fig. 7. 6 round per cycle

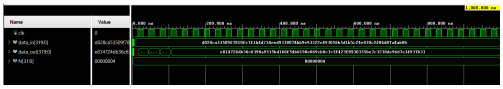


Fig. 8. 3 rounds per cycle

Fig. 9. 2 rounds per cycle

Fig. 10. 1 round per cycle