

AN FPGA NETWORK ARCHITECTURE FOR ACCELERATING 3DES – CBC

Chin Mun Wee, Peter R. Sutton and Neil W. Bergmann

School of Information Technology and Electrical Engineering,
The University of Queensland, Brisbane, QLD, 4072, Australia

email: cmwee@itee.uq.edu.au, p.sutton@itee.uq.edu.au, bergmann@itee.uq.edu.au

ABSTRACT

This paper presents a DES/3DES core that will support Cipher Block Chaining (CBC) and also has a built in keygen that together take up about 10% of the resources in a Xilinx Virtex II 1000-4. The core will achieve up to 200Mbit/s of encryption or decryption. Also presented is a network architecture that will allow these CBC capable 3DES cores to perform their processing in parallel.

1. INTRODUCTION

Field Programmable Gate Arrays (FPGAs) are becoming more popular in the field of embedded computing. Due to the increase in the number of gates, speed and the reduction of prices, they have become a viable alternative to ASICs in certain applications. Virtual Private Networks (VPNs) incur a high processing overhead, especially when implemented in software. This is where hardware acceleration can significantly improve the performance. Although Advanced Encryption Standard (AES) has already been introduced, many VPNs still use Data Encryption Standard (DES) or 3DES as their default encryption algorithm. Much of the work that has been done on encryption with FPGAs has been focussed on unrolling the DES loop and pipelining the process in the Electronic Code Book (ECB) mode. This has produced results that support data rates on the order of 10Gigabit/s but it can take up most of the resources of a FPGA. In reality, much of the DES and 3DES ciphers that are used in the industry are performed in the Cipher Block Chaining (CBC) mode that pipelining with this method cannot accelerate effectively.

This paper presents a DES/3DES core with CBC and a network architecture that can accelerate them in parallel. An outline of the remainder of the paper is as follows. The next section will describe VPNs and the following section, the DES and 3DES algorithm. Section 4 will describe the implementation of DES and 3DES on an FPGA along with the numerical results. The method of accelerating feedback cores will be discussed in section 5 and conclusions will be presented in section 6.

2. VIRTUAL PRIVATE NETWORKS

A VPN is a private network that makes use of the public telecommunication infrastructure, maintaining privacy through the use of a tunnelling protocol and security procedures [1]. The main purpose of a VPN is to give the company the same capabilities as private leased lines at a much lower cost by using the shared public infrastructure. Companies today are looking at using a private virtual network for both extranets and wide-area intranets.

As the Internet became more popular as a corporate communications medium, security became much more of a pressing issue for both customers and providers. Secure VPNs allow traffic to be encrypted at the edge of one network or at the originating computer, moved over the Internet like any other data, and then decrypted when it reached the corporate network or a receiving computer. The main reason that companies use it is so that they can transmit sensitive information over the Internet without needing to worry about who might see it. Everything that goes over a secure VPN is encrypted to such a level that even if someone captured a copy of the traffic, they could not decrypt the traffic in reasonable time even if they used a great amount of computing effort. Furthermore, using a secure VPN allows the company to know that an attacker cannot alter the contents of their transmissions, such as by changing the value of financial transactions. Secure VPNs provide authentication, confidentiality and also integrity of the data.

2.1. IPsec

The various VPN protocols are defined by a large number of standards and recommendations that are listed by the Internet Engineering Task Force (IETF). IPsec is by far the most dominant protocol for secure VPNs [2]. IPsec provides security services at the IP layer by enabling a system to select required security protocols, determine the algorithm(s) to use for the service(s), and put in place any cryptographic keys required to provide the requested services. IPsec can be used to protect one or more "paths" between a pair of hosts, between a pair of security gateways, or between a security gateway and a host.

The IPsec series of protocols makes use of various cryptographic algorithms in order to provide security services. To ensure interoperability between different implementations it is necessary to specify a set of mandatory-to-implement algorithms to ensure at least one algorithm that all implementations will have available. At the writing of this paper, the latest recommendations are summarized in Table 1 [3].

Table 1. IPsec Encapsulating Security Payload protocol

Requirement	Encryption Algorithm (notes)
MUST	NULL
MUST-	TripleDES-CBC
SHOULD+	AES-CBC with 128-bit keys
SHOULD	AES-CTR
SHOULD NOT	DES-CBC

From this informative draft for the IETF, it can be seen that TripleDES-CBC is still the lowest common denominator for IPsec VPN functionality.

3. DES AND 3DES ALGORITHM

The DES (Data Encryption Standard) algorithm is the most widely used encryption algorithm in the world. DES will live on in government and banking for years to come through a life extending version called 3DES. DES is a block cipher, it operates on plaintext blocks of a given size (64-bits) and returns ciphertext blocks of the same size.

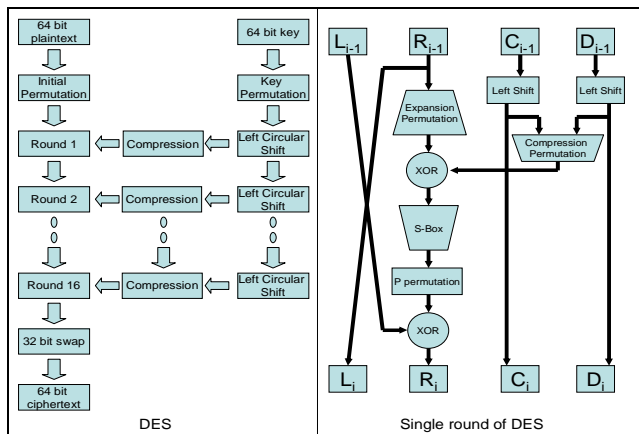


Fig. 1. Operation of the DES algorithm

The operation of DES can be seen in Fig.1. First, the plaintext is passed through an initial permutation that produces a permuted input. The permuted input is then split into a left half (L_i) and a right half (R_i), each has 32 bits. The right half of each round is passed through an expansion permutation and then exclusive-ored with the subkey for that particular round. That result is then passed through the s-box and forms a 32 bit result that is again passed through another permutation before another exclusive-or with the left portion. This is repeated for 16 times for a single 64 bit block of data. [4]

DES takes a 64 bit value as a key although the actual key length is only 56 bits since every 8th key bit is not used. This is done by mapping as detailed in Permuted Choice One. The resulting 56 bit key is separated to two 28 bit quantities named C_0 and D_0 . At each round of the subkey generation, each C and D values are subjected to either a 1 bit or 2 bit circular shifts that is pre-governed. They are then passed through another permutation called Permuted Choice Two before being applied to the respective rounds of DES.

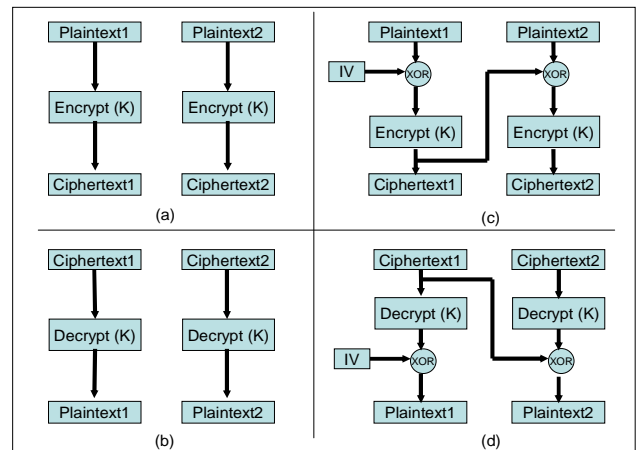


Fig. 2. Different Modes of Operation for DES

The different modes of operation for DES are shown in Fig. 2. Fig.2 (a) and (b) details the encryption and decryption process for DES-ECB and Fig. 2 (c) and (d) the same process for DES-CBC. In the Electronic Code Book (ECB) mode of DES, the ciphertext output will be the same if the plaintext and key are the same. This is actually a compromise on the security of the cipher [5]. Therefore a more secure mode is usually used in practice. It is Cipher Block Chaining (CBC) version where the ciphertext is exclusive-ored in to the next set of plaintext to be encrypted.

The 3DES-CBC or DES-EDE3-CBC algorithm is a simple variant of the DES-CBC Algorithm. The "outer" chaining technique is described. In 3DES-CBC, an Initialization Vector (IV) is exclusive-ored with the first 64-bit (8 byte) plaintext block. The keyed DES function is iterated three times, an encryption followed by a decryption followed by an encryption, and generates the ciphertext for the block. Each iteration uses an independent key: k_1 , k_2 and k_3 . For successive blocks, the previous ciphertext block is exclusive-ored with the current plaintext. The keyed DES-EDE3 encryption function generates the ciphertext for that block. To decrypt, the order of the functions is reversed: decrypt with k_3 , encrypt with k_1 , and exclusive-or with the previous ciphertext block. Note that when all three keys (k_1 , k_2 and k_3) are the same, 3DES-CBC is equivalent to DES-CBC. This property

5. NETWORK OF CRYPTOGRAPHIC CORES

As has been stated earlier, pipelining is not a suitable method to speed up CBC type encryption and decryption. However, the recommended mode of operation for real world industrial application uses the CBC mode of operation. In fact, any form of implementation which involves feedback will be difficult to implement via pipelining. The proposed design involves the use of Block RAM (BRAM) which is available on the Xilinx FPGA platform. On the Xilinx Virtex II and Spartan III platform and above, BRAMs are based on the RAMB16_Sm_Sn components. They are dual-ported dedicated random access memory blocks with synchronous write capability. Each port is fully synchronous with independent clock pins. Any inverter placed on a RAMB16 port is absorbed into the block and does not use a CLB resource [10].

5.1. BRAM Network Architecture

On a high usage network switch/router, there are potentially multiple VPN streams at any one time. If only one highly pipelined 3DES system is used, there will be wastage of resources since the input of the next 64 bit plaintext depends on the previous ciphertext in the same stream. Therefore the next 64 bit block to be processed cannot be put into the pipeline until the previous 64 bit of data has been fully encrypted/decrypted. One method to solve this is to use various non pipelined 3DES cores to handle different encryption/decryption streams.

The BRAM can be driven by different clocks on different ports. The advantage of this architecture is the insulation of the controller system and the actual 3DES core. Another core that contains other cipher or hash functions can easily be added with minimal effort. This

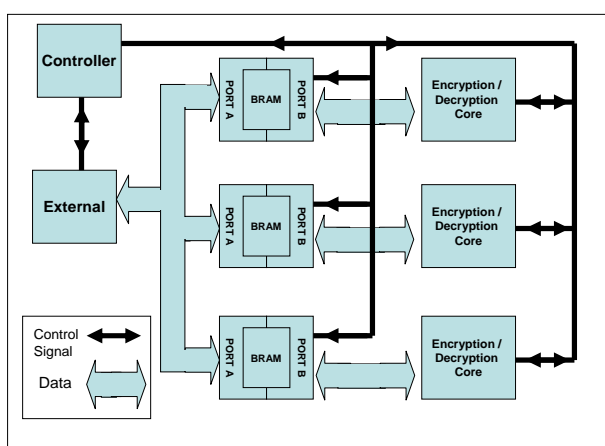


Fig. 4. Multiple BRAM and Cipher cores

further benefit from the reconfigurable nature of FPGA that additional BRAM buffer and cores can be added when there

are demands for it. This concept can easily be extended to the case of multiple BRAM to be used as a buffer for multiple tunnels using different keys as shown in Fig. 4.

6. CONCLUSION

This paper has presented a DES and 3DES core that supports CBC and also a network architecture that can accelerate these cores effectively using an FPGA. From the resources that are used in the implementation of one 3DES-CBC core and one BRAM, an XC2V1000FG456 would be able to accommodate up to 10 sets of these in theory but in application perhaps only 7 sets can be added. This would give a theoretical bitrate of more than 1 Gigabit per second if multiple streams/tunnels are being processed. In fact, any other processing core can be added for parallel acceleration, even those with feedback like DES-CBC or AES-CBC.

7. REFERENCES

- [1] Virtual Private Network Consortium, <http://www.vpnc.org>.
- [2] S. Kent, R. Atkinson, "Security Architecture for the Internet Protocol", RFC2401, November 1998
- [3] Donald E. Eastlake 3rd, "Cryptographic Algorithm Implementation Requirements For ESP And AH" Internet Draft.
- [4] Data Encryption Standard, Federal Information Processing Standard (FIPS) Publication 46, National Bureau of Standards, U.S. Department of Commerce, Washington D.C. (January 1977).
- [5] B.Schneier, "Applied Cryptography", John Wiley & Sons, 2nd Edition, 1996.
- [6] R. Pereira, R. Adams, "The ESP CBC-Mode Cipher Algorithms" RFC 2451, November 1998]
- [7] C. Patterson, "High Performance DES Encryption in Virtex FPGAs Using JBits", IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM 2000), pp. 113-121, April 2000
- [8] T. Wollinger, J. Guajardo, C. Paar, "Security on FPGAs : State-of-the-Art Implementations and Attacks", ACM Transaction on Embedded Computing systems, Vol.3, No.3, August 2004, pp. 534-574.
- [9] P. Hamalainen, M. Hannikainen, T. Hamalainen, J. Saarinen, "Configurable hardware implementation of triple-DES encryption algorithm for wireless local area network", Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing, 7-11 May 2001, Volume: 2 pp. 1221 - 1224
- [10] Xilinx, "Libraries Guide : Design Elements : RAMB16_Sm_Sn", 2003, pp. 165