Available online at www.sciencedirect.com

SCIENCE DIRECT®

Computers and Electrical Engineering 31 (2005) 380–390

Computers and Electrical Engineering

www.elsevier.com/locate/compeleceng

A fault-tolerant pipelined architecture for symmetric block ciphers

Min-Kyu Joo, Yoon-Hwa Choi [*]

Department of Computer Engineering, Hongik University, Seoul, Republic of Korea

Received 14 September 2004; accepted 8 July 2005
Available online 4 November 2005

Abstract

Secure transmission over wired/wireless networks requires encryption of data and control information. For high-speed data transmission, it would be desirable to implement the encryption algorithms in hardware. Faults in the hardware, however, may cause interruption of service. This paper presents a simple technique for achieving fault tolerance in pipelined implementation of symmetric block ciphers. It detects errors, locates the corresponding faults, and readily reconfigures during normal operation to isolate the identified faulty modules. Bypass links with some extra pipeline stages are used to achieve fault tolerance. The hardware overhead can be controlled by properly choosing the number of extra stages. Moreover, fault tolerance is achieved with negligible time overhead.

Keywords: Block ciphers; Fault tolerance; Concurrent error detection; Recovery

1. Introduction

Data encryption is used to provide security in various wired/wireless network applications. Implementing encryption algorithms in hardware is desirable for various applications to meet the speed requirements under the power constraints. Secret-key block ciphers, such as DES,

[*] Corresponding author.
E-mail address: yhchoi@cs.hongik.ac.kr (Y.-H. Choi).

0045-7906/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compeleceng.2005.07.003

IDEA, and Rijndael [6,7], have similar internal structure, although their encryption functions may differ. Encryption is done with rounds of function, realizable with repeated use of a single functional module. Due to this feature the encryption algorithms can easily be implemented in hardware. Faults in the systems, however, may cause interruption of service.

Several techniques have been proposed in [1–4] to detect errors in block ciphers. By suppressing the erroneous ciphertext, fault-based side-channel cryptal analysis can be tolerated [4]. In Vinci [3], encryption/decryption hardware was functionally duplicated and the resulting outputs were compared to detect errors. Although this technique does not cause any notable time delay, it requires a significant hardware overhead. Coding techniques have been applied to detect errors in block ciphers [2]. The overhead required for encoding/decoding is relatively small compared to the hardware redundancy technique [3]. The error detection coverage, however, depends on the particular coding scheme employed.

Wolter et al. [1] have proposed two concurrent error detection techniques based on information redundancy and redundant test words. The first one uses extra hardware for a low-cost residue code. The residue generated from the original block has been compared with that from the added residue building block. Error detection, if there is a detectable error, can readily be done. Its coverage, however, might be limited depending on the chosen coding scheme. The second one is considered to be a time-redundancy technique using redundant test words. Testing is performed by stealing some of the cycles and the encrypted test words are decrypted for comparison. This technique, however, does not check errors occurred in the encryption process of the actual message. In other words, there is no guarantee of confining errors within the cipher block. Hence no immediate action can be taken prior to transmitting the erroneous ciphertext generated. Error detection latency could also be problematic.

More recently, a new concurrent error detection technique for symmetric encryption algorithms, exploiting the inverse relationship between encryption and decryption, has been proposed [4,5]. It has been applied to three different levels of encryption: algorithm-level, round-level, and operation-level. The technique, however, is insufficient to tolerate faults since no recovery measure is provided.

In this paper, we present a fault-tolerant architecture for symmetric block ciphers. It is based on a hardware pipeline for encryption and decryption. Some extra stages with bypass links [8] are employed to achieve fault tolerance. Errors generated can readily be detected and the corresponding faulty stage is removed from the rest of the system by reconfiguration with the bypass links.

The rest of the paper is organized as follows. In Section 2, faults in symmetric block ciphers will be briefly addressed. Section 3 discusses fault model and presents our fault-tolerant architecture for block ciphers. A unidirectional pipeline for achieving fault tolerance is introduced in Section 4. Performance of the proposed architecture will be evaluated in Section 5. Conclusions are made in Section 6.

2. Faults in block cipher architecture

In a block cipher, the message is split into n-bit blocks, each of which is applied to the cipher unit along with an encryption key to generate the output (encrypted) block. Encryption algorithms widely used for block ciphers, such as DES, IDEA, RC5, Twofish, Rijndael, etc. have