J Electron Test (2011) 27:99–108
DOI 10.1007/s10836-011-5204-0

Balanced Secure Scan: Partial Scan Approach for Secret Information Protection

Michiko Inoue · Tomokazu Yoneda · Muneo Hasegawa · Hideo Fujiwara

Received: 30 September 2009 / Accepted: 1 February 2011 / Published online: 22 February 2011

Abstract Scan-based Design-for-Testability technique is widely used to enhance the testability. However, it increases the vulnerability to attacks through scan chains for secure chips such as cryptographic circuits with embedded secret keys. This paper proposes a secure scan design method which protects the circuits containing secret information such as cryptographic circuits from scan-based side channel attacks. The proposed method prevents the leakage of secret information by partial scan design based on a balanced structure. We also guarantee the testability of both the design under test and DFT circuitry, and therefore, realize both security and testability. Experiments for RSA circuit shows the effectiveness of the proposed method.

Keywords Cryptographic circuit · Partial scan · Security · Testability

Responsible Editor: C. Metra

This work was supported in part by Japan Society for the Promotion of Science (JSPS) under Grants-in-Aid for Scientific Research ((B)20300018, (C)18500038).

M. Inoue (✉) · T. Yoneda · M. Hasegawa · H. Fujiwara
Nara Institute of Science and Technology,
Takayama, Ikoma, 630-0192, Japan
e-mail: kounoe@is.naist.jp

T. Yoneda
e-mail: yoneda@is.naist.jp

H. Fujiwara
e-mail: fujiwara@is.naist.jp

1 Introduction

Cryptographic circuits are often embedded in secure systems requiring high throughput. Since such cryptographic circuits include encryption and/or decryption keys in the circuits, their security is important issue.

Scan design is a widely used Design-for-Testability technique, which enables FFs in sequential circuits to be directly controlled and observed through scan chains. However, it is too vulnerable in scan-based side-channel attacks if test pins are still available after production tests. On normal microcontrollers, the test circuitry remains fully accessible after the test, while it is common practice to disable access to these test circuits using fuse or anti-fuse in smartcard processors [8]. However, it is possible to reconnect these test circuits with microprobes or FIB editing [8]. Therefore, even if access to the test pins are simply disabled after the production test, chips with scan design are vulnerable in scan-based side-channel attacks. Several works have been investigated to achieve both security and testability for scan design.

Hely et al. [5] introduced an authentication mechanism. If the authentication is failed, FF order in a scan chain is periodically changed, and it makes impossible for the attackers to analyze the circuits. They [4] also proposed a test controller which isolates registers with secret information and resets values of the other registers at the beginning of test mode to protect secret information to be leaked. However, the method has room for improvement on testability since the method cannot test secret registers and the test controller is tested by only checking whether all the registers are reset.

Lee et al. [9] presented a Lock & Key technique where a scan chain is divided into smaller subchains and access to subchains are randomized for unauthorized users. The method requires a large test controller including FSM for the authorization, LFSR to randomize the subchain order. Moreover, testability of the test controller is not mentioned.

Yang et al. [13] proposed a Mirror Key Register (MKR) that keeps a copy of secret key in normal mode. The proposed method introduced a test controller that can transfer the circuit to test mode only when it is powered on. That prevents attackers from extracting the data on the way of encryption or decryption. In test mode, the register storing the secret information is isolated from the remaining circuit, and the MKR is reset at the beginning of the test mode. Therefore, the secret information is never leaked in test mode. The proposed method enables the whole circuit to be tested except the register for the secret information and the signal lines between the register and the MKR.

Paul et al. [11] proposed a VIm-Scan which utilizes some FFs in a scan chain for authentication to move to test mode. In this method, the circuit can move to test mode only if the proper sequence of test keys are inputed to these FFs. This method is superior to the other methods in a sense that the test controller is testable. However, it needs a long sequence of test keys to move test mode, and it takes a long time for authentication.

Chandran et al. [1] proposed a SS-KTC which uses two-level key authorization to move to the test mode. The proposed method integrates the test keys into the test patterns by utilizing the don't care bits and achieves low area overhead.

In this paper, we propose a new secure scan method as a combination of balanced structure [3] and kernel logic confusion. The balanced structure is a structure for testable sequential circuits. We adopt a partial scan to make a *kernel* balanced, where a kernel is the portion of the circuit excluding the scan chains. The partial scan protects non-scan registers completely from scan-based attacks. In addition, we introduce a mechanism to confuse the kernel logic in test mode to protect scan registers. Our proposed method makes the circuit behavior in test mode completely different from normal mode. We use a test controller that transfers the circuit to test mode only when it is powered on like [13]. The proposed test controller is very small and fully testable. Therefore, the proposed partial scan method based on balanced structure guarantees high security and high testability simultaneously. Moreover, because of the nature of partial scan, the proposed method can achieve lower area overhead and reduce over-testing compared to full scan design.

The remainder of the paper is organized as follows. In Section 2, we assume the potential attackers and discuss the vulnerability of some well-used cryptographic circuits. Section 3 introduces a balanced structure, and we propose a new secure scan design and evaluate it in Sections 4 and 5. Finally, we conclude the paper in Section 6.

2 Assumption on Attackers

In general, the security requirements are varied with attackers' knowledge level. In this paper, we suppose the attackers can use only generally obtainable knowledge, and assume the attackers as follows.

1. Attackers know the cryptographic algorithm to be implemented as a circuit, and can suppose some candidates for RTL design.
2. Attackers can identify the test pins if scan design is adopted as DFT.
3. Attackers do not know detailed information on the gate level design or DFT including the order of FFs in the scan chains.

Some scan-based side-channel attacks are reported for DES (Data Encryption Standard) [12] and AES (Advanced Encryption Standard) [13]. These attacks identify the order of scan FFs by applying input patterns repeatedly, and discover the secret information from the analysis of registers.

RSA (Rivest-Shamir-Adleman) is also vulnerable to several side-channel attacks including timing attack [6] and simple power analysis [7]. RSA needs a secret key for decryption in which modulo-multiply operations and modulo-square operations are repeatedly executed. The attack to RSA exploits the fact that the executions of modulo-multiply operations depends on a bit pattern of the secret key. It can be applied to scan-based side-channel attack where the secret key can be discovered by analyzing scan FFs repeatedly.

3 Balanced Structure

We use a balanced structure [3] as a testable sequential circuit structure. To use the structure, we give some definition according as [3].

A synchronous sequential circuit S consists of blocks of combinational logic and registers. A register is a collection of one or more FFs controlled by the same

control signal. There are two kinds of registers. A LOAD register is a register whose FFs have no explicit LOAD ENABLE control signal, and a HOLD register is a register whose FFs have an explicit LOAD ENABLE control signal. The LOAD ENABLE control signals must be controlled by primary inputs. The combinational logic in S are partitioned into clouds, each of which is a maximal region of connected combinational logic such that its inputs are either primary inputs or outputs of FFs and its outputs are either primary outputs or inputs to FFs.

A *topology graph* of a sequential circuit is a directed graph $G = (V, A, H, w)$ in which V is a set of clouds, A is a set of registers between clouds, $H \subset A$ is a set of HOLD registers, and $w : A \rightarrow Z^+$ (positive integers) denotes the number of FFs in each register. The weight $w(a)$ of a register a represents the cost of converting the register a into a scan register.

Definition 1 (Balanced Structure [3]) Let S be a synchronous sequential circuit with a topology graph $G = (V, A, H, w)$. S is said to be balanced structure if

1. G is acyclic,
2. $\forall v_1, v_2 \in V$, all directed paths from v_1 to v_2 are of equal length, and
3. $\forall h \in H$, if h is removed from G, the resulting graph is disconnected.

Figure 1a shows an example of a sequential circuit S_1. The sequential circuit consists of clouds $C1, \cdots, C5$, a HOLD register $R1$ and LOAD registers $R2, \cdots, R7$. Figure 1b shows another example of a sequential circuit S_2 to clarify the definition of a cloud, where there is an additional connection from $C4$ to $C5$ in S_1. In this case, both $C4$ and $C5$ are combinationally connected, and hence, a region $C6$ including $C4$ and $C5$ becomes a cloud. The sequential circuit S' is obtained from S_1 by replacing two registers $R4$ and $R7$ with primary inputs and outputs. Figure 1c and d show the sequential circuit S' and its topology graph. The topology graph satisfies Definition 1 and S' is a balanced structure.

If a sequential circuit is a balanced structure, we can obtain a test sequence for the circuit using test patterns for its combinational equivalent. A combinational equivalent for a balanced sequential circuit S is a combinational circuit obtained from S by replacing each FF in every register in S with a wire. Let d be the longest directed path length in a topology graph of S. If some fault f is detected when applying an input pattern t to S in continuous d clocks, t is said to be a single-pattern test for f.

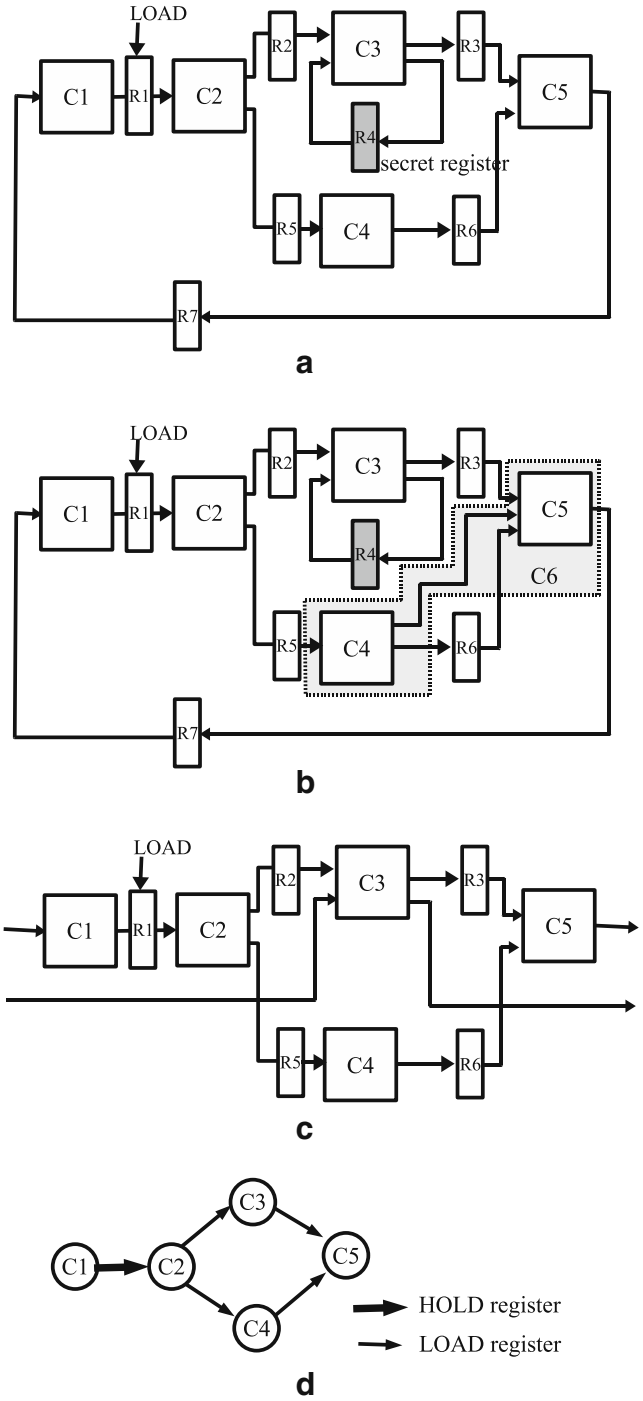

Fig. 1 **a** Sequential circuit S_1, **b** sequential circuit S_2, **c** balanced sequential circuit S', **d** topology graph

Theorem 1 [3] *Let S and C be a balanced sequential circuit and its combinational equivalent, respectively. Then any complete test set for all detectable stuck-at faults in C is a complete single-pattern test set for all detectable stuck-at faults in the combinational logic of S.*

In [3], a partial scan method BALLAST that obtains a balanced sequential circuit as a *kernel* is proposed. A heuristic algorithm is also proposed to select scan registers resulting in the minimum area overhead.

4 Balanced Secure Scan

4.1 Outline

We propose a new secure scan method based on a balanced structure, called *balanced secure scan*. The proposed balanced secure scan protects secret information stored in some registers in a circuit under test. We call the registers which keep the secret information or whose values depend on the secret information *secret registers*. In this paper, we assume that secret registers are designated in advance. The outline of the proposed method is shown as follows.

1. Select scan registers so that the kernel becomes a balanced structure and the number of FFs in secret registers selected as scan registers is minimized.
2. If some secret registers are selected as scan registers, add confusion circuits into the kernel to confuse the values of the secret registers in test mode while preserving balanced structure.

The proposed method protects some secret registers by partial scan, and protects the other secret registers by kernel logic confusion. The kernel logic confusion realizes different behaviors between normal and test modes and prevents the secret information from leakage from the scannable secret registers in test mode. Moreover, we propose a test controller that transfers the circuit to test mode only when the circuit is powered on, so that that the scan shift operation is unavailable once the circuit becomes normal mode. The proposed confusion circuit and test controller are testable, and therefore, the proposed method guarantees that the additional circuit does not reduce the security by their malfunction. We will analyze the security and the testability realized in the proposed method later in Sections 5.1 and 5.2, respectively.

Figure 2 shows an example where the proposed method is applied to a sequential circuit in Fig. 1a. We first make the kernel balanced by selecting $R4$ and $R7$ as scan registers. Since the selected $R4$ is a secret register, we then confuse the input of $R4$ using the output of a cloud $C4$.

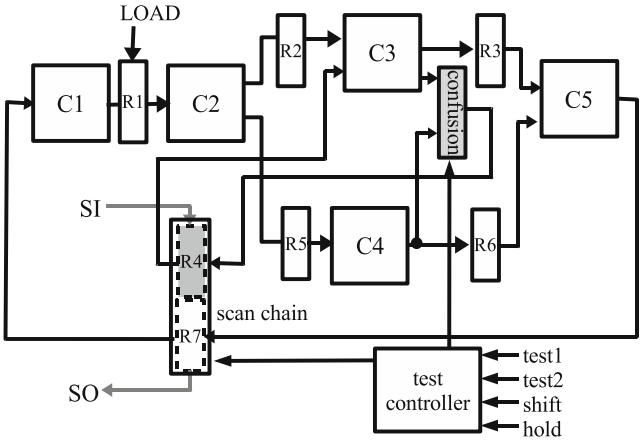

Fig. 2 Proposed method

4.2 Scan Register Selection

For a given sequential circuit, we first select scan registers so that the kernel becomes a balanced structure, where we try to minimize the number of FFs in the secret registers selected as scan register and then minimize the number of FFs in the scan registers.

We select scan registers by an enhanced method of the scan register selection proposed in [3]. The method [3] selects scan registers for a given topology graph $G = (V, A, H, w)$ to make the kernel balanced, where the total weight $\sum_{r \in SR} w(r)$ of selected scan registers is minimized, where SR is a set of selected registers.

In the proposed method, we first replace the weight $w(r)$ of each secret register r with $C \cdot w(r)$ by multiplying a sufficient large value C, then apply the scan register selection method in [3]. Consequently, we select a small number of scan FFs in secret registers and a small number of scan FFs. For example, $C > \sum_{r \in A} w(r)$ is enough to give higher priority to be selected as scan registers to non-secret registers, since a weight of a register is a positive integer.

4.3 Kernel Logic Confusion

If some secret registers are selected as scan registers, we confuse values of the registers only in test mode. In the kernel logic confusion, the inputs of the secret registers are exclusive-ORed with other signals (Fig. 3). The additional connections are chosen while preserving balanced structure. In the current version, we randomly choose the signals that confuse the secret scan registers within the range that preserves a balanced structure. For example, the input of a secret register

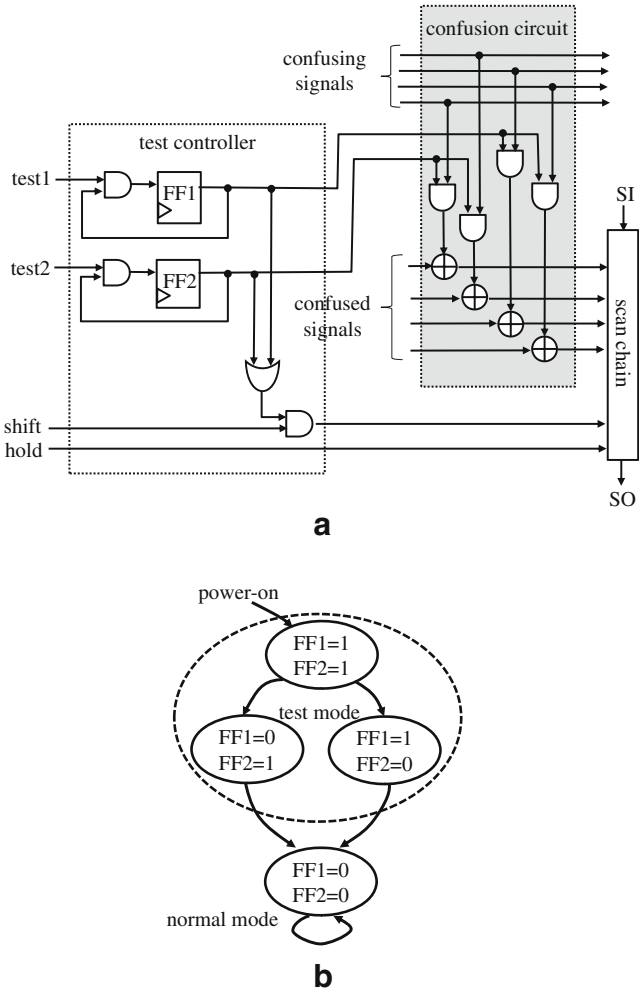

Fig. 3 **a** Confusion circuit and test controller; **b** state transition of test controller

$R4$ is confused with the output of $C4$ in Fig. 2. In this case, two clouds $C3$ and $C4$ are combinationally connected through a confusion circuit, and hence, these $C3$, $C4$ and the confusion circuit compose a cloud. The insertion of the confusion circuit preserves a balanced structure of a kernel, and this guarantees the kernel is testable. Figure 3a shows a confusion circuit. Though this circuit is static, a value of a confused signal is flipped only if a confusing signal has value 1. That is, the signal seems to be confused dynamically according to a value of the confusing signal.

Since the kernel logic confusion is needed only in test mode, we mask the signals which confuse the secret scan registers in normal mode. These mask elements are controlled by a test controller.

4.4 Test Controller

We propose a test controller to switch normal mode and test mode. Figure 3a shows a test controller and a confusion circuit, and Fig. 3b shows a state transition of the test controller. The test controller has four inputs: $test1$, $test2$, $shift$, and $hold$, where $test1$ and $test2$ control the mode of the circuit, and $shift$ and $hold$ control the scan chains. The shift operation is available only when $shift = 1$ holds in test mode. The test controller has two FFs, and the circuit is said to be in test mode when at least one FF in the test controller has a value 1. The proposed test controller has the following features.

1. We adopt power-on set FFs to the test controller, and this brings the circuit to be test mode when it is powered on and once the circuit moves to normal mode it cannot go back to test mode while being powered on.
2. Values of registers in normal mode cannot be shifted out through scan chains. Since the circuit cannot be transferred from a normal mode to test mode, it is impossible to make the circuit operate for several clock cycles in normal mode and then shift out the register values using the scan operation.
3. The kernel logic confusion and scan shift operation are available only in test mode.
4. The test controller and the kernel logic confusion circuit are testable. Since the circuit is in test mode if at least one FF in the test controller has a value 1, two FFs can have value 0 exclusively. That is, we can fully control each FF value in test mode. Since the circuit under test including the confusion circuits is a balanced structure, and in addition, the test controller has a quite simple structure, both the test controller and the confusion circuits are testable.

5 Evaluation

5.1 Security Analysis

In general, scan-based side-channel attacks analyze the circuits with secret information based on the implemented algorithm and shifted-out FF values. The known attack methods for AES or DES repeat normal operations and scan operations, identify the order of FFs in a scan chain, and then analyze the secret information. That is, information on the implemented

algorithm and/or all the FF values are necessary to analyze secret information.

In the proposed method, a part of registers are protected as non-scan registers. Even if some secret registers are selected as scan registers, their values are confused in test mode. In addition, the circuit cannot go back to test mode once it moves to normal mode, and hence, any attacker cannot extract FF values in normal mode. Since attackers are assumed to have no knowledge on gate level design, they cannot know the algorithm implemented by confusion circuits. In addition, a part of FF values are not shifted out and the attackers cannot know all the FF values. From these facts, it is impossible to analyze secret information.

In the proposed method, the inputs of secret registers selected as scan registers are confused. However, we do not think that all the bits of such inputs have to be confused. Of course, such security level depends on encryption algorithms. To discuss which registers or how many bits should be confused, we need more precise quantitative analysis for security. This is one of future works, and we do not discuss any more in this paper.

The test controller and the confusion circuits are testable, and this testability avoids a risk of secret information leakage due to some malfunction of these additional circuit.

5.2 Testability Analysis

We analyze testability of the circuit obtained by the proposed method.

Theorem 2 *For the circuit obtained by the balanced secure scan, we can generate test patterns for any detectable stuck-at faults in the combinational logic of a given circuit before DFT using combinational ATPG.*

Proof In the proposed method, we first select scan registers so that the kernel becomes a balanced structure. Theorem 1 implies that we can generate test patterns for all the detectable stuck-at faults in the kernel by applying combinational ATPG to its combinational equivalent if we do not add any confusion circuit. Since confusion circuit confuses the inputs of the secret scan registers using exclusive-OR gates, any error propagated to confused signal can be propagated to the scan register if the other input of exclusive-OR is an error-free value 0 or 1. We can control the value to 0 independently from the confused signal by setting the corresponding FF in the test controller to 0, and hence, the theorem holds. □

For additional DFT circuitry, we consider lines bound to the control inputs to the scan registers and the others separately.

Theorem 3 *All the single stuck-at faults in the test controller and the confusion circuit are detectable, if any confusing signals can be controllable to both 0 and 1, and errors propagated to the control inputs of the scan registers can be observable.*

Proof First, we prove the detectability of faults on lines $a, b, \cdots, g$ in Fig. 4 that shows lines relevant to any FF in the test controller and any confused signal. Suppose there is some single stuck-at fault on one of these lines. Since we assume a single stuck-at fault, a confused signal s_2 is fault-free, and hence, we can control the signal to an error-free value 0 or 1. Therefore, each fault is detectable if its error can be propagated to a line g. Now we consider each stuck-at-fault on these lines. Since the FF is set to 1 when powered on, errors of stuck-at-0 faults on lines d and e and errors of stuck-at faults on lines f and g are propagated to the line g. Errors of stuck-at-0 faults on lines a, b and c are propagated after applying one clock with $test1(\text{or } test2) = 1$. Errors of stuck-at-1 faults on lines a, c, d and e are propagated to g after applying one clock with $test1(\text{or } test2) = 0$. Finally, an error of a stuck-at-1 fault on a line b is propagated to g after applying clocks with $test1(\text{or } test2) = 01$.

Figure 5 shows lines bound to the control inputs of the scan registers. From the assumption, errors propagated to l or m are detectable. Two lines i and h are independently controllable and k is controllable to 1, and hence, any stuck-at faults in lines h, i and j are detectable. □

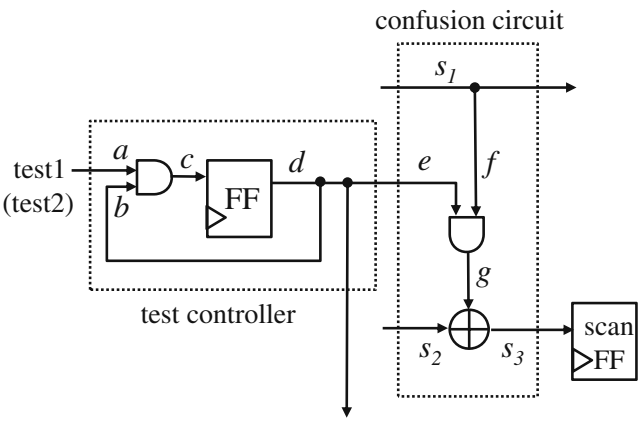

Fig. 4 Faults in the test controller and the confusion circuit

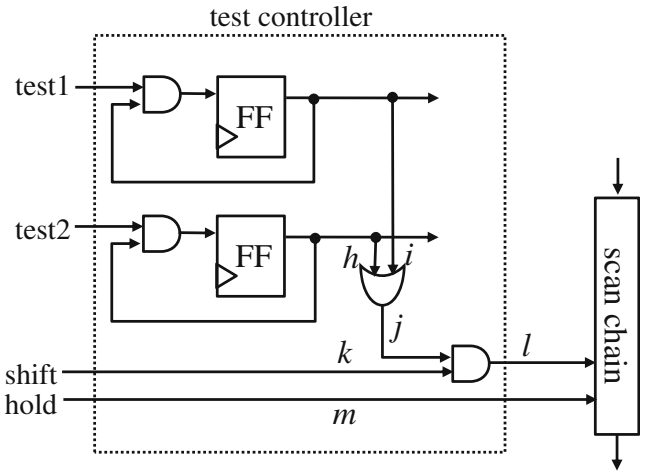

Fig. 5 Faults in lines bound to scan register control

Note that we do not discuss the detectability of lines s_1, s_2 and s_3 in Theorem 2. Since s_1 and s_2 are a confusing signal and a confused signal, respectively, their detectability is discussed in Theorem 2. Both single stuck-at-0 and single stuck-at-1 faults in a line s_3 are detectable, since single fault assumption implies fault-free value in a line s_2 and detectability of both stuck-at faults in a line g implies the detectability of stuck-at faults in a line s_3.

In the above theorem, we assume that any confusing signals can be controllable to both 0 and 1, and errors propagated to the control inputs of the scan registers can be observable. In general, the latter condition holds since the errors at the scan chain control inputs can be observable through a functional test for the scan chain. The former condition holds, if we can choose controllable confusing signals. We consider there are many signals controllable to both 0 and 1 in secure circuits such as cryptographic circuits (see [2] for example), and hence, the proposed balanced secure scan can provide a testable test controller.

5.3 Comparison with Full Scan

We evaluated the proposed method using an RSA decryption circuit. We use the 1024-bit RSA decryption circuit available as an open source IP core [10]. For the evaluation, we first modified the circuit to fit a *circuit model* described in [3]. We modified the circuit so that the registers have explicit LOAD ENABLE control signals if the registers have HOLD function. This modification increases the area of circuits from 302,292 gates to 331,566 gates.

We applied the proposed balanced secure scan and full scan methods to the modified circuit, and compared area overhead and testability. We used Design Compiler (Synopsys) for logic synthesis, DFT Compiler (Synopsys) to insert scan chains, and TetraMAX (Synopsys) for test generation on SunFireV4100 (3GHz AMD Opteron256, 16GB memory).

Figure 6 shows the RSA decryption circuit, which is composed of a datapath and a controller. The datapath has a modulo-square part (modsqr), and a modulo-multiply part (modmult). The RSA decryption circuit decodes an encrypted text y to a plaintext x using a secret key d and a public key m as $x = y^d \bmod m$. The encrypted text y is applied from indData and a plaintext x is obtained at cypher, and the keys d and m are stored in inMod and inExp, respectively in Fig. 6. The input ds is a start signal of decryption, and the output ready is a ready signal for the next input. The two parts modsqr and modmult actually have the same architecture, where it multiplies a multiplicand with a multiplier and takes modulo operation. The modulo-square part modsqr takes root as both a multiplicand

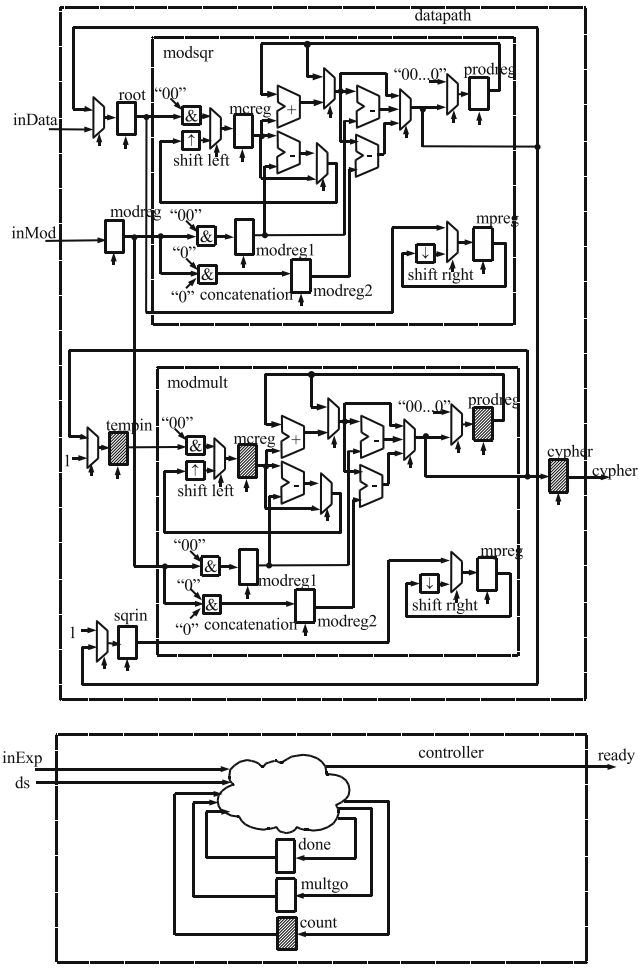

Fig. 6 RSA decryption circuit

Table 1 Area overhead (TC denotes the area for a test controller)

	Confusion	Area (gates)				Area overhead (gates)					
	(%)	Total	Comb.	Register	TC	Total	Total (%)	Confusion	MUX	Scan	TC
Original		331,566	206,042	125,524	0	0	0	0	0	0	0
Full scan		359,378	206,042	153,336	0	27,812	8.39	0	0	27,812	0
Proposed	100	361,578	218,466	143,088	24	30,012	9.05	12,288	136	17,564	24
	50	355,434	212,322	143,088	24	23,868	7.20	6,144	136	17,564	24
	25	352,362	209,250	143,088	24	20,796	6.27	3,072	136	17,564	24

and a multiplier, while the modulo-multiply part modmult takes tempin and sqrin as a multiplicand and a multiplier, resistively. Each part has 1024 bit wide inputs and an output, and internal data signals has two more bits to handle intermediate results. In both modsqr and modmult, modulo-multiplication is calculated by iteratively taking shift, add and modulo (subtraction in this case) operations.

The modulo-square part calculates y^{2^0} mod m, y^{2^1} mod m, y^{2^2} mod m, $\cdots$ in this order. The module-multiply part calculates a modulo-multiply operation with the output of the modulo-square part as needed. For example, for $d = 5 = 101_{(2)}$, the plain text x is obtained as $x = y^{101_{(2)}}$ mod $m = (y^{2^0}$ mod $m) \times (y^{2^2}$ mod $m)$ mod m. That is, the operation of the module-multiply part depends on a bit pattern of the secret key d. This is controlled by controller where the key in inExp is first stored in counter and it is shifted one by one while controlling the module-multiply operation.

Since the modulo-square part and a public key in inMod are not related with secret information, only the registers prodreg, mcreg in modmult, and registers count, cypher and tempin are designated as secret registers.

In the proposed method, we first separated the datapath and the controller by inserting primary input with MUXes and primary output to the signals between them to make the LOAD ENABLE control signals directly controllable from the primary inputs. This modification is to apply DFT for balanced structure [3]. We then selected scan registers to make the kernel balanced. As a result, registers mcreg, mpreg, modreg1, prodreg in both modsqr and modmult, and registers done, multgo, count are selected as scan registers. The secret registers selected as scan registers are mcreg and prodreg in modmult and count in the controller, and we confuse the inputs of these three registers. The bit-width of mcreg, prodreg and count are 1,026, 1,026 and 1,024, respectively. In the experiment, we confused 1,024 (100%), 512 (50%) and 256 (25%) bits of the input of each register, and compared these area overhead and testability.

Table 1 shows the area and the area overhead for the proposed method and the full scan design. We used the circuit after modification on LOAD ENABLE signals as an original circuit. The column *confusion* denotes confusion ratio. The column *comb*. denotes the combinational logic area including confusion circuits and MUXes for the datapath-controller isolation. The column *register* denotes the area for non-scan and scan registers. The column *TC* denotes the area for a test controller. The proposed method achieves a little bit larger area overhead than the full scan when confusing all the bits of input of the secret registers selected as scan register. Practically, it is not considered that we need to confuse all the bits to protect such registers. Our method achieve lower area overhead than the full scan design when we confuse a part of these bits.

Table 2 shows the test generation result for the combinational logic parts, where two keys in inMod and inExp are set to some fixed random values. The columns *confusion*, *fault*, *FC*, *FE*, *redundant*, *abort* and *TGT* are confusion ratio, the numbers of total stuck-at faults, fault coverage, fault efficiency, the number of identified redundant and

Table 2 Test generation result (inMod and inExp are fixed)

	Confusion (%)	Fault	FC (%)	FE (%)	Redundant	Abort	TGT (s)
Full scan	–	459,342	98.66	100.00	6,156	0	30.47
	100	496,268	96.10	100.00	19,374	0	80.81
Proposed	50	477,836	95.95	100.00	19,374	0	72.66
	25	468,620	95.87	100.00	19,373	0	68.31

Table 3 Test generation result (inMod and inExp are primary inputs)

Method	Confusion (%)	Fault	FC (%)	FE (%)	Redundant	Abort	TGT (s)
Full scan	–	459,342	99.999	100.000	4	0	29.51
	100	496,268	99,999	100.000	1	0	7.64
Proposed	50	477,836	99.999	100.000	1	0	6.98
	25	468,620	99.999	99.999	0	4	8.19

aborted faults, and test generation time, respectively. The combinational part for the proposed methods include the confusion circuits and therefore the number of faults are increased. We achieved complete (100%) fault efficiency for all the cases with reasonable test generation time. However, the proposed methods identified more redundant faults than the full scan. We considered this is because the register modreg is not selected as a scan register and there are redundant faults at both input and output of modreg which is connected with inMod with the fix value. However, in the case of the full scan design, modreg is selected as a scan register, and this makes the output of modreg testable.

To confirm this prediction, we gave an additional experiment for the case where inMod and inExp are primary inputs. Table 3 shows the result. In this case, there are little redundant faults for both the proposed method and the full scan design. That is, the most redundant faults are caused by the embedded fixed values. This implies that the proposed method can avoid over-testing.

5.4 Comparison with Related Works

We now qualitatively compare the proposed method with related works. Table 4 summarizes the comparison. All the related works are based on full scan design, and they incorporate additional circuitry to improve security.

Since the related works need additional circuits to full scan design, they evidently have larger area overhead than full scan design. On the other hand, the proposed method has comparable area overhead with full scan design, and we can conclude the proposed method has less area overhead with the related works.

The security of [1, 5, 9, 11] rely on authentication mechanisms ([5] does not explicitly mention an authentication mechanism, but the proposed scan chain scrambling is used when some authentication fails.) Therefore, the circuits are secure against scan-based side-channel attacks as long as the authentication keys are well protected. Other works [4, 13] and our proposed method do not need any authentication, and they provide special test modes to prevent secret information from leakage instead. Therefore, the circuits are secure against scan-based side-channel attacks as long as the test controllers can work well. In the work [4], JTAG test controller is augmented so that it resets all FFs before scan operation to eliminate secret information from FFs, and the function of the reset operation is checked online. However, verifying all the FFs is impractical since it needs large area overhead. In the work [11] and our proposed method, testing of the test controllers is discussed. Therefore, the security of the circuits are guaranteed.

Since all the related works are based on full scan design, the user logic circuits are as testable as full scan design in [4, 5, 9, 11], and a method [13] is less testable since it excludes registers with secret information from design under test. As we mentioned, our proposed method guarantees as high testability as full scan design for a circuit including the test controller and the kernel logic confusion circuit.

Table 4 Comparison with related works

Reference	Scan	Design for security	Test for additional circuit
[5]	Full	Scan chain scrambling, (authentication)	Not considered
[4]	Full	Scan FF reset, full/partial reset verification	Reset verification
[9]	Full	FSM, key comparator, LFSR, decoder	Not considered
[13]	Full	MKR, test controller	Not considered
[11]	Full	Pattern matching, counter	Testable
[1]	Full	Key authentication	Not considered
Proposed	Partial	Logic confusion	Testable

6 Conclusion

In this paper, we proposed a partial scan method to make sequential circuits testable without sacrificing security. The proposed method protects secret registers through partial scan and the kernel logic confusion, and hence guarantees high security of the circuits. On the other hand, a partial scan method based on the balanced structure guarantees high testability. In addition, the proposed method guarantees the testability of additional circuits, and this testability avoids a risk of secret information leakage due to some malfunction of these additional circuits.

Though full scan design is a de-facto standard for DFT method, this paper demonstrated a potential of partial scan design which can protect the secret information from scan-based side-channel attacks. The proposed method may affect the design flow, but it is very powerful and area-efficient method when highly security level is required for circuits.

References

1. Chandran U, Zhao D (2009) SS-KTC: a high-testability low-overhead scan architecture with multi-level security integration. In: Proceedings of VLSI test symposium, pp 321–326
2. Doulcier M, Flottes M-L, Rouzeyre B (2008) AES-based BIST: self-test, test pattern generation and signature analysis. In: Proceedings of IEEE international symposium on electronic design, test & applications, pp 314–321
3. Gupta R, Gupta R, Breuer M (1990) The BALLAST methodology for structured partial scan design. IEEE Trans Comput 39:538–544
4. Hely D, Bancel F, Flottes M-L, Rouzeyre B (2007) Securing scan control in crypto chips. J Electron Test Theory Appl 23:457–464
5. Hely D, Flottes M-L, Bancel F, Rouzeyre B, Berard N (2004) Scan design and secure chip. In: Proceedings of 10th IEEE international on-line testing symposium (IOLTS'04), pp 219–224
6. Kocher PC (1996) Timing attacks on implementations of diffie-hellman, RSA, DSS, and other sysetms. In: Proceedings of advances in cryptology—CRYPT0 '96, pp 104–113
7. Kocher P, Jaffe J, Jun B (1999) Differential power analysis. In: Proceedings of advances in cryptology—CRYPT0 '99, pp 388–397
8. Kommerling O, Kuhn MG (1999) Design principles for tamper-resistant smartcard processors. In: Proceedings of USENIX workshop on smartcard technology
9. Lee J, Tehranipoor M, Patel C, Plusquellic J (2007) Securing designs against scan-based side-channel attacks. IEEE Trans Depend Secure Comput 4:325–336
10. OPENCORES. "RSA processor." http://www.opencores.org/projects.cgi/web/rsa/overview
11. Paul S, Chakraborty RS, Bhunia S (2007) VIm-Scan: a low overhead scan design approach for protection of secret key in scan-based secure chips. In: Proceedings of 25th IEEE VLSI test symposium (VTS'07), pp 455–460
12. Yang B, Wu K, Karri R (2004) Scan based side channel attack on dedicated hardware implementations of data encryption standard. In: Proceedings of international test conference 2004 (ITC'04), pp 339–344
13. Yang B, Wu K, Karri R (2006) Secure scan: a design-for-test architecture for crypto chips. IEEE Trans Comput-Aided Des Integr Circuits Syst 25:2287–2293

Michiko Inoue received her B.E., M.E, and Ph.D. degrees in computer science from Osaka University in 1987, 1989, and 1995, respectively. She worked at Fujitsu Laboratories Ltd. from 1989 to 1991. She is an associate professor of Graduate School of Information Science, Nara Institute of Science and Technology (NAIST). Her research interests include distributed algorithms, parallel algorithms, graph theory and design and test of digital systems. She is a member of IEEE, the Institute of Electronics, Information and Communication Engineers, the Information Processing Society of Japan, and Japanese Society for Artificial Intelligence.

Tomokazu Yoeda received the B.E. degree in information systems engineering from Osaka University, Osaka, Japan, in 1998, and M.E. and Ph.D. degrees in information science from Nara Institute of Science and Technology, Nara, Japan, in 2001 and 2002, respectively. Presently he is an assistant professor in Graduate School of Information Science, Nara Institute of Science and Technology. His research interests include VLSI CAD, design for testability, and SoC test scheduling. He is a senior member of IEEE.

Muneo Hasegawa received B.E. degree in information science from Kansai University, Osaka, Japan, in 2006. His research interests include VLSI CAD, design for testability, and security of cryptographic circuits.

Hideo Fujiwara received the B.E., M.E., and Ph.D. degrees in electronic engineering from Osaka University, Osaka, Japan, in 1969, 1971, and 1974, respectively. He was with Osaka University from 1974 to 1985 and Meiji University from 1985 to 1993, and joined Nara Institute of Science and Technology in 1993. Presently he is a Professor at the Graduate School of Information Science, Nara Institute of Science and Technology, Nara, Japan. His research interests are logic design, digital systems design and test, VLSI CAD and fault tolerant computing, including high-level/logic synthesis for testability, test synthesis, design for testability, built-in self-test, test pattern generation, parallel processing, and computational complexity. He is the author of Logic Testing and Design for Testability (MIT Press, 1985). He received many awards including Okawa Prize for Publication, IEEE CS (Computer Society) Meritorious Service Awards, IEEE CS Continuing Service Award, and IEEE CS Outstanding Contribution Award. He served as an Editor and Associate Editors of several journals, including the *IEEE Transactions on Computers*, and *Journal of Electronic Testing: Theory and Application*, and several guest editors of special issues of *IEICE Transactions of Information and Systems*. Dr. Fujiwara is a fellow of the IEEE, IEICE and IPSJ, and a Golden Core member of the IEEE Computer Society.