Utilisation de ressources cryptographiques pour le test des circuits sécurisés

Marie-Lise Flottes, Marion Doulcier, Bruno Rouzeyre

Utilisation de ressources cryptographiques pour le test des circuits sécurisés

M. Doulcier, M. L. Flottes, B. Rouzeyre

Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier
Univ. Montpellier II / CNRS
161 rue Ada, 34932 Montpellier, France
{doulcier, flottes, rouzeyre}@lirmm.fr

Résumé— *La réutilisation de ressources cryptographiques, présentes dans un circuit, pour implanter des mécanismes de test intégré, permet de réduire la surface additionnelle due à celui-ci. Dans cet article nous présentons comment le cœur cryptographique AES peut être utilisé pour le test intégré des circuits sécurisés. D'une part, en tant que générateur de vecteurs de test, l'AES produit des séquences de qualité au moins égales à celles produites par un LFSR. D'autre part, utilisé comme compacteur de réponses, la probabilité de masquage d'erreurs de l'AES est de l'ordre de* 2^{-128} *donc négligeable. Ces deux caractéristiques font de l'AES une alternative intéressante aux solutions standards.*

I. INTRODUCTION

Les circuits sécurisés font maintenant partie de notre quotidien : téléphonie, télévision, sécurisation des accès, cartes de paiement, carte vitale, passeport électronique... Ces circuits doivent permettre de préserver la confidentialité des données mais également de garantir leur intégrité et leur authenticité ainsi que la non-répudiation de l'information. Tout ceci est assuré par l'implantation de mécanismes cryptographiques. Ces mécanismes permettent de coder et décoder l'information à l'aide d'un ensemble de données appelé clef secrète.

Le test de ces circuits exige une attention toute particulière car la moindre faute non détectée et l'ajout de ressources de test peuvent induire une faille sécuritaire. Par exemple, si la technique de test dite « scan path » facilite grandement l'application de séquences de test, elle fragilise à contrario la sécurité de la puce. En effet, cette technique consiste à rendre le contenu d'un circuit observable et contrôlable, tandis que la sécurité repose sur la non-observation et le non-contrôle du circuit. Dans [1] et [2], des attaques basées sur l'emploi d'une technique de scan path pour retrouver la clef secrète sont décrites.

Les techniques de test intégré BIST (« Built-In Self Test ») semblent plus appropriées à ce type de circuits car elles ne donnent pas d'accès extérieur. La technique de BIST la plus commune consiste à appliquer une séquence pseudo-aléatoire générée par un LFSR (Linear Feedback Shift Register) au circuit sous test (CUT) puis de comparer la signature obtenue en sortie à l'aide d'un MISR (Multiple Input Signature Register) avec celle attendue. Les techniques de BIST sont donc plus adaptées à ce type de circuits, mais sont coûteuses en terme de surface additionnelle car elles nécessitent l'implantation d'un générateur de vecteurs de test, d'un analyseur de signature et de logique de contrôle.

Ce papier présente une méthode permettant de substituer l'implantation de ressources de test intégré usuelles par le cœur cryptographique déjà présent dans le circuit sécurisé. Le cœur cryptographique est utilisé comme générateur de vecteur de test (TPG) et analyseur de signature (SA).

Le crypto algorithme, son implantation comme ressource de test sont présentés en section II. Ces propriétés comme générateur de séquences de test sont évaluées en section III. La section IV discute la probabilité de masquage de la structure proposée.

II. L'AES ET SON IMPLANTATION

L'algorithme cryptographique étudié est l'AES (Advanced Encryption Standard) [3] de 128 bits de clef. Cet algorithme permet de chiffrer un texte en clair constitué de 128 bits de données à l'aide d'une clef secrète. L'algorithme est constitué de plusieurs étapes appelées ronde composées de 4 opérations, ici 10 rondes car 128 bits de clef.

Dans [4], il a été montré que l'AES pouvait être utilisé pour s'autotester. Les résultats montrent qu'un taux de couverture de 100% est atteint en 12 cycles d'encryptions.

Le test du cœur AES pouvant s'effectuer sans ajout de LFSR ou MISR, nous avons voulu évaluer ces capacités en tant que ressource de test pour tester les autres cœurs de la puce.

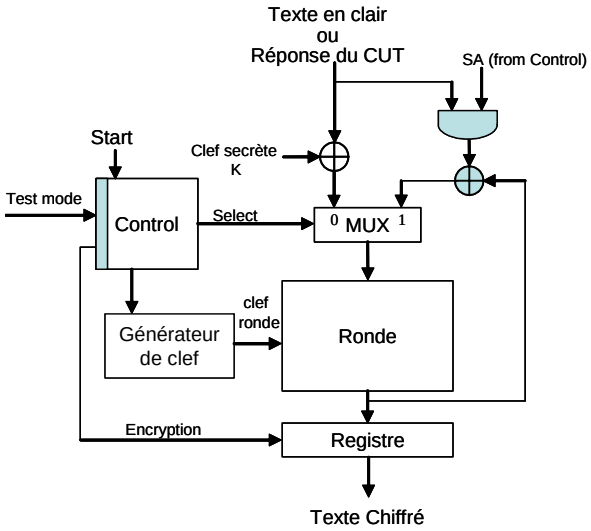

Figure 2 : AES TPG/SA implantation

La figure 2 présente l'architecture de l'AES et les modifications apportées (en grisé) pour l'utiliser en tant que TPG et SA. Ces deux nouveaux modes de

fonctionnement nécessitent l'addition de nouveaux signaux de contrôle et d'opérations xor et and. Le signal select est à 0 pendant la première étape en mode normal, TPG ou d'autotest. En mode TPG, ce signal permet de charger la graine du générateur. Après, le signal select est mis à 1 tandis que SA est mis à 0. Les vecteurs de test sont issus du registre à chaque cycle d'horloge. En mode SA, les signaux select et SA sont à 1. Une opération de ou-exclusif est effectuée entre une réponse du CUT et le résultat précédent de ronde. La signature finale est obtenue après que toutes les réponses ont été compressées.

III. GENERATEUR DE PATTERN DE TEST : TPG

L'approche BIST est souvent liée à l'utilisation de générateur de nombre pseudo-aléatoire pour produire les vecteurs de test. La structure proposée en mode TPG génère un vecteur à chaque cycle d'horloge. La séquence de vecteurs obtenue a été caractérisée à l'aide de la suite de tests statistiques définis dans le NIST [5] afin de déterminer ses propriétés aléatoires et a été comparée à une séquence générée par un LFSR de 128 bits.

Cependant une solution utilisant le processus complet de chiffrage AES (10 rondes), moins coûteuse en logique de contrôle, a été présentée dans [6] comme un bon générateur de mots aléatoire. Toutefois, cette dernière solution ne permet de générer qu'un vecteur toutes les 10 rondes, soit un vecteur tout les 10 cycles d'horloge.

Nous avons étudié individuellement les flots de bits sortant par chacune des pads du registre. Les flots ont été choisis de sorte à contenir 1,5 millions de bits. A chacun de ces flots nous avons appliqué l'ensemble des tests du NIST. Les tests du NIST donnent une réponse vrai/faux (aléatoire/non aléatoire).

Le tableau 1, ci-dessous, décrit la proportion (P) de flots de bits validant un test sur le nombre total de flots de bits (128), ceci pour chacun des 15 tests

	LFSR	AES
	P en %	P en %
Freq	21,09	99,22
BlkFreq	100	97,66
CuSum	21,09	100
Runs	100	80,47
LongRuns	100	99,22
Rank	96,88	99,22
DFFT	100	99,22
Univ	97,66	99,22
Apen	100	94,53
Serial1	100	97,66
Serial2	100	99,22
LinComp	0	99,22
non over	99,18	98,64
overl	100	96,88
Random	95,16	99,17

Tableau 1: proportion de flots de bits validant un test

Au regard de ce tableau, on constate que les flots de bits générés par l'AES ont des propriétés d'aléatoirités équivalentes ou meilleures à ceux d'un LFSR.

Les séquences générées par l'AES et le LFSR ont été appliquées à trois circuits de la gamme des benchmark ISCAS'89. Le tableau 2 reporte le taux de couverture obtenu pour ces 3 circuits implantés avec 1, 16 ou 128 chaînes de scan.

Circuit	Patterns	Nombre de chaînes de scan	LFSR	AES
			FC (%)	FC (%)
s9234	21483	1	95.77	95.96
		16	95.54	95.96
		128	95.52	95.84
s13207	15000	1	99.37	95.83
		16	94.62	96.02
		128	86.48	95.42
s38584	84898	1	91.26	91.93
		16	90.82	92.25
		128	91	91.53

Tableau 2: taux de couverture

Les taux de couvertures obtenus sont du même ordre de grandeur. Sur l'ensemble des simulations effectuées, l'AES donne plus souvent de meilleurs résultats.

L'AES apparaît comme une alternative intéressante à l'implantation d'un générateur de vecteurs de test.

IV. ANALYSEUR DE SIGNATURE

Le rôle d'un analyseur de signature est de compacter les réponses du CUT en un seul mot appelé « signature ». La signature ainsi obtenue est comparée à celle attendue. Si elles diffèrent, ceci signifie qu'au moins une réponse est incorrecte et que le circuit est donc fautif. Cependant, une séquence contenant une ou plusieurs réponses incorrectes peut produire la signature attendue. Pour cette raison, la qualité d'un analyseur de signature est évaluée en terme de probabilité de masquage.

La probabilité de masquage de la structure proposée est égale à $\frac{1}{2^m} - \left(\frac{1}{2^m}\right)^n$

où m représente la taille de la signature (128 bits) et n le nombre de réponses du CUT. Pour un grand nombre de réponses (n grand), la probabilité de masquage est égale à $0{,}29.10^{-38}$ et donc négligeable.

BIBLIOGRAPHIE

[1] B. Yang, K. Wu, R. Karri, "Scan-based Side-Channel Attack on Dedicated Hardware Implementations on Data Encryption Standard", Proc. International Test Conference (ITC 2004), pp 339-344

[2] B.Yang, K.Wu, R.Karri, "Secure Scan: A Design-for-Test Architecture for Crypto Chips", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems TCAD 06, Oct. 06, Vol 25, Issue: 10, pp 2287-2293

[3] FIPS 197, Advanced Encryption Standard (AES), November 2001.

[4] B. Yang, R. Karri, "Crypto BIST: A Built-In Self Test Architecture for Crypto Chips", 2nd Workshop on fault diagnosis and tolerance in cryptography (FDTC 2005), pp 95-108.

[5] NIST Special Publication 800-22 (with revisions dated May 15, 2001) "A statistical test suite for random and pseudorandom number generators for cryptographic applications"

[6] P. Hellekalek, S. Wegenkittl, "Empirical evidence concerning AES", ACM Trans. Model. Comput. Simul., Vol. 13, Issue 4 (Oct. 2003), pp 322-333.