Volume 4, Issue 4, April 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

FPGA Implementation of Blowfish Cryptographic Algorithm

Ankita Deshpande*
Department of Electrical & Electronics Engg .
PRMCEAM Badnera
Affiliated to SGBAU Amravati, India

P.S.Choudhary
Department of Electronics & telecomm.Engg,
PRMCEAM Badnera
Affiliated to SGBAU Amravati, India

Abstract— ***Information science involves not only the efforts made for gathering, acquiring or collecting the data that corresponds to information but also contains the ways to save it, protect it and preserve it. Encryption and decryption is a vital part in cryptography. The main aim is to protect and save the data as a small leakage may lead to the loss of information. For that the data is converted into gibberish form by using certain encryption algorithms. One of the encryption algorithms is the blowfish algorithm. The Blowfish algorithm in VHDL can provide a simple, robust implementation of Blowfish in hardware. A hardware implementation of Blowfish would be a powerful tool for any mobile device or any technology requiring strong encryption. The performance indices here are the security and speed of algorithm. The overall design is an incredibly fast, efficient blowfish implementation suitable for a plethora of applications. Encryption is used to disguise data making it to unintelligible unauthorized observers. Providing such security is especially important when data is being transmitted across open network such as internet. Blowfish encryption algorithm suitable for wireless network application security.***

Keywords- Blowfish, crypto, encryption, decryption, key expansion, gibberish

I. INTRODUCTION

Cryptography is referred to as the study of secret. Only the one which is intended to read the message can get it and is unreadable to others, thus converting readable message into an unreadable form. When the confidential information is sent over the communication channel, there is possibility of an unauthorized third party attack in order to learn the confidential information [1][7].

The blowfish cryptographic algorithm can be implemented by using VHDL. The Blowfish cryptosystem, designed by Bruce Schneier in 1993, is a very fast and useful scheme, even though it was introduced over a decade ago. If this technique is available in hardware, such a system may be the most powerful tool for any communication system where high security is needed. This cryptosystem is designed and is implemented using VHDL. It is used for high speed embedded applications such as mobile phone networks. Wireless communication schemes greatly need highly secured data encryption technique. In many of such applications it is difficult to use software cryptosystems. Thus the hardware implementations of such systems can be very useful for wireless applications [6].

Blowfish is a variable key size encryption algorithm which is based on block cipher technology. It is a symmetric kind of algorithm that uses same key for encryption as well as for decryption. One basic advantage of Blowfish is that it can use different key size up to the length of 448 bits [2]. The fig below shows the symmetric encryption/decryption process using blowfish algorithm [4].

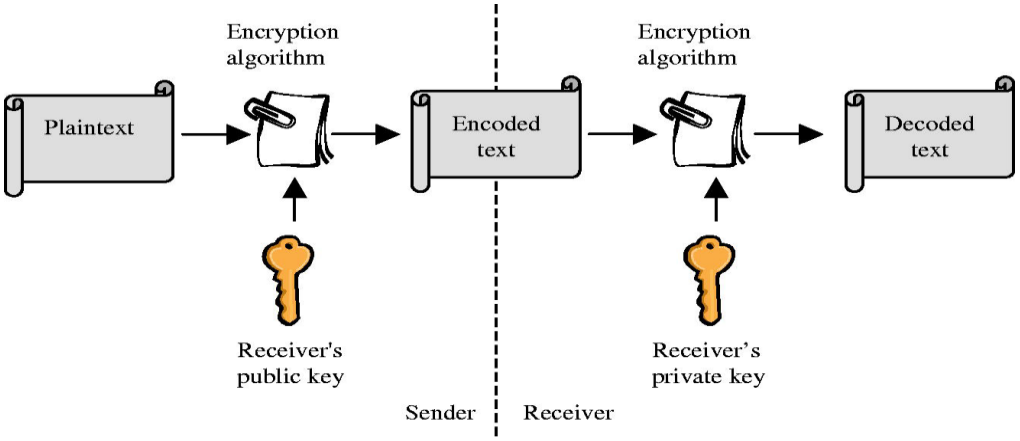

Fig. 1: A client/server implementation of an encryption system

Messages transmitted across the internet are susceptible to eavesdropping attacks via any path along the transmission of message. Here cryptography is required to protect information from being intercepted and stolen by unwanted third party. For information that needs to be secure for only minutes, hours, or perhaps weeks, a 64-bit symmetric key will suffice. For data that needs to be secure for years, or decades, a 128-bit key should be used. For data that needs to remain secure for the foreseeable future, one may want to go with as much as a longer key. The elementary operators of Blowfish algorithm include table lookup, addition and XOR. The encryption concept can be used in many applications like internet applications to encrypt the password, visa card number, banks, military communications, satellite channels and some other communication system.

A network is a series of individual elements transmitting and receiving various data. Whenever sensitive or confidential information is transmitted, there is a possibility of an unauthorized third party ”eavesdropping” on a transmission and learning contents of the sensitive message. This possibility is unacceptable in many scenarios. Cryptography is the process of translating a message into a form which is unreadable to everyone except the intended recipient. This is typically done with use of keys. A cryptographic key is roughly equivalent to the concept of a physical which can unlock the correct lock. In cryptography, keys are used to encrypt the message into a format which would appear as unreadable random information to an unauthorized third party.

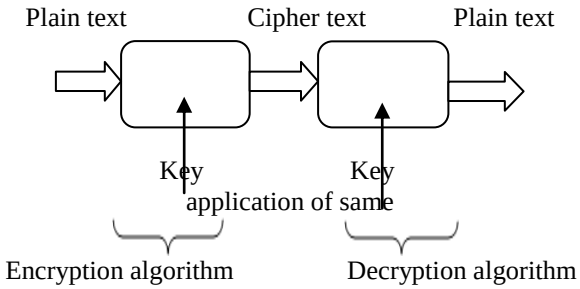

Fig. 2: Symmetric encryption/decryption process for blowfish algorithm

Blowfish was designed to meet following design goals:

Speed: It is meant to be significantly faster than DES on 32-bit microprocessors with relatively large caches. This type of architecture is readily available today to everyone.

Compactness: It is designed to run in a relatively small memory space, less than 5K.

Simplicity: Only simple operations are used, including addition, exclusive- or, and table lookups.

Flexibility of key size: The size of key can vary up to 448 bits (in 32 bit increments).

II. DESCRIPTION OF ALGORITHM

The Blowfish algorithm is conceptually simple, but is actual implementation and use is complex. Blowfish has fixed 64 bit block size. The key length of Blowfish is anywhere from 32 bits to 448 bits. The cipher is 16 round Feistel network which utilizes a structure that makes encryption and decryption very similar through use of following elements: P-boxes (permutation boxes - these perform bit shuffling), S-boxes(substitution boxes – similar to non-linear function) and Xoring to achieve Linear mixing[8].

Blowfish is a Feistel network block cipher with a 64 bit block size and a variable key size up to 448 bits long. The Blowfish algorithm is unencumbered by patents and is free to use for any one.

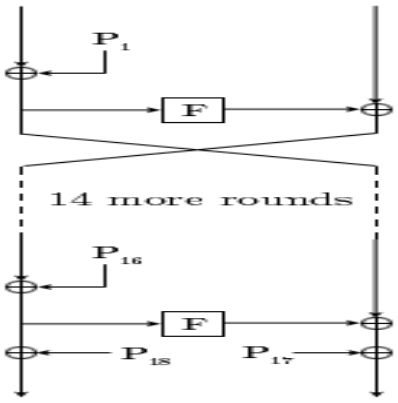

Fig. 3: Blowfish Algorithm

The F function is the feistel function of Blowfish, the contents of which are shown below.

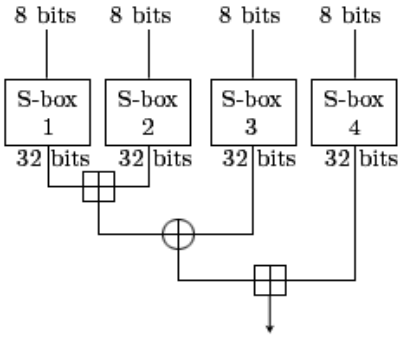

Fig. 4: The Feistel Function of Blowfish

Blowfish consists of three parts:
A. Encryption algorithm
B. Key-expansion
C. Decryption algorithm

A. Encryption algorithm:

During the key expansion stage, the input key is converted into several sub key arrays total 4168 bytes. There are the P-arrays, which has eighteen 32-bit boxes, and the S-boxes, which are four 32-bit arrays with 256 entries each. All of these boxes are initialized with a fixed string, the hexadecimal digits of pi [10].

After the string initialization, the first 32 bits of the key are XOR with P1 (the first 32-bit box in the P-array). The second 32 bits of the key are XOR with P2, and so on, until all 448, or fewer, key bits have been XOR. Cycle through the key bits is completed by returning to the beginning of the key, until the entire P-array has been XOR with the key[12].

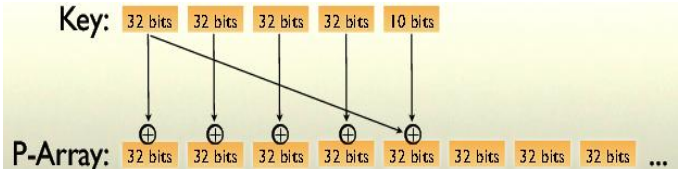

Fig. 5: XORing bits once the key has been traversed through once

Encrypt the all zero string using the Blowfish algorithm, using the modified P-array above, to get a 64 bit block. Replace P1 with the first 32 bits of output, and P2 with the second 32 bits of output (from the 64 bit block). Use the 64 bit output as input back into the Blowfish cipher, to get a new 64 bit block. Replace the next values in the P-array with the block. Repeat for all the values in the P-array and all the S boxes in order[10].

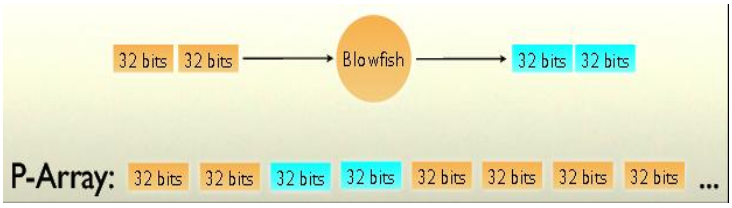

Fig. 6: The second 64 bit block is dropped into the P-array

The Blowfish algorithm is now ready for encryption. The encryption is a simply Feistal network of 16 rounds. For the input of 64 bits, do:

Divide x into two 32-bit halves: xL, xR
For i = 1 to 16:
xL = xL XOR P
xR = F(xL) XOR xR
Swap xL and xR
Next i
Swap xL and xR (Undo the last swap.)
xR = xR XOR P17

xL = xL XOR P18
Recombine xL and xR

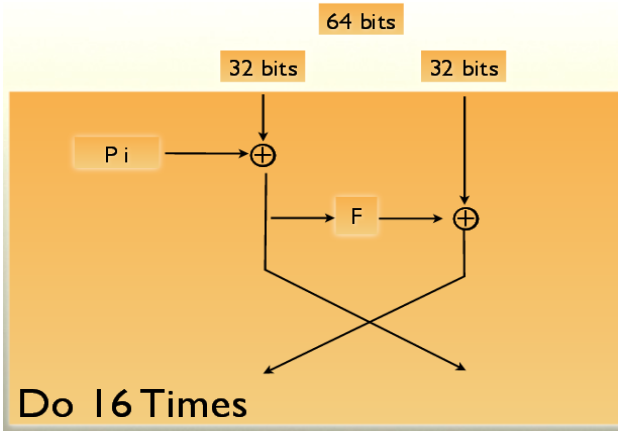

Fig. 7: The 16 rounds

The F function is: F(xL) = ((S1,a + S2,b mod 2^{32}) XOR S3,c) + S4,d mod 2^{32} where a,b,c,d are four 8 bit quartered derived from xL.

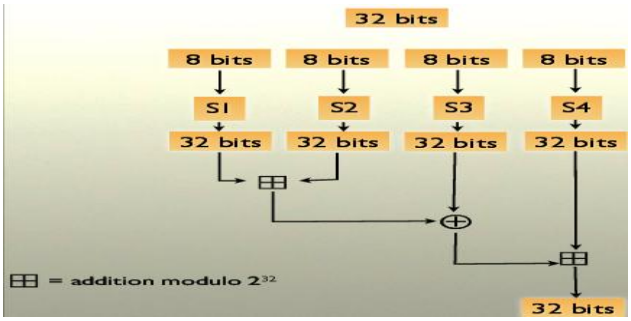

Fig. 8: The F function

B. Key-expansion:

In a simple cipher, one might exclusive-OR the key with the plaintext. Such a step is easily reversed by another exclusive-OR of the same key with the cipher text. In the case of the Blowfish, there are a number of rounds, needing the key, so the actual key size is 64 bytes[9][12].

C. Decryption algorithm:

Decryption is the same as encryption, except the P-arrays are used in reverse.

Hence, Blowfish encrypts by splitting half the block (32 bits) into 8-bit chunks (quarters) and inputting this into the S-box. The result from S-boxes then are added and XOR. Decryption is quite simple and accomplished by merely inverting the P17 and P18 cipher blocks and using P entries in reverse. The S-boxes and P-boxes are initialized with values from hex digits of pi. The variable length user-input key is then XOR with P-entries. Then a block of zeros is encrypted, and this result is used for P1 and P2 entries. The cipher text resulting from the encryption of a zero block is then encrypted again and use for P3 and P4. This process continues until every P-box entry and S-box entry has been replaced, resulting in 521 successive key generations. This involves about 4KB of data processing. This relatively complex key schedule makes Blowfish an effective and durable cryptographic algorithm[10].

Blowfish is among the fastest block ciphers available and yet remains cryptographically secure.

III. CRYPTANALYSIS OF BLOWFISH

It is not surprising that many interesting results were proposed; however, none came close to actually successfully cracking or providing a cryptanalysis of Blowfish[11].

Only five results in total were submitted. John Kesley could only break 3-round Blowfish. The most promising attack was proposed in 1996 by Vincent Rijmen in his doctoral disserertation, but this attack can only break 4 rounds of Blowfish and no more[11]. Given that these attempts are only ones known thus far and they are surprisingly weak in decrypting the actual Blowfish algorithm, future of Blowfish as a secure algorithm is very promising indeed[11]

IV. SIMULATION RESULT

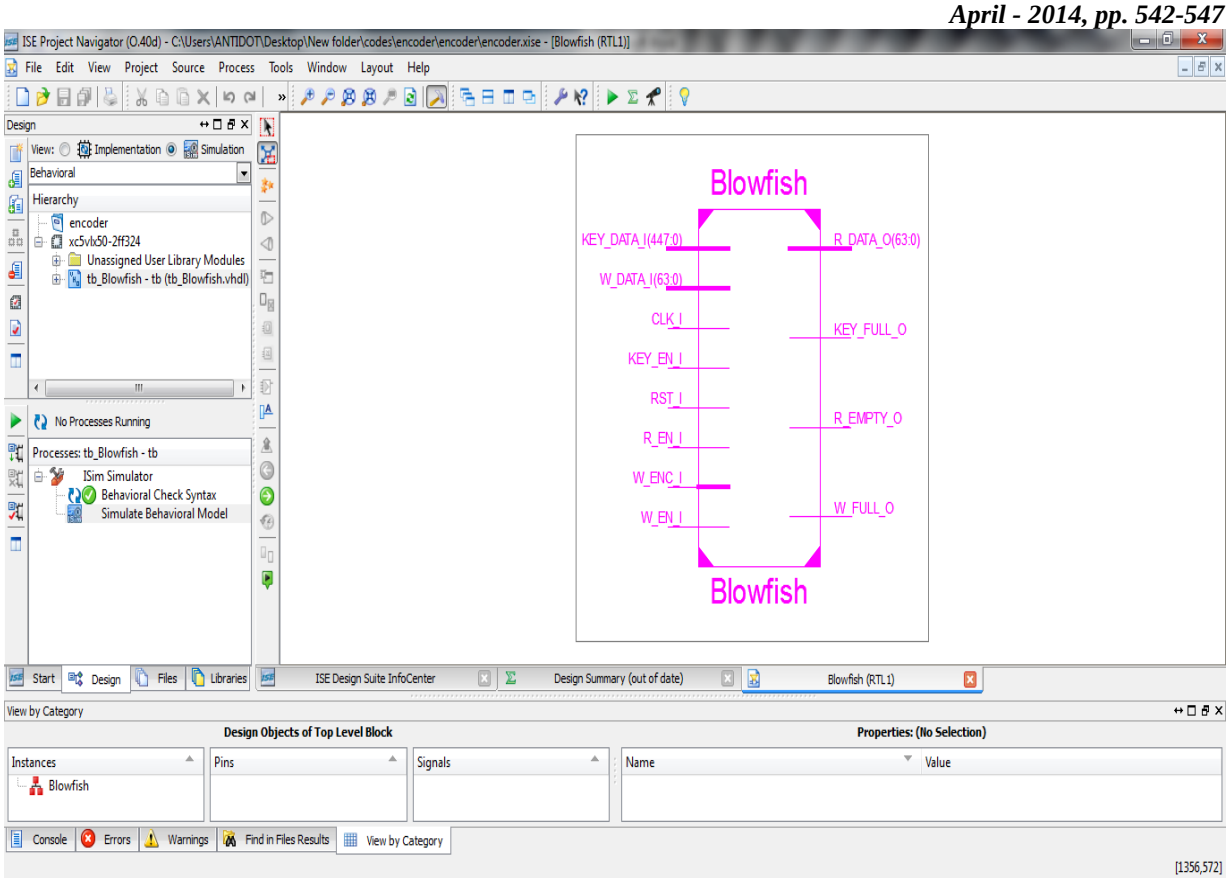

Fig. 9: RTL Schematic of Blowfish Algorithm

A. Encryption Algorithm Process

1. **Plaintext: abcdef7645231978**
2. **Key data: acdbafe321568783**
3. **Encrypted data: 39fe773110d00863**

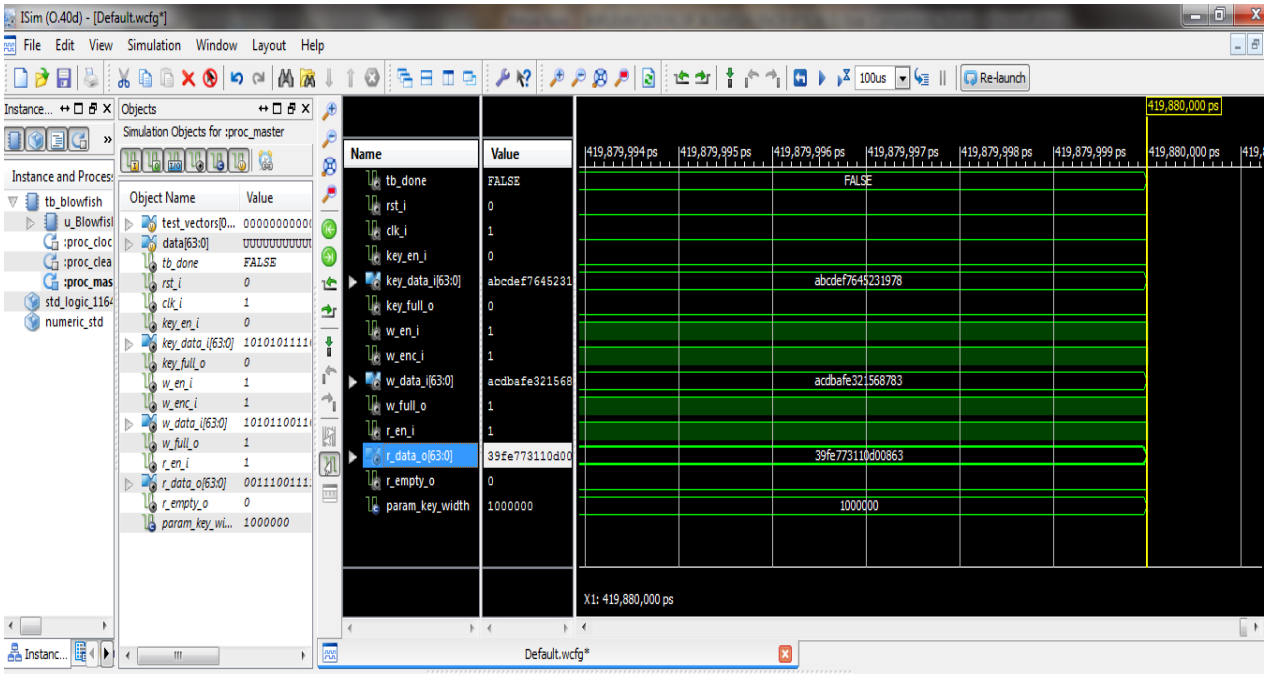

Fig. 10: Encrypted result of Blowfish Algorithm

V. CONCLUSION

Blowfish is not only secure, but also fast, and suitable for different platforms, therefore, it has high value of application in the field of information security. One should find it important that the maximum key size was used, and the key was chosen at random from the full key space of size 2^{448}, since maximum key length is 448 bits.

The various encryption algorithms to accommodate the wireless network applications can be optimized further in future. Furthermore, a stronger encryption algorithm with high speed and minimum energy consumption can be developed to achieve better security and the performance evaluation parameters can be optimised.

REFERENCES

[1] Akshatha.B.R, Amitabha.K.Kumar, Neha Choubey, Jamuna.S, Raja Jitendra Nayaka, *FPGA Implementation of Modified Blowfish Algorithm* International Conference on Electronics and Communication Engineering, 28th April-2013, Bengaluru. www.ijert.org

[2] B. Schneier, *Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish),*Fast Software Encryption, Cambridge Security Workshop Proceedings (December 1993), Springer-Verlag, 1994.

[3] D. S. Abdul. Elminaam, H. M. Abdul Kader and M. M. Hadhoud, *Performance Evaluation of Symmetric Encryption Algorithms*, Communications of the IBIMA Volume 8, 2009.

[4] Deepak Kumar Dakate, Pawan Dubey, *Blowfish Encryption: A Comparative Analysis using VHDL*, International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-1, Issue-5, June 2012.

[5] Gurjeevan Singh, Ashwani Kumar, K. S. Sandha, *A Study of New Trends in Blowfish Algorithm*, International Journal of Engineering Research and Applications (IJERA) Vol. 1, Issue 2, pp.321-326. www.ijera.com.

[6] L.Kranthi Kiran J. E. N. Abhilash P. Suresh Kumar, FPGA *Implementation of Blowfish Cryptosystem Using VHDL*, International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 1, January- 2013.

[7] William Stallings ,*Cryptography and Network Security*,5th edition, Pearson Publications.

[8] B. Schneier, *High Speed SOC Design for Blowfish Cryptographic Algorithm*, 2007 IEEE.

[9] B. Schneier, *Description of a New Variable Length Key, 64-bit Block Cipher (blowfish)*, Proceedings of Fast Software Encryption, pp. 191-204.

[10] Russell K. Meyers and Ahmed H. Desoky, *An Implementation of the Blowfish Cryptosystem*, 2008 IEEE.

[11] Wikipedia," *Advanced Encryption Standard Process*" http://en.wikipedia.org/wiki/advanced_Encryption_standard_process.

[12] D. Schmidt, *On the key schedule of Blowfish Cryptology*, ePrint Archive,2005.

