HIGH THROUGHPUT AND HIGH SPEED BLOWFISH ALGORITHM FOR SECURE INTEGRATED CIRCUITS

Kumara Swamy V., Dr. Prabhu Benakop

Aurora's Engineering College, JNTUH, Bhongir, Nalgonda Dist., Andhra Pradesh, India

ABSTRACT: Information Security is prime focus for current computer data communications. Insecurity in data transmission has increased cybercrimes through hacking. This paper presents four different implementations of Blowfish algorithm and analyzed the performance of it with and without Wave Dynamic Differential Logic (WDDL) style to provide security against Differential Power Analysis (DPA) attack. It compares propagation delay (T_t) and frequency (F) of Blowfish, Modified Blowfish with and without WDDL logic [VBS09, VBS10]. Throughput [S+12, SSS11] of Blowfish (BF) with modified modulo adder and WDDL Logic implementation has 840 Mbps compared to 570 Mbps of that of BF with modulo adder [VD12, TN08] and WDDL logic implementation. This paper is implemented using ModelSim6.1d, Leonardo Spectrum8.1 and Xilinx webpack9.2i with Verilog Hardware Description language.

KEYWORDS: WDDL, SIC, BF and DPA.

I. INTRODUCTION

Plaintext is encrypted to produce cipher-text for data transmission through wired or wireless means. The cipher-text message contains all the information of the plaintext, but is not in a format readable by a human or computer without the proper mechanism to decrypt it. If key length is more, number of iterations is more, the possibility of hacking is less and vice versa. At the encryption we apply plaintext and key as inputs and it produces cipher-text. At the other end, cipher-text and key are the input to decryption and the result is the recovery of original plaintext as shown in the figure no.1. It is a symmetric key algorithm.

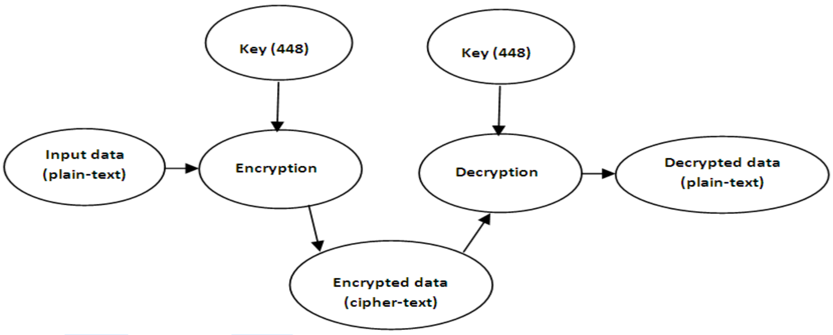

Figure 1: Process of Encryption and Decryption with symmetric Key

Analyzing Data Encryption Standard (DES), Triple DES, Advanced Encryption Standard (AES) and BF, BF algorithm is highly secured and yields high throughput (T_p) with large key size and its chosen as choice of cryptographic algorithm to implement secure ICs [TV04, TV06].

a. Wave Dynamic Differential Logic (WDDL)

WDDL logic is a constant power consumption logic which can overcome the DPA attack by the hacker. It consists of a parallel combination of two positive complementary gates, one calculating the true output using the true inputs, the other the false output using the false inputs. A positive gate produces a zero output for an all zero input. The AND gate and the OR gate are examples of positive gates. The AND gate fed with true input signals and the OR gate fed with false input signals are two dual gates [VBS10].

During the Precharge phase (clk-signal high), the normal and complemented outputs of the digital circuit produce equal outputs. Thus the differential power analysis results in zero differential power to not to allow the hacker to gain the information from the hardware integrated circuits. During evaluation phase (clk-signal low), it generates actual outputs as per logic with correct key.

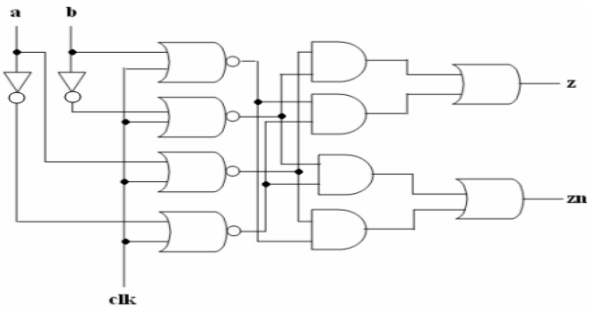

Figure 2: WDDL XOR GATE

In fig no.2, when clock is precharge mode (high), output is zero for both. When clock is evaluation mode (low), outputs are complemented and worked as XOR and XNOR.

b. Blowfish Algorithm

Blowfish is a 64-bit block cipher [AM12, PR12] is the replacement for DES (Data Encryption Standard). DES was the standard cryptographic algorithm for more than 19 years. It has a variable-length key block cipher of up to 448 bits. Although a complex initialization phase is required, the encryption of data is very efficient and highly secured. WDDL can be implemented for any logic design. Since the discussion moves around crypto processors, it would be wise to consider a cryptographic algorithm called Blowfish is a fast algorithm [S+12, VBS09].

II. ANALYSIS OF BLOWFISH ALGORITHM

Blowfish is a symmetric block cipher that encrypts and decrypts data in 8-byte (64-bit) blocks. The algorithm has two parts, key expansion and data encryption. Key expansion consists of generating the initial contents of one array (P-array), namely, eighteen 32-bit sub-keys, and four arrays (the S-boxes), each of size 256 by 32 bits, from a key of at most 448 bits (56 bytes). The data encryption and Decryption uses a 16-round Feistel Network as shown below in fig no.3 and fig no.4 [VBS09, VBS10] respectively.

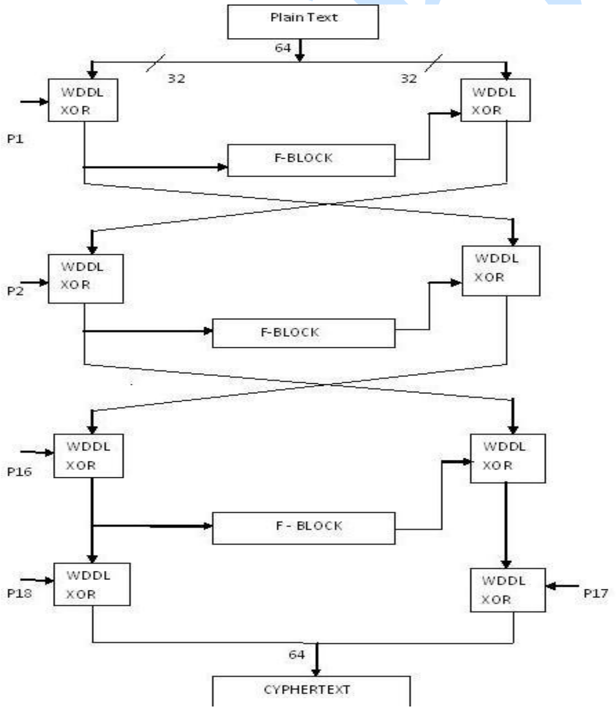

Figure 3: Blowfish Encryption

The encryption algorithm can be defined by the following pseudocode equation no.1:

```
For   $i$= 1 to 16 do
          $RE_i = LE_{i-1} \oplus P_i$;
          $LE_i = F[RE_i] \oplus RE_{i-1}$;
$LE_{17} = RE_{16} \oplus P_{18}$;
$RE_{17} = LE_{16} \oplus P_{17}$;                    (1)
```

The Decryption algorithm can be defined by the following pseudo code equation no.2:

```
For   $i$= 1 to 16 do
          $RD_i = LD_{i-1} \oplus P_i$;
          $LD_i = F[RD_i] \oplus RD_{i-1}$;
$LD_{17} = RD_{16} \oplus P_1$;
$RD_{17} = LD_{16} \oplus P2$;                    (2)
```

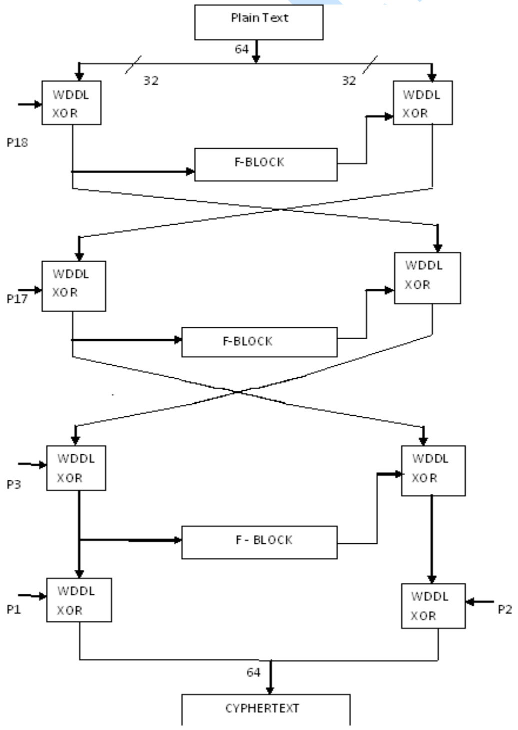

Figure 4: Blowfish Decryption

III. DESIGN OF BLOWFISH ALGORITHM

Encryption consists of sixteen rounds of operations. Each round-one operation consists of xor, 8-bit to 32-bit substitution, 32-bit modulo addition, xor, 32-bit modulo addition and swapping of result of Left Encryption (LE) to Right side and Right Encryption (RE) to left side of the data flow as shown in fig no.3. After performing 16 round-one operations right side output(31:0) xored with subkey p16 (31:0) and left hand side output (31:0) xored with subkey p17 (31:0) and then we get final cipher text(63:0). Decryption is same as that of encryption except we applied sub keys p0 to p17 in reverse order. Input data is the cipher text (output of encryption) and then we get the output as Plaintext. Decryption consists of sixteen- round one operation. Each round-one operation consists of xor, 8-bit to 32-bit substitution, 32-bit modulo addition, xor, 32-bit modulo addition and swapping of result of Left Encryption (LE) to Right side and Right Encryption (RE) to left side of the data flow as shown in fig no.4. The input data ciphertext (63:0) performs 16 round-one operations with 16 subkeys (p17 to 2) and then after performing 16 round-one operations right side output (31:0) xored with subkey p1(31:0) and left hand side output (31:0) xored with subkey p0 (31:0) and then we get final plaintext. A crypto processor implementing Blowfish algorithm may be shown in fig no.5.

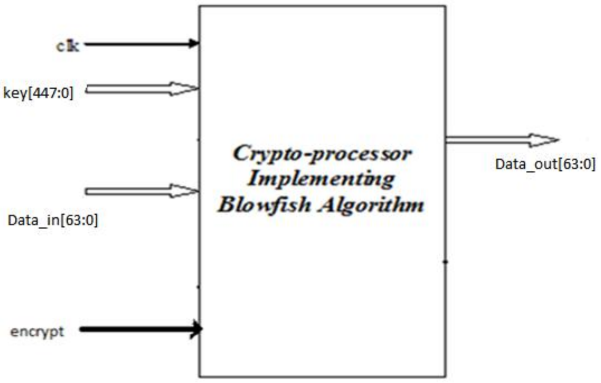

Figure 5: Top level module of Blowfish Crypto-processor

A. Substitution Boxes (S-boxes)

A substitution box (or S-box) is a basic component of symmetric key algorithm used to obscure the relationship between the plaintext and the cipher text In general, an S-box takes some number of input bits, 8-bit, and transforms them into some number of output bits, 32-bit: an 8×32 S-box, implemented as a lookup table [1, 3, 8].

B. Feistel Function Block

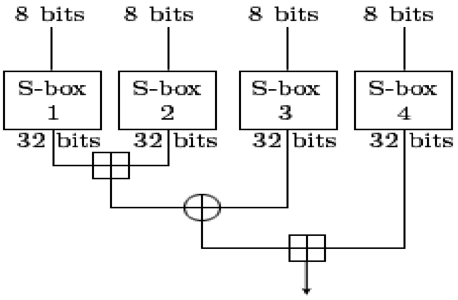

Figure 6: Function Block Internal Structure

Function ‘F’ is used to create ‘confusion’ to thwart cryptanalysis based on statistical analysis. ‘Confusion’ seeks to make the relationship between the statistics of the cipher text and the value of encryption key as complex as possible. One advantage of this model is that the round function F does not have to be invertible, and can be very complex as shown in fig no.6 [AM12, S+12, VBS09].

C. Modulo 32-bit adder

To increase the speed of blowfish adders in this fig no.7 can be operated in parallel. one adder adds Two h-bit residues, X and Y to form their sum $S_1+2^hC_{out1}$ .Another one is 3-operand adder that computes “X+Y+m”. Note that if $m=2^n+1$, we have h=n+1.It has been reported that if either Cout1 or Cout2 of this addition is ‘1’ then the output is X+Y+m instead of X+Y. However, in the following we illustrate that only if the carry of “X+Y+m” is ‘1’, it is sufficient to select it as the final output [VD12, TN08].

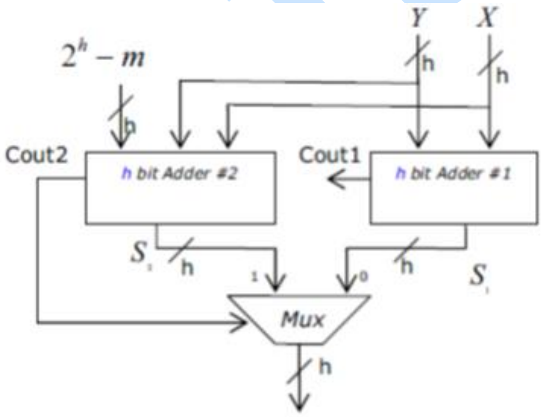

Figure 7: Modulo M –bit adder

D. Sub-key Generation Unit

The sub-key generation unit expands the given 448-bit key into 14 sub-keys and 4 more subkeys are internally generated, each of 32 bits, so that they can be used at different stages in the algorithm. The sub key generation process is designed to preserve the entire entropy of the key and to distribute that entropy uniformly throughout the sub keys. It is also designed to distribute the set of allowed sub keys randomly throughout the domain of possible sub keys. Then bit wise XOR of the P-array and K-array is performed reusing the words from K-array as needed shown in equation no.3:

$$P_1= P_1\wedge K \dots P_{14}= P_{14} \wedge K_{14}$$
$$P_{15}= P_{15} \wedge K_1\dots P_{18}= P_{18}\wedge K_4 \qquad (3)$$

The performance parameter throughput (TP) is defined as number of bits encrypted and decrypted per second.

$$TP = \frac{\text{Number of bits encrypted}}{\text{Propagation delay}} \qquad (4)$$

IV. RESULTS AND DISCUSSION

The encryption and decryption modules are integrated in the top level module to obtain the blowfish crypto-processor and the simulation and synthesis results are analyzed. Comparison is done for four forms i.e., Blowfish (BF), Modified Blowfish (MBF), Blowfish with WDDL (BFWDDL) and Modified Blowfish with WDDL (MBFWDDL) is given below in the table no.1 and the corresponding bar charts are shown in the fig no.8, 9 and 10 for performance parameters Et, Dt and Tt respectively.

Table 1: Comparison of four implementations of Blowfish Algorithm for Et, Dt and Tt

SNo	Name of Crypt-algorithm	Performance parameters		
		Et(ns)	Dt(ns)	Tt(ns)
1	Blowfish	98.663	98.663	99.395
2	Modified Blowfish	70.08	70.08	71.067
3	Blowfish with WDDL	107.62	107.62	112.56
4	Modified Blowfish with WDDL	73.985	73.985	76.337

Et: Encrypt Time, Dt: Decrypt Time, Tt: Total Time

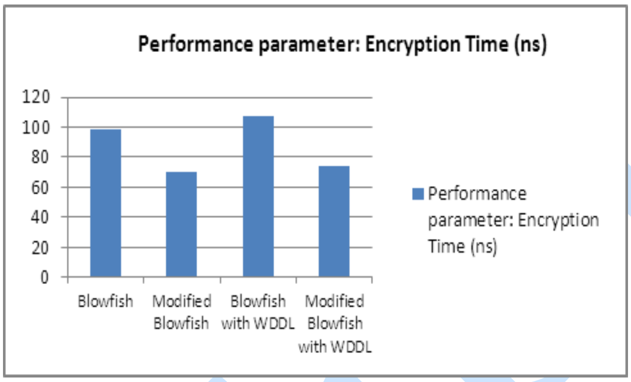

Figure 8: Bar Chart for Performance parameter Encryption Time of four implementations of Blowfish Algorithm

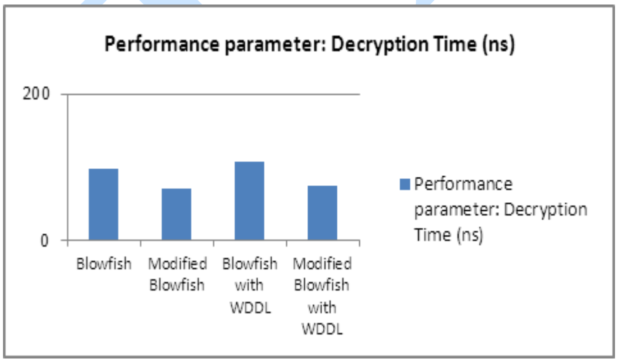

Figure 9: Bar Chart for Performance parameter Decryption Time of four implementations of Blowfish Algorithm

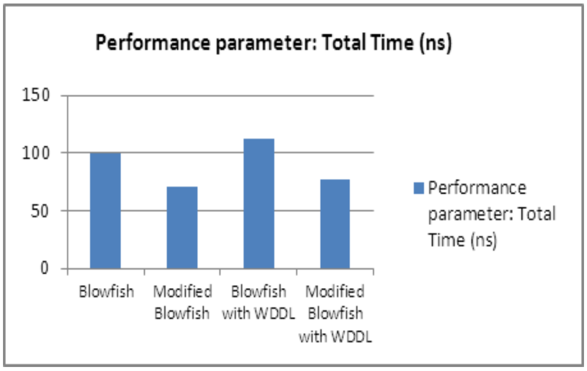

Figure 10: Bar Chart for Performance parameter Total Time (i.e., Propagation Delay) of four implementations of Blowfish Algorithm

Blowfish Algorithm is implemented in four forms and compared its performance parameters which are given below in the table no.2 and the MBF with WDDL is yielded better results in terms of propagation delay (76.337ns) and throughput (840 Mbps) compared to that of BFWDDL with 112.56ns and 570 Mbps respectively . Analysis is done for blowfish with and without WDDL logic to secure the ICs against DPA attack by the hackers.

Table 2: Comparison of four implementations of Blowfish Algorithm for propagation delay, frequency and Throughput

SNo	Name of Crypt-algorithm	Performance parameters		
		Tt(ns)	F (MHz)	TP (Mbps)
1	Blowfish	99.395	10.06	640
2	Modified Blowfish	71.067	14.07	900
3	Blowfish with WDDL	112.56 6	8.884	570
4	Modified Blowfish with WDDL	76.337	13.09	840

Tt: Propagation Delay, F; Frequency, TP: Throughput

Comparison is also shown below in the form of bar charts in fig no.11, 12 and 13 for performance parameters Tt, F and TP respectively.

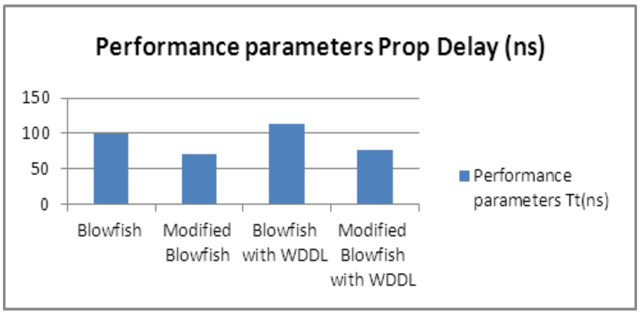

Figure 11: Bar Chart for Performance parameter propagation delay of four implementations of Blowfish Algorithm

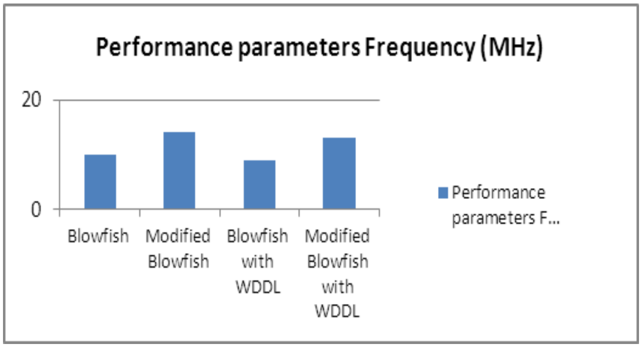

Figure 12: Bar Chart for Performance parameter Frequency of four implementations of Blowfish Algorithm

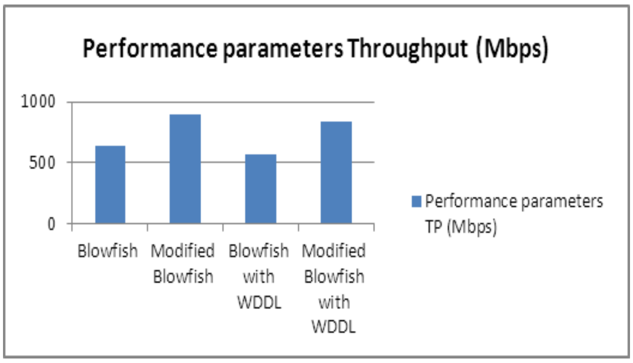

Figure 13: Bar Chart for Performance parameter Throughput of four implementations of Blowfish Algorithm

V. CONCLUSION

In this paper, an implementation of Blowfish Algorithm is designed using WDDL Logic style. In the implementation bottom-up approach is used. The sub-keys generated for a particular key can be used for the encryption of the entire data to be encrypted with that key. The sub keys are given in reverse direction of the decryption data path without changing the design for decryption. The crypto processor has been designed for the key size of 448 bits and plain text of 64 bits. The code for the implementation has been written in Verilog HDL. The functional verification has been done using the ModelSim 6.1d simulation package. The synthesis of the design is done using the Xilinx Web Pack9.2i.Comparison with different implementations has been given in table no.1 and table no.2 and proved that Modified Blowfish and MBF with WDDL logic yielded the best results in delay, frequency and throughput compared to blowfish with and without WDDL logic respectively.

REFERENCES

[AM12] **Monika Agrawal, Pradeep Mishra** - *A Modified Approach for Symmetric Key Cryptography Based on Blowfish Algorithm*, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-1, Issue-6, August 2012.

[L+12] **Chen Liu, Rolando Duarte, Omar Granados, Jie Tang, Shaoshan Liu, Jean Andrian** - *Critical Path Based Hardware Acceleration for Cryptosystems,"* Journal of Information Processing Systems (JIPS), Vol. 8, No. 1, pp.133-144, 2012.

[PR12] **S. Pavithra, E. Ramadevi** - *Study and Performance Analysis of Cryptography Algorithms*, International Journal of Advanced Research in Computer Engineering & Technology Volume 1, Issue 5, July 2012.

[S+12] **Walied W. Souror, Ali E. Taki el-deen, Rasheed Mokhtar-awady Ahmed, Adel Zaghlul Mahmoud** - *An Implementation of High Security and High Throughput Triple Blowfish Cryptography Algorithm*, International Journal of Research and Reviews in Signal Acquisition and Processing (IJRRSAP) Vol. 2, No. 1, March 2012, ISSN: 2046-617X.

[SSS11] **Gurjeevan Singh, Ashwani Kumar Singla, K. S. Sandha** - *Through Put Analysis of Various Encryption Algorithms*, IJCST Vol.2, Issue3, September 2011.

[TN08] **Somayeh Timarchi, Keivan Navi** - *Improved Modulo 2n +1 Adder Design*, International Journal of Computer and Information Engineering 2:7 2008.

[TV04] **K. Tiri, I. Verbauwhede** - *A logic level design methodology for a secure DPA resistant ASIC or FPGA implementation*, in Proc. Design, Automation and Test Eur. Conf. (DATE), Paris, France, 2004, pp. 246–251.

[TV06] **Kris Tiri, Ingrid Verbauwhede** - *A Digital Design Flow for Secure Integrated Circuits*, IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, Vol. 25, No. 7, July 2006.

[VBS09] **V. Kumara Swamy, Prabhu G. Benakop, P. Sandeep** - *Design and Implementation of DPA Resistant Crypto-Processor using Blowfish Algorithm*, International Conference on Advanced Communication and Informatics (ICACI-2009), TPGIT, Vellore, Tamilnadu, India, January 11,12&13th, 2009, pp. 25-32.

[VBS10] **V. Kumara Swamy, Prabhu G. Benakop, B. Sandeep** - *Implementation of digital design flow for DPA secure WDDL crypto processor using blowfish algorithm*, The Libyan Arab International Conference on Electrical and Electronic Engineering (LAICEEE-2010), Tripoli, Libya, October 23-26, 2010, pp. 565-73.

[VD12] **Haridimos T. Vergos, Giorgos Dimitrakopoulos** - *On Modulo 2n +1 Adder Design*, IEEE Transactions on Computers, vol. 61, no. 2, february 2012.