# AN OPTIMAL LOW-POWER/HIGH PERFORMANCE DDP-BASED COBRA-H64 CIPHER

A. Rjoub M. Musameh O. Koufopavlou

Jordan University of Science and Technology
Faculty of Computer and Information Technology
Department of Computer Engineering
22110, P. O. Box 3030.
abdoul@just.edu.jo

Patras University
Department of Computer Engineering
VLSI Design Laboratory
26110 Rio
odysseas@ece.upatras.gr

## ABSTRACT
A new layout design for a data dependent permutation (DDP)-Cobra H64-bit cipher optimized for low-power and high speed operation is presented in this paper. The layout is characterized as a design for mobile and handheld equipment. The design achieves low power consumption by using low power logic gates in layout level. Through the technique of pipelining in the internal rounding blocks of the Cobra cipher and an increase in the frequency of the circuit from 90MHz to 140MHz, the design achieves increased speed and performance, resulting in throughput ranging from 5.5 Gbps to 8.4 Gbps. Simulation results based on the layout level have confirmed the validity of the proposed technique, as well as confirmed that low power, speed and performance can be optimized through design at the layout level.

## Keywords
Cobra-H64, Security, Cryptography, Encryption, Decryption.

## 1. INTRODUCTION
Recently, the demands for low-power, high-speed, secure handheld devices for wired and wireless communication is increasing. The secure communications capability is a common feature of many devices including, cell phones, personal digital assistants, laptops, and biomedical devices [1][2][3]. The wide spread of mobile and portable devices running over their, various multimedia applications demands secured data transfer. One of the main critical points of the mobile equipment is the power dissipation, it is high demanded to have low power algorithms in order to save the life batter in the maximum time.


* Mobimedia'07, Month 8, 2007, Nafpaktos, Aitolokarnania, Greece


The main critical point of Multimedia mobile applications is the power dissipation; this is because the Image, voice video and text are all required enough power resources to keep their quality at maximum without noise.

A review of the relevant literature reveals that low power dissipation techniques have been discussed and analyzed widely in the last few decades [4]. The major techniques that have

improved the efficiency, increased speed of operation and reduced the power dissipation of the devices [4][5] include: reducing the threshold voltage to less than 0.08V, reducing the supply voltage to less than 1 volt at the circuit or transistor level [5], and using parallel and pipelining architectures at the system level.

Cryptographic algorithms have often been implemented using ASIC and FPGA technologies [6][7] are using mainly in multimedia devices. Those technologies are capable of producing high performance circuits through parallel and pipelining techniques [7]. Unfortunately, those technologies are based on standard cell libraries which use a variety of fixed, standard cell components. In most cases, these components have not been designed using techniques that minimize power dissipation and area to maximize speed operation. Consequently, most ASIC or FPGA designs can be optimized further for power, speed, and area.

Cryptographic algorithms based on block ciphers have also been described in recent articles [8][9]. The demand for 64-bit block ciphers in the market is increasing because of their ability to meet the demands of present applications [10][11]. The Cobra H64-bit is a powerful cryptography algorithm that has been shown to be secured against a variety of attacks [10], and to meet the performance demands of a variety of wireless communications network applications. For these applications FPGA and ASIC synthesis tools [7] were used to implement Cobra-H64 and others. Consequently, further optimization of the performance, power dissipation, and area is possible through layout level changes.

In this paper, we optimized the power dissipation and speed operation of Cobra –H64: The power dissipation is optimized by

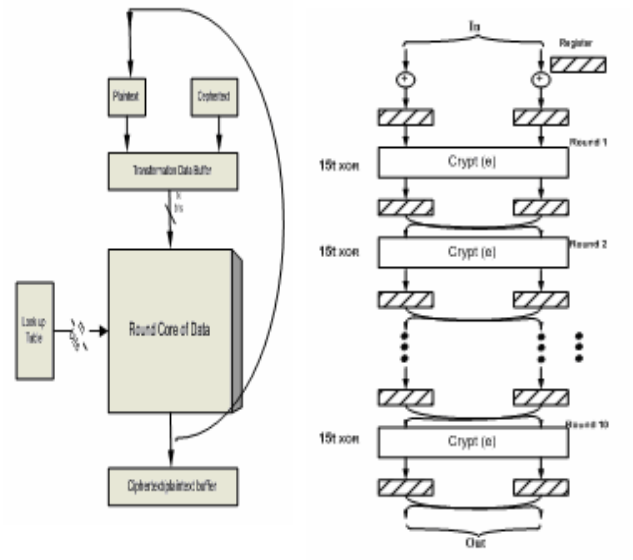
minimizing the layout silicon area through the use of multiplexers instead of logic gates and by choosing complementary pass transistor logic as the logic family. The Speed operation is optimized by using the pipeline architecture in the internal rounding blocks which both reduces power dissipation and increases the speed of operation.

The techniques used to achieve this optimization are described in the remaining sections of this paper. However, in section 2 of this paper, an overview of block cipher architectures is provided, including the advantages and disadvantages of full rolling architectures and pipelining architectures. The main characteristics of Cobra-H64 block ciphers are described in section 3. In section 4, the implementation methodology of the optimized layout designs for most of the parts of the Cobra-H64 algorithm are described. Section 5 discusses the simulation results for the optimized designs.

## 2. BLOCK CIPHERS ARCHITECTURE

Block ciphers are implemented using two different architectures: basic loop architecture and full loop rolling architecture [9]. Figure 1.a illustrates the basic block diagram of the basic loop architecture, while Figure 1.b illustrates the basic block diagram of the full loop rolling architecture.

The basic round unit of Figure 1.a encrypts a piece of plaintext through several cycles of the loop. The main advantages of the basic loop architecture are relatively smaller layout silicon area and lower cost. The main disadvantage of this architecture is that it cannot start encryption of a second piece of plaintext until all the cycles of the loop needed to encrypt a first piece of plaintext have been executed. Accordingly, there is a delay time that is a function of 1) the period of a cycle, $\partial t$, and 2) the number of cycles, m, needed for each round of data. If each round of plaintext takes one clock cycle and each plaintext takes n rounds, it means that m=n and there is a need for n clock cycles to encrypt n-round plaintext.




a: Basic Loop

b: Full loop Rolling

**Figure1: Block Cipher Architectures**

While the full loop architecture has more hardware requirements than the basic loop architecture because it requires multiple hardware round units, it performs n-round plaintext in pipelining saving n-1 clock cycles[11].

Different types of 64-bit and 128-bit block ciphers have been implemented using the foregoing two main architectures described above. Most of them have been simulated using either high level software languages or hardware description languages such as VHDL and Verilog.

## 3. CHARACTERISTICS OF COBRA-H64

Cobra-H64 is a new data dependent permutation based cipher. It consists of ten round blocks, where in each round block a basic operation is repeated. The operations performed in the round blocks guarantee that the cipher is symmetric. If some plaintext generates a cipher text by encryption with a key, then the ciphertext can be decrypted with the same key to generate the original plaintext. The main hardware parts of Cobra H64 are: Control Permutation boxes, DDP (P32/96 , P32/96-1), Extension Box (E), Non-linear operation (G), Permutational Involution (I), and Switchable fixed permutation $\pi$(e). Controlled Permutation (CP) boxes appear to be very efficient cryptographic primitives for fast hardware encryption. A CP box is considered to be a dynamic permutation unit that performs a permutation depending on the values on control lines coupled to the CP box. The main characteristics of Cobra-H64 ciphers use CP-Boxes as the primitive units and use the data itself to drive the control lines; it uses a 128-bit key which is divided into four 32-bit blocks, each one of these blocks are permutated to four rounds, each permutated key is divided into four sub-keys. This simple key scheduling results in high performance especially in the case of frequent key refreshing, since no preprocessing is used to perform key scheduling, the whole secret key is directly used in each round.

The ten rounds of data transformation are performed based on the procedure shown in Figure 2 followed by a Final Transformation. First, the 64-bit data input X is divided into sub-blocks (L) left and (R) right. Then an Initial Transformation is executed which performs XOR-ing between each of the data sub-blocks and two different sub-keys.

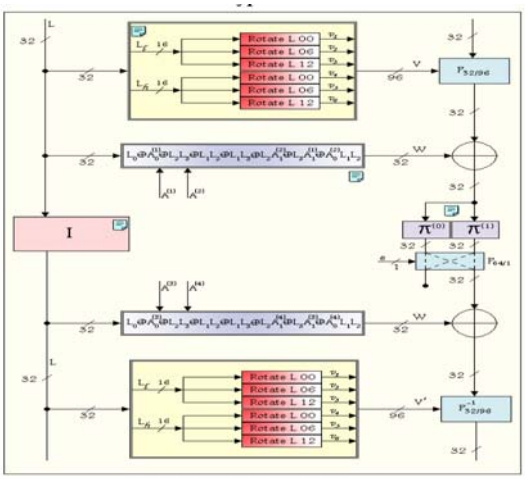



**Figure 2: Proposed procedure of Cryptography**

## 4. IMPLEMENTATION OF THE NEW DESIGN

In order to achieve low power, high speed and low silicon area, a new generation of the Cobra-H64 algorithm is implemented using pipelining architecture to reduce the delay time as well as to increase the throughput as shown in Figure 2. In Figure 2, each bit of sub-block L influences exactly six bits of sub-block R. Such distribution of the controlling bits allows an arbitrary input bit of the boxes [P32/96] and [P-1 32/96] to move each output position with the same probability provided L is a uniformly distributed random variable.

The algorithm has been optimized at the layout level by selecting the proper SPICE parameters including the proper supply voltage and simplifying the design of the critical path including setting the width and length of each transistor in the design. This optimization could not have been achieved if the algorithm were implemented using ASIC or FPGA technologies.

The layered CP-Boxes of Figure 3 are constructed as a superposition of S = 2m/n active layers separated by S–1 fixed permutations $\pi 1,\ldots, \pi S\text{-}1$ that are implemented in hardware with the simple connections. Each active layer in a CP box with an n-bit input is represented by the set of n/2 parallel elementary boxes P2/1.

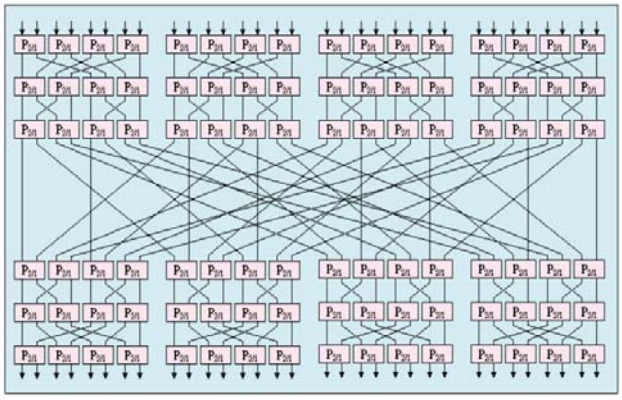


**Figure 3: Structure of the CP-boxes P32/96**

The design can be split into two parts: The first part uses AND gates and one inverter, while the second part uses multiplexers implemented using the complementary pass transistor logic family. The implementation of extension box E is easily achieved by a wiring layout and easily implemented using different metals and layers. At the layout level, we have the opportunity to design a part by using different types of circuits. For example, we can choose between the AND gates and the multiplexers. It can be shown that using multiplexers saves more power than using AND gates.

The implementation of this function could be easily done by using two different ways. The Cobra H-64 can be implemented through parallel technique or a ripple method: In case of parallel gates technique, the delay time is minimized and the power consumption is reduced also, by using six XOR gates in parallel. In the ripple method, the delay time is greater than the delay time in case of parallel as well as the power is more also. Indeed, if ripple method is used, the delay has no significant effect because the delay of P32/96 is only one XOR delay [15]. In our design we used the parallel method to speed up the circuit as well as to decrease the power dissipation. The implementation of non-linear operation is implemented in the layout level so that to achieve the minimum power dissipation as well as layout silicon area.

## 5. PROPOSED ROUNDS IMPLEMENTATION

The core part of Cobra-H64 cipher is the round which can be implemented by two different methods: the first method is the traditional full rolling architecture and the second one is the pipelining architecture. The full rolling architecture is important because it can encrypt/decrypt a data block in ten round times. Each round contains a combinational logic circuit and is supported by a 64-bit register and 64-bit multiplexer. Both of them were implemented in ASIC and FPGA and simulation results showed the validity of the proposed technique.

The second proposed architecture, Figure 4, is 10-stage pipeline architecture. The pipelining architecture offers the benefit of the high-speed performance and can be applied in applications with demanding hardware throughput needs. To support the pipelining technique, look up tables are used for the round keys storage and loading which are pre-computed. The proposed architecture uses 10 basic round blocks for Cobra-H64, which are cascaded by using an equal number of pipeline registers. Based on this approach, 10 different 64-bit data blocks can be processed at the same time. The pipeline proposed architecture produces a new plaintext/ciphertext block every clock cycle, it is already reduced the delay time and increased the performance comparing with the full rolling method.

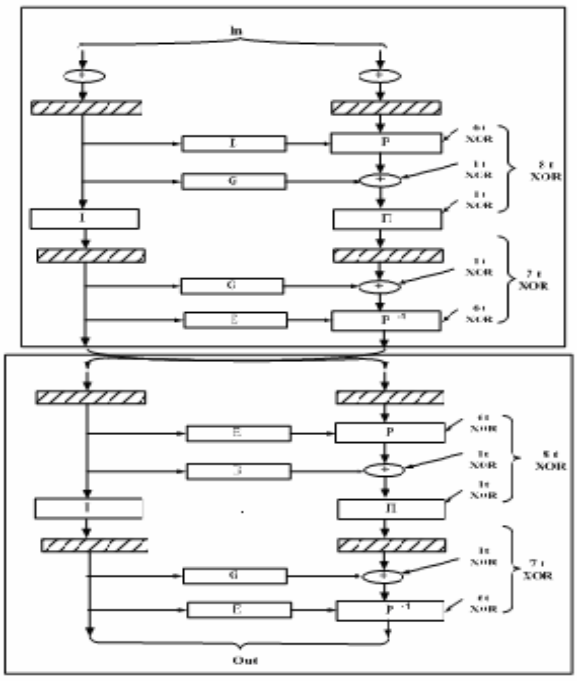



**Figure 4: Proposed Pipeline architecture**

The optimization of speed and therefore the operation should include optimizing many aspects of the design in order to

increase the speed and enhance the performance. In [10] they succeeded in increasing the performance by inserting the pipelining technique between the rounds. Simulation results have demonstrated the validity of the technique. In our optimized design we use the pipelining technique within the rounds themselves, and refer to it as internal pipelining. Internal pipelining enhances the speed of operation internally and provides better performance. To implement pipelining within a round itself, we divided the round block into two parts. The first part contains six XOR gates, where the output of each XOR gate is an input to a buffer register and the output of the buffer register is an input to the next four XOR gates of the next stage. While the second block contains another four XOR gates as illustrated clearly in Figure 4.

## 6. DISCUSSION AND SIMULATION RESULTS

The Cobra-H64 cipher is proposed as an efficient cryptographic primitive for designing fast block ciphers suitable for cheap hardware implementation in which simple key scheduling is used.

Three different implementations of the Cobra H-64 have been designed and implemented in this paper. All of them have been implemented with an internal pipeline architecture and using the various layout level techniques for decreasing the power dissipation and increasing the speed of operation of the ciphers. Simulation results using VHDL demonstrate that the internal pipeline architecture and proposed layout level techniques higher speed performance in comparison to the design published in [11]. While the throughput of the design published in [10] is around 5.5 Gbps, our internal pipeline architecture design provides a throughput of at least 6.7Gbps. Using Microwind 3 to simulate the layout level optimization in Figure 5.

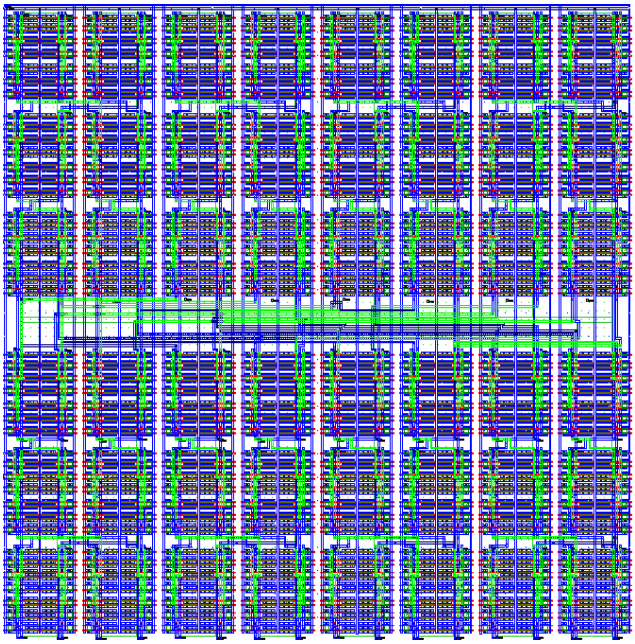


**Figure 5: Layout Implementation of Cobra H-64**

The design achieves a frequency more than 120MHz and throughput of at least 8.8 Gbps as shown in Figure 6. This of course will never been happened without the layout level where the number of transistors, the interconnections including the buses, the wires and the connectors are limited. It is important at layout level to reduce at minimal value these parameters so that the total power dissipation minimized at minimal value as well as the cost and the area are reduced also at the lowest measurements.

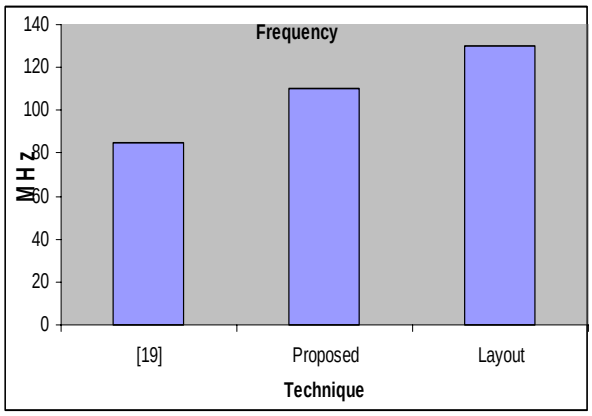



**Figure 6.a: Comparison Results Between Pipelining Archit. [19] and the proposed architecture in Verilog and Layout.**

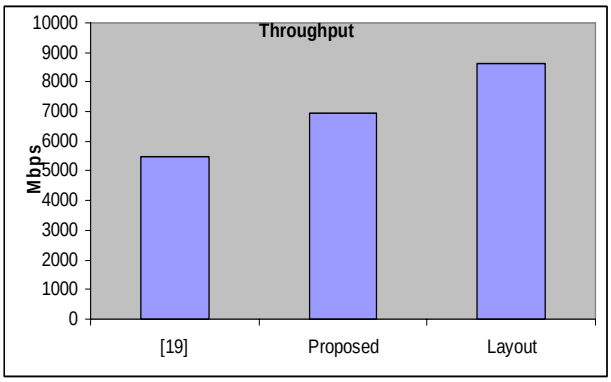



**Figure 6.b: Throughput between the proposed technique proposed in [19] and proposed architecture in Verilog and Layout**

The main benefit of pipelining is high-speed performance. The frequency of the design described in [10] was 90 MHz while for our proposed design using internal pipelining the frequency can be as high as 115 MHz. The proposed design including the layout level optimization achieves a frequency of at least 130 MHz. The improvement shown by our designs validates our hypothesis that layout level changes can offer better performance.

We first implemented the proposed designs in VHDL and compared them with designs that have been described in[10]. Simulation results show even that the ASIC technique is faster than the VHDL but the proposed idea shown an improvement in the speed operation of the encryption and decryption phases and decreasing radically the delay time. Table I illustrates a comparison between the conventional design of [10] and our proposed design simulated using Verilog, Xilinx.

**Table I: Implementation synthesis results**

| Architecture\ Hardware Device | FPGA Technology (Xilinx)[19] | | | | | Proposed Technique (Quartz) | | | | | Proposed Technique Layout Design (AMS 0.12, 1.2Volt) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Covered Area | | | Freq. (MHz) | Rate (Mbps) | Covered Area | | | Feq. (MHz) | Rate (Mbps) | Area | Freq. (MHz) | Rate (Mbps) |
| | CLBs | FGs | DFFs | | | CLBs | FGs | DFFs | | | | | |
| Full Rolling | 615 | 1229 | 204 | 82 | 525 | 680 | 1600 | 310 | 98 | 580 | | | |
| 10_Stage Pipe | 3020 | 6040 | 640 | 85 | 5500 | 3750 | 7000 | 735 | 110 | 6950 | 18550 | 130 | 8640 |

Because of the very simple key scheduling used in the proposed cipher it appears to be important to study how a single bit of key statistically influences ciphertext. For this purpose the secret key could be considered as input vector and fixing different plaintexts [12].

## 7. CONCLUSION

A proposed design of a Cobra-H64 cipher including layout level optimization techniques is described in this paper. The design includes inserting a pipeline architecture within the round blocks, which increases the speed of operation and throughput. In order to examine optimization of power dissipation, the design is simulated at the layout level using SPICE with parameters indicative of a low supply voltage and a low threshold voltage. Simulation results have demonstrated the validity of the layout level design.

## 8. REFERENCES


[1] J.-P. Kaps, "Cryptography for Ultra-Low Power Devices", PhD Dissertation, Worcester Polytechnic Institute, May 2006.

[2] J.-P. Kaps, K. Yüksel, B. Sunar, "Energy Scalable Universal Hashing", IEEE Transactions on Computers, volume 54, number 12, pages 1484-1495, December, 2005.

[3] K. Yüksel, J.-P. Kaps, and B. Sunar, "Universal Hash Functions for Emerging Ultra-Low-Power Networks", Proceeding of The Communications Networks and Distributed Systems Modeling and Simulation Conference (CNDS), San Diego, CA, January, 2004.

[4] Low-Power Processors And Systems on Chips, Christian Piguet, CRC Press, 2005.

[5] Low-Power Electronics Design, Laurie Kelly, Piguet Piguet, Christian Piguet, CRC Press, 2004.

[6] John Fry, and Martin Langhammer, "RSA & Public Key Cryptography in FPGAs," Altera Corporation – Europe, 2005.

[7] Synthesis of Arithmetic Circuits: FPGA, ASIC and Embedded Systems Jean-Pierre Deschamps, Gery J.A. Bioul, Gustavo D. Sutter,Wiley Press, 2006.

[8] Bruce Schneier, Applied Cryptography – Protocols, Algorithms and Source Code in C, Second Edition, 1996.

[9] Nicolas Sklavos, Xinmiao Zhang, Wireless Security and Cryptography, CRC Press, 2006.

[10] N. Sklavos, N.A. Moldovyan, O. Koufopavlou, High Speed Networking Security: Design and Implementation of Two New DDP-Based Ciphers, Mobile Networks and Application 10, 219-231, 2005.

[11] Nikolay A. Moldovyan1, Peter A. Moldovyanu1, and Douglas H. Summerville On Software Implementation of Fast DDP-based Ciphers, International Journal of Network Security, Vol.4, No.1, PP.81–89, Jan. 2007.

[12] Koufopavlou, "Mobile Communications World: Security Implementations Aspects - A State of the Art", CSJM Journal, Institute of Mathematics and Computer Science, Vol. 11, Number 2 (32), pp. 168-187, 2003.