# Implementation of Pipelined Data Encryption Standard (DES) Using Altera CPLD

Teo Pock Chueng
Faculty of Electrical Engineering
Universiti Teknologi Malaysia
81310 Skudai, Johor, Malaysia
email: pcteojb@tm.net.my

Dr. Zulkalnain Mohd Yusoff
Faculty of Electrical Engineering
Universiti Teknologi Malaysia
81310 Skudai, Johor, Malaysia
email:zul@suria.utm.my

Ahmad Zuri Sha'ameri
Faculty of Electrical Engineering
Universiti Teknologi Malaysia
81310 Skudai, Johor, Malaysia

## Abstract

The paper presents a Pipelined Data Encryption Standard (DES) architecture design implemented in Altera CPLD. The architecture contains of three main parts, DES module, pipeline module and control unit module. Four segments pipeline is used in this architecture to burst the throughput of DES. Although the processing time for a single encryption operation is still the same; but with more encryption operations, this pipelined DES can increase significantly the throughput. Altera Hardware Description Language (AHDL) is used to implement the pipelined DES design.

## 1. Introduction

Cryptography exists in the world long ago. People use encryption to protect information from undesired party. The most famous example of encryption in the history is Caesar encryption. Caesar used encryption to protect military information.

Internet is one of the most revolutionary communication inventions in this century. It provides highly flexible, mobile, wide-coverage communication. With the growth of Internet, more and more communication methods are becoming tightly coupled with Internet, such as, electronic mail, video conference, virtual community, news group, Internet television, Internet radio, Internet phone and so on.

One of the advantages of Internet is the open system architecture. Its flexibility makes Internet developed fast. On the other hand, the lack of privacy in the Internet becomes obstacle to growing further.

However, this weakness can be eliminated with the introduction of cryptography. Cryptography is used to transform intelligible information to unintelligible data. The encrypted data is sent via unsecured channel, such as the Internet. Although one can intercept the transmitting data but it is useless because of its unintelligible form. Only the desired receiver can retransform the data into intelligible information.

As a result, sensitive communication, electronic fund transfer, electronic commerce can be realized in the Internet.

In this paper, a pipelined Data Encryption Standard (DES) module is presented. The cryptographic theory behind the DES, pipeline architecture, implementation in Altera CPLD is discussed in the following section.

## 2. Data Encryption Standard

The Data Encryption Standard (DES), known as the Data Encryption Algorithm (DEA) by ANSI and the DEA-1 by the ISO, has been a worldwide standard for over 20 years. Although it was introduced as federal data encryption standard of United States of America on November 23 1976, it has held up remarkably well against years of cryptanalysis and is still secure against all but possibly the most powerful of adversaries.

DES is authorized for use on all unclassified government communication. Private sector, especially financial and banking sector, uses DES intensively to protect privacy of sensitive information. After years of research, DES is accepted worldwide as a strong encryption algorithm.

DES is a block cipher (Figure 1), which takes 64-bit input and 64-bit key. A 64-bit output is produced. The effective key length is 56 bits because 8 bits are used as parity-checking bits. There are a total of $2^{56}$ possible keys available in 56-bit key length.

DES is a symmetric algorithm. The same key is used for both encryption and decryption. DES uses two fundamental techniques in its algorithm, confusion and diffusion. The basic building block of DES is a single combination of these operations, substitution and permutation, on the plaintext based on the key.

DES has 16 rounds of this basic building block in a row (Figure 2). As a result, the ciphertext will be completely different with the corresponding plaintext and thus to achieve the objective of encryption

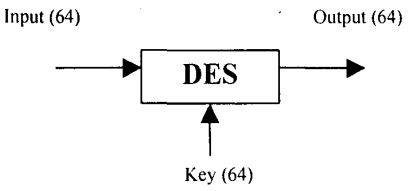


Figure 1. Block Diagram of DES

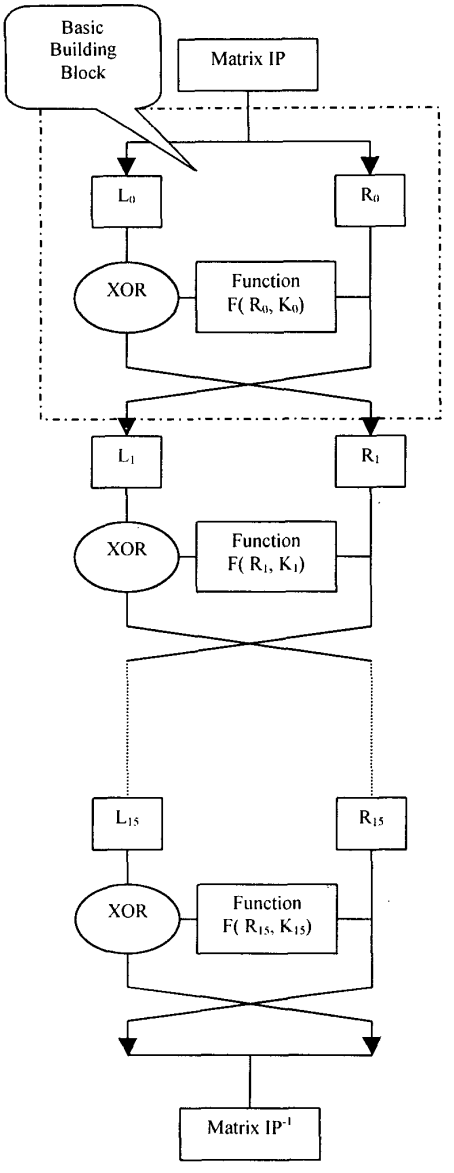


Figure 2: Algorithm of DES

Function F combines input and key to produce a key dependent output. Every round uses a subkey. There are a total of 16 subkeys derived from the 56-bit key. The procedure to derive subkey is not included in the scope of this paper. The focus of this paper is the pipeline architecture implementation of DES. For detailed algorithm description, refer to [1].

From Figure 2, it can be seen that the basic building block of DES is repeated for 16 times. The only difference between each round of basic building block is the subkey used.

In each basic building block of DES, the input will be split into two, left half and right half. The right half will become left half for the next round. Meanwhile, The right half will go through the function f to produce a key-dependent output and then xor with left half. The result will become right half for the next round.

After 16 rounds, the input will become totally different with the output. Output is the encrypted data, in this case of encryption.

Decryption will use the same algorithm. The only difference is the sequence of subkeys used in each round. In decryption, subkey $K_{15}$ will be used in round 1, subkey $K_{14}$ in round 2 and so on.

## 3. Practical Architecture

There are sixteen rounds of identical operation in a row to produce the corresponding output. This is an important characteristic to be exploited in pipelined DES design. In real world implementation, only one set of logic circuit of basic building block is used to build the entire DES (Figure 3). The same logic circuit will be repeated 16 times to accomplish the algorithm. This can significantly save the hardware resource required. An overhead control module is required to control the flow of data in this design.

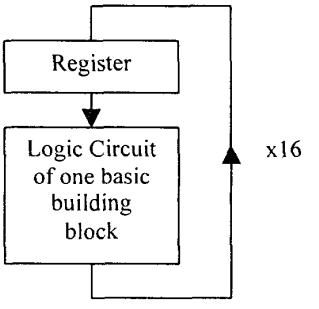


Figure 3. Real World Implementation of DES

As a result, the design can only take one input at a time. The new input can only come into the design after the output is produced after 16 rounds of operation. This sacrifices the throughput of DES. For example, if the processing time for one round is n. The total processing time for 1 encryption operation is 16n.

## 4. Fully Pipelined Architecture

To fully utilize DES for maximum throughput, all 16 rounds can be built separately. 16 rounds form a pipeline with 16 segments. A total of 16 different input can be fed into pipeline at the same time (Figure 4).

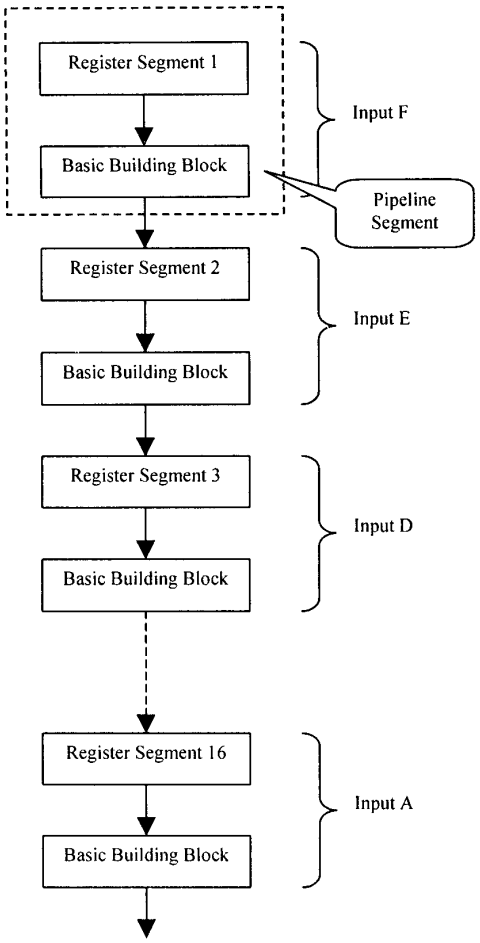


Figure 4. Fully Pipelined DES (16 segments)

Figure 4 shows a fully pipelined DES. It consists of 16 separate segments that can process different input at the same time. Each segment is formed by one register and one basic building block. The segment register stores the current input for the segment.

In the case of Figure 4, input A is first fed into segment 1. After being processed by basic building block of segment 1, the output is fed into register of segment 2. At the same time, input B is fed into segment 1. There are two input, input A in segment 2, input B in segment 1, in pipeline now.

The process will continue. Input C will be fed into segment 1 and so on. Each segment processes different input. The total processing time for 16 encryption operations is 16n+15n = 31n. Compared to practical implementation mentioned in the section above, 16n x 16 = 256n for 16 encryption operations.

The improvement of throughput is significant. However, this kind of 16-segment pipeline implementation consumes much more hardware resource.

Table 1. Comparison of Performance between Practical and Fully Pipelined DES

| Type | 16 Encryption | Average Time |
|---|---|---|
| Practical | 256n | 16n |
| Fully Pipeline | 31n | 1.938n |

## 5. Compromised Pipelined DES

After going through the advantages and disadvantages of practical DES and fully pipelined DES, one can realize that a compromised DES will probably fulfill most people's need.

Consequently, a compromised DES with 4 segments pipeline is built. The compromised pipelined DES (hereafter known as pipelined DES) consumes less hardware resource than fully pipelined DES does, and provides more throughput than practical DES does.

The following section will focus on the compromised pipelined DES.

### 5.1. Pipelined DES Architecture

The pipelined DES is designed to fit to Altera FLEX10K100A CPLD. The Altera CPLD will be interfaced to personal computer ISA bus. Due to the limitation of personal computer ISA bus architecture, 64-bit input and key have to be split into smaller length and be transmitted sequentially.

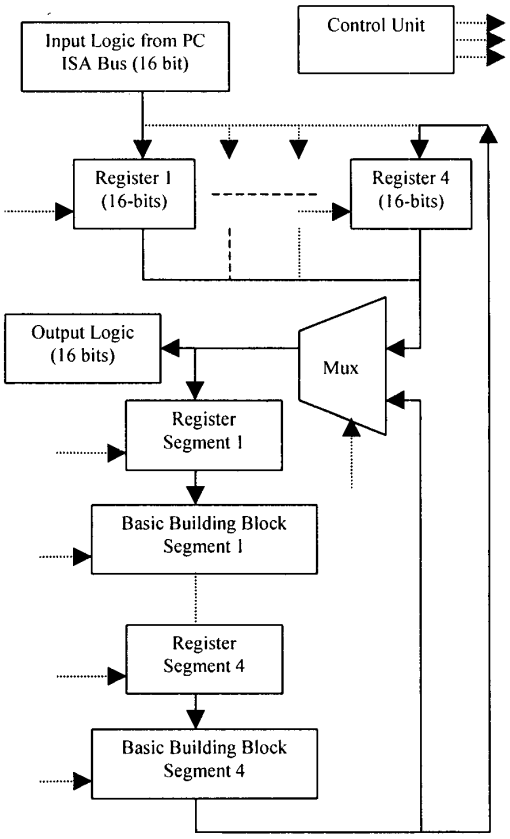


Figure 5. Pipelined DES Architecture

Figure 5 shows the pipelined DES architecture. There are four registers, each 16-bit, to store 64-bit input. It is to overcome the limitation of personal computer bus architecture. ISA bus architecture can only support up to 16-bit data bus.

Pipelined DES will load 4 different input from ISA bus for the first four rounds. This will fill the pipeline with 4 different input.

After the first four rounds, the result from the fourth segment will be fed back to the first segment and so on. 4 different input will be encrypted in the architecture simultaneously. The first encrypted output will be produced at $16^{th}$ round, second at $17^{th}$ rounds, third at $18^{th}$ round and fourth at $19^{th}$ round.

Output will be split up too before being sent via ISA bus to CPU. Control unit is in charge of coordinating operations of components and data flow. The flow chart is shown in Figure 6.

Table 2. Performance of Pipelined DES

| Implementation Type | 4 Encryption | Average Time |
|---|---|---|
| Practical | 64n | 16n |
| Pipeline | 19n | 4.75n |

Table 2 lists the processing time for practical and pipelined DES. Let's assume that n is the processing time for single basic building block. n is dependent on the technology used in the CPLD. Different target CPLD will produce different n.

There are 16 basic building block in one encryption operation. Thus, 16n time for a single encryption operation. For practical DES, 4 encryption operations take 16n x 4=64n. Average processing time per each encryption operation is 16n.

For pipelined DES, 4 encryption operations take 16n+3n=19n. Average processing time per each encryption operation is 4.75n.

### 5.2 AHDL

The hardware description language used to build the pipelined DES is AHDL. AHDL is a hardware description language from Altera. AHDL is a powerful tool to program Altera CPLD. Altera CPLD can be fully utilized to achieve maximum effectiveness with AHDL.

AHDL is an easy-to-learn HDL (hardware description language). The basic structure of AHDL is the same with VHDL, the worldwide standard of HDL. It is very easy to master AHDL if one has the background of VHDL.

From the experience and statistics, it shows that AHDL can produce better performance on Altera CPLD than VHDL does. The most possible reason behind this is probably that AHDL is specially designed to program Altera CPLD.

Consequently, pipelined DES design can gain better performance and faster processing speed with AHDL.

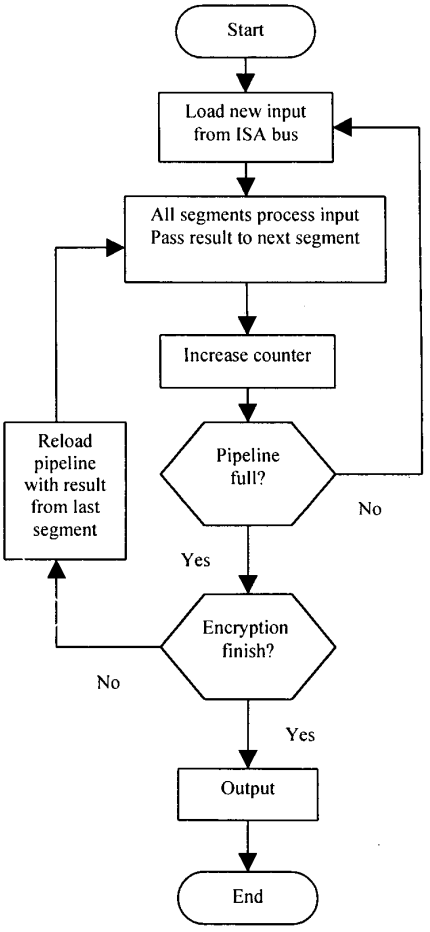


Figure 6. Flow Chart of Pipelined DES

### 5.3 Altera CPLD

The Altera CPLD model used in this paper is FLEX10K100A. Altera CPLD is the first embedded programmable logic device (PLD) family in the industry, providing System-on-a-Programmable-Chip integration. Based on reconfigurable CMOS SRAM elements, the Flexible Logic Element MatriX (FLEX) architecture incorporates all features necessary to implement common gate array megafunctions. The FLEX10K family provides the density, speed, and features to integrate entire systems, including multiple 32-bit buses, into a single device.

Table 3. FLEX10K100A Device Features

| Feature | FLEX10K100A |
|---|---|
| Typical gates | 100,000 |
| Maximum system gates | 158,000 |
| LEs | 4,992 |
| LABs | 624 |
| EABs | 12 |
| Total RAM bits | 24,576 |
| Maximum I/O pins | 406 |

## 6. Application

This pipelined DES design increases the throughput of DES significantly. It can be used in intensive cryptographic computation application. Applications such as, electronic commerce, Internet banking, electronic fund transfer, secure and private communication require better performance cryptographic system.

Pipelined DES can lighten the burden of heavy computational consumption. It is the solution to overcome the bottleneck caused by software limitation.

With dedicated pipelined DES hardware, special purpose system can be built without personal computer. This can reduce the cost of the system. On the other hand, the size of system can be further minimized too.

## 7. Discussion and Conclusion

The bottleneck is located at the interface between personal computer ISA bus and pipelined DES. 16-bit data bus really slows down the whole process of encryption. Most of the time is spent on sending data to and fetching data from pipelined DES.

A suggestion to curb this problem is to design a dedicated system without personal computer. A broader bus architecture can fully utilize the pipelined DES.

As a conclusion, pipelined DES is a greatly improved version of DES. It increases the throughput of the system to a higher level.

## 8. References

[1] Schneier, B. (1996). "Applied Cryptography, Protocols, Algorithms, and Source Code in C." 2nd edition. United States of America

[2] Eggebrecht, L. C. (1990). "Interfacing to the IBM Personal Computer." 2nd edition. United States of America.

[3] Gajski, D. D. (1997). "Principles of Digital Design." 1st edition. United States of America.

[4] Teo P. C. (2000). "DES Cryptographic System for Information Security"

[5] Nova Engineering. (1997). "Constellation FLEX10K Development System User's Manual."