LETTER

Properties of Generalized Feedback Shift Registers for Secure Scan Design

Hideo FUJIWARA[†a)], ***Fellow and*** **Katsuya FUJIWARA**[††], ***Member***

SUMMARY In our previous work [12], [13], we introduced generalized feed-forward shift registers (GF^{2}SR, for short) to apply them to secure and testable scan design. In this paper, we introduce another class of generalized shift registers called *generalized feedback shift registers* (GFSR, for short), and consider the properties of GFSR that are useful for secure scan design. We present how to control/observe GFSR to guarantee scan-in and scan-out operations that can be overlapped in the same way as the conventional scan testing. Testability and security of scan design using GFSR are considered. The cardinality of each class is clarified. We also present how to design *strongly secure* GFSR as well as GF^{2}SR considered in [13].

key words: *design-for-testability, scan design, generalized feedback/feed-forward shift registers, security, scan-based side-channel attack*

1. Introduction

Scan design is a powerful design-for-testability (DFT) technique that offers high controllability and observability over a chip and yields high fault coverage [1]. However, it also allows reverse engineering, which contradicts security. It is essential to protect secrete data from side-channel attacks and other hacking schemes [2]. Hence, it is important to find an efficient DFT approach that satisfies both security and testability. Various approaches to secure scan design have been reported [3]–[9]. We have reported a secure and testable scan design approach by using extended shift registers called *"SR-equivalents"* that are functionally equivalent but not structurally equivalent to shift registers [10] and *"SR-quasi-equivalents"* [11]. The proposed approach only replaces part of the original scan chains to SR-equivalents or SR-quasi-equivalents, which satisfy both testability and security of digital circuits. This method requires very little area overhead and no performance overhead.

We then introduced a new class of extended shift registers called *generalized feed-forward shift registers* (GF^{2}SR, for short) by relaxing the condition of the SR-equivalents and SR-quasi-equivalents and considered the testability and security of GF^{2}SR [12]. In [13], we introduced a more secure concept called *strong security* such that no internal state of strongly secure circuits leaks out, and presented how to design such strongly secure GF^{2}SRs.

Manuscript received August 14, 2015.
Manuscript revised December 4, 2015.
Manuscript publicized January 21, 2016.

[†]The author is with Osaka Gakuin University, Suita-shi, 564–8511 Japan.
[††]The author is with Akita University, Akita-shi, 010–8502 Japan.
a) E-mail: fujiwara@ogu.ac.jp

DOI: 10.1587/transinf.2015EDL8183

In this paper, we introduce another class of generalized shift registers called *generalized feedback shift registers* (GFSR, for short), and consider the properties of GFSR that are useful for secure scan design. We present how to control/observe GFSR to guarantee scan-in and scan-out operations that can be overlapped in the same way as the conventional scan testing. Testability and security of scan design using GFSR are considered. The cardinality of each class is clarified. We also present how to design *strongly secure* GFSR as well as GF^{2}SR considered in [13].

2. Generalized Shift Registers

In our previous works [10], [11], to organize secure and testable scan design, we introduced five types of linear structured shift registers called *inversion-inserted SR (I^{2}SR), linear feed-forward SR (LF^{2}SR), linear feedback SR (LFSR), inversion-inserted linear feed-forward SR (I^{2}LF^{2}SR) and inversion-inserted linear feedback SR (I^{2}LFSR)*. In [12], we then introduced an extended class called *generalized feed-forward shift registers (GF^{2}SR)*, shown in Fig. 1 (a). In this figure, $f_0, f_1, \ldots, f_k$ are arbitrary logic functions. Figures 1 (b) and (c) show examples of 3-stage GF^{2}SRs, R_1 and R_2. Generally, for any GF^{2}SR with k flip-flops, the output z at time $t+k$ behaves in accordance with the following equation.

$$z(t+k) = x(t) \oplus f(x(t+1), x(t+2), \ldots, x(t+k)).$$

Here, we introduce another class of generalized shift registers called *generalized feedback shift registers (GFSR)*, shown in Fig. 2 (a). Figures 2 (b) and (c) show examples of 3-stage GFSRs, R_3 and R_4. The difference between GFSR and GF^{2}SR is whether the structure is feedback type or feed-forward type. From the feedback structure of Fig. 2 (a), we can see that for any GFSR with k flip-flops, the output z at time $t+k$ behaves in accordance with the following equation.

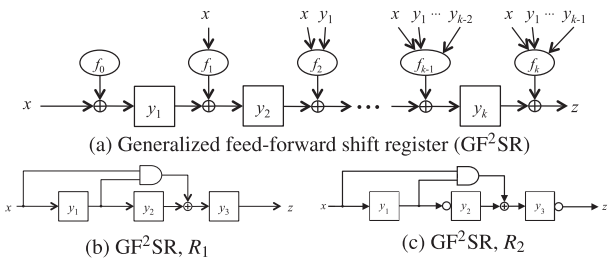


Fig. 1 Generalized feed-forward shift register (GF^{2}SR).

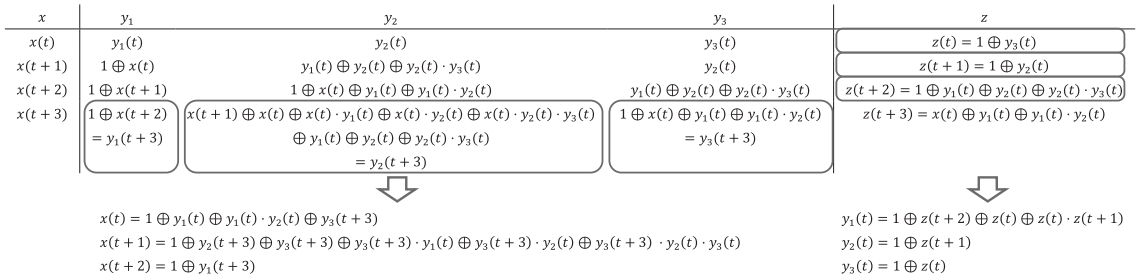


Fig. 4 Symbolic simulation of R_4.

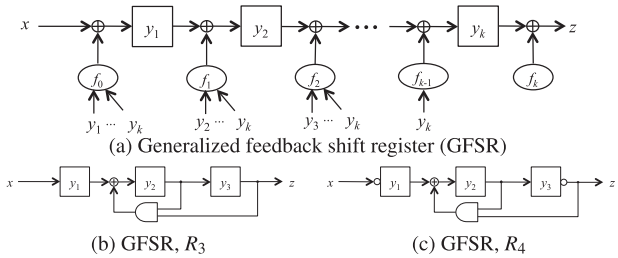


Fig. 2 Generalized feedback shift register (GFSR).

x	y_1	y_2	y_3	z
$x(t)$	$y_1(t)$	$y_2(t)$	$y_3(t)$	$z(t) = y_3(t)$
$x(t+1)$	$x(t)$	$y_1(t) \oplus y_2(t) \cdot y_3(t)$	$y_2(t)$	$z(t+1) = y_2(t)$
$x(t+2)$	$x(t+1)$	$x(t) \oplus y_1(t) \cdot y_2(t) \oplus y_2(t) \cdot y_3(t)$	$y_1(t) \oplus y_2(t) \cdot y_3(t)$	$z(t+2) = y_1(t) \oplus y_2(t) \cdot y_3(t)$
$x(t+3)$	$x(t+2)$	$x(t+1) \oplus x(t) \cdot y_1(t) \oplus x(t) \cdot y_2(t) \cdot y_3(t) \oplus y_1(t) \cdot y_2(t) \oplus y_2(t) \cdot y_3(t)$	$x(t) \oplus y_1(t) \cdot y_2(t) \oplus y_2(t) \cdot y_3(t)$	$z(t+3) = x(t) \oplus y_1(t) \cdot y_2(t) \oplus y_2(t) \cdot y_3(t)$

Fig. 3 Symbolic simulation of R_3.

$$z(t+k) = x(t) \oplus f(y_1(t), y_2(t), \ldots, y_k(t)).$$

Consider a 3-stage GFSR, R_3, given in Fig. 2 (b). By using symbolic simulation, we can obtain the output $z(t+3) = x(t) \oplus y_1(t)y_2(t) \oplus y_2(t)y_3(t)$ as shown in Fig. 3.

3. How to Control/Observe GFSRs

For a generalized shift register, GFSR/GF^{2}SR, the following two problems are important in order to utilize the generalized shift register as a scan shift register in testing. One problem is to generate an input sequence to transfer the circuit into a given desired state. This is called *state-justification problem*. The other problem is to determine the initial state by observing the output sequence from the state. This is called *state-identification problem*. In [12], we showed the following properties of GF^{2}SR.

Let C be any circuit of GF^{2}SR with k flip-flops, (1) for any internal state of C a transfer sequence (of length k) to the state (final state) can be generated only from the connection information of C, independently of the initial state, and (2) any present state (initial state) of C can be identified from the input-output sequence (of length k) and the connection information of C.

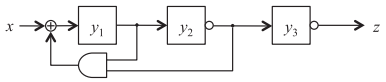


Fig. 5 GFSR, R_5.

The above properties can be easily seen from the feedforward structure of GF^{2}SR. In contrast, the feedback structure of GFSR derives the following properties.

Let C be any circuit of GFSR with k flip-flops, (1) for any internal state of C a transfer sequence (of length k) to the state (final state) can be generated from a given initial state and the connection information of C, and (2) any present state (initial state) of C can be identified only from the output sequence (of length k) and the connection information of C, independently of the initial state and the input sequence.

Consider a GFSR, R_4, of Fig. 2 (c). Figure 4 shows the result of symbolic simulation. As illustrated in the figure, we can derive equations to obtain an input sequence $(x(t), x(t+1), x(t+2))$ that transfers R_4 to the desired final state $(y_1(t+3), y_2(t+3), y_3(t+3))$. The transfer sequence depends on the initial state $(y_1(t), y_2(t), y_3(t))$. Similarly, we can derive equations to determine uniquely the initial state $(y_1(t), y_2(t), y_3(t))$ only from the output sequence $(z(t), z(t+1), z(t+2))$.

4. Application to Scan Testing

A scan-designed circuit under consideration consists of a single or multiple scan chains and the remaining combinational logic circuit (*kernel*). A scan chain can be regarded as a circuit consisting of a shift register with multiplexers that select the normal data from the combinational logic circuit and the shifting data from the preceding flip-flop. Here, we replace the shift register with a GFSR. However, to reduce the area overhead as much as possible, not all scan chains are replaced with extended scan chains. Only parts of scan chains necessary to be secure, e.g. secret registers, are replaced with GFSRs, and the size of the extended scan chains is large enough to make it secure. The delay overhead due to additional logic and Exclusive-OR gates influences only scan operation, and hence there is no delay overhead for normal operation.

For a GFSR, the initial state can be identified only from

the output sequence. However, the information of not only the final state but also the initial state is needed to generate a transfer sequence. Hence, at first of scan testing, it is necessary to identify the initial state by observing the output sequence. After knowing the initial state of the scan testing, both state-justification and state-identification can be performed simultaneously, i.e., both scan-in and scan-out operations can be overlapped.

From the above observation, we can easily generate scan-in and scan-out sequences such that both scan-in and scan-out operations can be overlapped and hence testing can be done in the same way as the conventional scan testing. The test sequence is of the same length as the conventional scan design. There is no need to change traditional ATPG algorithm though a logic implication process is needed only for the extended shift register after ATPG.

In [12], we also showed that GF^{2}SR can be used for secure scan path design. Comparing the properties of GF^{2}SR and GFSR mentioned in the previous section, we can see the following. As for state-justification, the scan-in operation for GF^{2}SR is easier than GFSR. On the other hand, as for state-identification, the scan-out operation for GFSR is easier than GF^{2}SR. Although there are those differences, both can be used for secure scan path design.

5. Security of GFSRs

When we consider a secure scan design, we need to assume what the attacker knows and how he can potentially make the attack. Here, we assume that *the attacker does not know the detailed information in the gate-level design, and that the attacker knows the presence of test pins (scan in/out, scan, and reset) and modified scan chains. However, he does not know the structure of extended scan chains.* Based on this assumption, we consider the security to prevent scan-based attacks.

A circuit C with a single input, a single output, and k flip-flops is called *scan-secure* if the attacker cannot determine the structure of C.

Consider two different structured 3-stage GFSRs, R_4 and R_5, shown in Figs. 2 (c) and 5. From the results of symbolic simulation, we can see their outputs $z(t+3)$ are the same, i.e., $z(t+3) = x(t) \oplus y_1(t) \oplus y_1(t)y_2(t)$. Therefore, their input/output behaviors after time $t+3$ are the same. On the other hand, their internal state behaviors are not the same because of their different structures. Hence one cannot control/observe internal states unless one knows the structure of the circuit, which means one cannot determine the structure of the circuit only from input/output behaviors, and hence they are scan-secure.

Next, let us consider the security level by clarifying the cardinality of the class of GFSRs. The security level of the secure scan architecture based on GFSR is determined by the probability that an attacker can guess right the structure of the GFSR used in the scan design, and hence the attack probability approximates to the reciprocal of the cardinality of the class of GFSR.

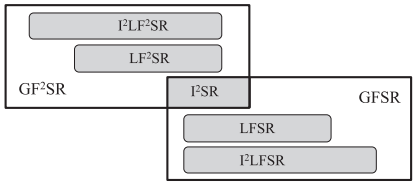


Fig. 6 Cover relation among classes.

Table 1 Cardinality of each class.

Class	# of circuits in the class
I^{2}SR	$2^{k+1}-1$
LF^{2}SR	$2^{k(k+1)/2}-1$
LFSR	$2^{k(k+1)/2}-1$
I^{2}LF^{2}SR	$(2^{k(k+1)/2}-1)(2^{k+1}-1)$
I^{2}LFSR	$(2^{k(k+1)/2}-1)(2^{k+1}-1)$
GF^{2}SR	$2^{(2^{(k+1)}-1)}-1$
GFSR	$2^{(2^{(k+1)}-1)}-1$

The class of GF^{2}SR covers I^{2}SR, LF^{2}SR, and I^{2}LF^{2}SR. The class of GFSR covers I^{2}SR, LFSR, and I^{2}LFSR. So, we have the covering relation as shown in Fig. 6. In [11], we showed the cardinality of each class of linear structured circuits (I^{2}SR, LF^{2}SR, LFSR, I^{2}LF^{2}SR and I^{2}LFSR). In [12], we showed the cardinality of the class of GF^{2}SR is $2^{(2^{(k+1)}-1)}-1$. Obviously, the cardinality of the class of GFSR is the same as GF^{2}SR. The summary of the cardinality of each class is shown in Table 1.

6. How to Design Strongly Secure GFSRs

Consider again the GFSR, R_3, of Fig. 2 (b) and the result of symbolic simulation shown in Fig. 3. When $y_1(t) = y_2(t) = 0$, it holds that $(x(t), x(t+1), x(t+2)) = (y_3(t+3), y_2(t+3), y_1(t+3))$, i.e., any input sequence $(x(t), x(t+1), x(t+2))$ that transfers R_3 to the desired final state $(y_1(t+3), y_2(t+3), y_3(t+3))$ becomes $(y_3(t+3), y_2(t+3), y_1(t+3))$ when $y_1(t) = y_2(t) = 0$. This means R_3 behaves in the same way as a shift register during scan-in operation when $y_1(t) = y_2(t) = 0$, and hence it is not secure when the attacker regards R_3 as a shift register and tries to initialize it. Similarly, when $y_2(t) = 0$ or $y_3(t) = 0$, it holds that $(y_1(t), y_2(t), y_3(t)) = (z(t+2), z(t+1), z(t))$, i.e., the output sequence $(z(t), z(t+1), z(t+2))$ equals to $(y_3(t), y_2(t), y_1(t))$ when $y_2(t) = 0$ or $y_3(t) = 0$. This means R_3 behaves in the same way as a shift register during scan-out operation when $y_2(t) = 0$ or $y_3(t) = 0$, and hence it is not secure when the attacker regards R_3 as a shift register and tries to observe a present state of R_3. In this way, it may happen that the attacker succeeds in initializing the contents of R_3 and/or observing the contents of R_3, though he/she does not notice them.

To avoid such leakage, we introduced a new secure concept called *strong security* as follows [13]. Consider a circuit C with a single input, a single output, and k-flip-flops. C is called to be *scan-in secure* if for any internal state of C a transfer sequence (of length k) to the state (final state) is always different from that of a k-stage shift register.

C is called to be *scan-out secure* if any present state (initial state) of C can be identified from the output sequence that is always different from that of a k-stage shift register. C is called to be *strongly secure* if C is scan-in secure and scan-out secure.

In [13], we presented a method for making a given GF^{2}SR strongly secure. Here, we consider GFSR and present how to make a given GFSR strongly secure.

Consider a GFSR C with input x, output z, and k flip-flops $y_1, y_2, \ldots, y_k$, such that the most left XOR gate is located between y_p and y_{p+1} as shown in Fig. 7 (a) and the most right XOR gate is located between y_{q-1} and y_q as shown in Fig. 7 (b). If there is no flip-flop between a primary input x and the most left XOR gate, we need to add a dummy flip-flop. As illustrated in Fig. 7 (a), if there is at least one NOT gate between a primary input x and flip-flop y_p, the final state of $(y_1, y_2, \ldots, y_k)$ of C is always different from that of a shift register. Hence, we can see C is scan-in secure. Similarly, as illustrated in Fig. 7 (b), if there is at least one NOT gate between flip-flop y_q and a primary output z, the output sequence of C is always different from that of a shift register, and hence C is scan-out secure.

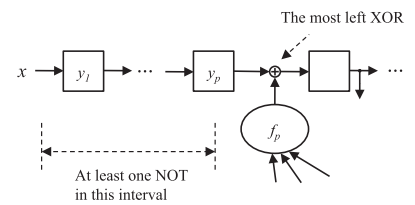
Method for making scan-in secure:

(1) If there is no flip-flop between a primary input and the most left XOR, add a dummy flip-flop between them.
(2) If there is no NOT gate between a primary input x and flip-flop y_p (see Fig. 7 (a)), insert at least one NOT gate between them.

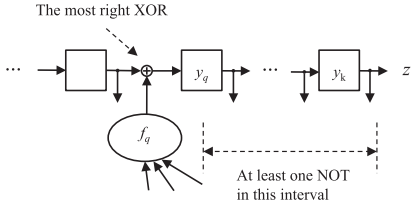
Method for making scan-out secure:

(1) If there is no NOT gate between flip-flop y_q and a primary output z (see Fig. 7 (b)), insert at least one NOT gate between them.

As mentioned at the beginning of this section, GFSR, R_3, shown in Fig. 2 (b) is neither scan-in secure nor scan-out secure. So, we apply both methods for making R_3 scan-in secure and scan-out secure. R_4 shown in Fig. 2 (c) is a result by inserting two NOT gates. It is obvious that the modified circuit is scan-in secure and scan-out secure and hence strongly secure.



(a) Design for scan-in secure GFSR



(b) Design for scan-out secure GFSR

Fig. 7 Design for strongly secure GFSR.

7. Conclusion

In our previous work, we reported a secure and testable scan design approach by using extended shift registers called *SR-equivalents* [10] and *SR-quasi-equivalents* [11], and *generalized feed-forward shift registers* (GF^{2}SR) [12]. In this paper, we introduced another class of generalized shift registers called *generalized feedback shift registers* (GFSR), and considered the properties of GFSR that are useful for secure scan design. We presented how to control/observe GFSR to guarantee scan-in and scan-out operations that can be overlapped in the same way as the conventional scan testing. Testability and security of scan design using GFSR were considered. The cardinality of each class was clarified. We also presented how to design *strongly secure* GFSR as well as GF^{2}SR considered in [13].

References

[1] H. Fujiwara, Logic Testing and Design for Testability, The MIT Press, 1985.
[2] K. Hafner, H.C. Ritter, T.M. Schwair, S. Wallstab, M. Deppermann, J. Gessner, S. Koesters, W.-D. Moeller, and G. Sandweg, "Design and test of an integrated cryptochip," IEEE Des. Test Comput., vol.8, no.4, pp.6–17, Dec. 1991.
[3] D. Hély, F. Bancel, M.-L. Flottes, and B. Rouzeyre, "Securing scan control in crypto chips," Journal of Electronic Testing, vol.23, no.5, pp.457–464, Oct. 2007.
[4] B. Yang, K. Wu, and R. Karri, "Secure scan: A design-for-test architecture for crypto chips," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.25, no.10, pp.2287–2293, 2006.
[5] J. Lee, M. Tehranipoor, C. Patel, and J. Plusquellic, "Securing designs against scan-based side-channel attacks," IEEE Trans. Dependable and Secure Computing, vol.4, no.4, pp.325–336, 2007.
[6] S. Paul, R.S. Chakraborty, and S. Bhunia, "VIm-Scan: A low overhead scan design approach for protection of secret key in scan-based secure chips," Proc. 25th IEEE VLSI Test Symposium, pp.455–460, 2007.
[7] G. Sengar, D. Mukhopadhyay, and D.R. Chowdhury, "Secured flipped scan-chain model for crypto-architecture," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.26, no.11, pp.2080–2084, Nov. 2007.
[8] U. Chandran, and D. Zhao, "SS-KTC: A high-testability low-overhead scan architecture with multi-level security integration," Proc. 27th IEEE VLSI Test Symposium, pp.321–326, May 2009.
[9] M.A. Razzaq, V. Singh, and A. Singh, "SSTKR: Secure and testable scan design through test key randomization," Proc. 20th IEEE Asian Test Symposium, pp.60–65, Nov. 2011.
[10] H. Fujiwara, and M.E.J. Obien, "Secure and testable scan design using extended de Brujin graph," Proc. 15th Asia and South Pacific Design Automation Conference, pp.413–418, Jan. 2010.
[11] K. Fujiwara, H. Fujiwara, and H. Tamamoto, "Secure and testable scan design utilizing shift register quasi-equivalents," IPSJ Trans. System LSI Design Methodology, vol.6, pp.27–33, Feb. 2013.
[12] K. Fujiwara and H. Fujiwara, "Generalized feed forward shift registers and their application to secure scan design," IEICE Trans. Inf. & Syst. vol.E96-D, no.5, pp.1125–1133, May 2013.
[13] H. Fujiwara and K. Fujiwara, "Strongly secure scan design using generalized feed forward shift registers," IEICE Trans. Inf. & Syst., vol.E98-D, no.10, pp.1852–1855, Oct. 2015.