

Chapitre 7

Conception en Vue du Test :

Test Intégré (BIST)

Abdelhakim Khouas

Département de Génie Électrique
École Polytechnique de Montréal

Plan

- C'est quoi le test intégré (BIST) ?
- Méthodes et techniques de génération des vecteurs de test
- LFSR « Linear Feedback Shift Register »
- Techniques d'analyse des réponses
- MISR « Multiple Input Signature Analyser »
- BILBO « Built In Logic Block Observer »

C'est quoi le test intégré (BIST) ?

☞ Test intégré « Built In Self Test » BIST

- Le principe du BIST consiste à ajouter de la circuiterie pour permettre au circuit ou au système de se tester tout seul (Auto-Test).
- Méthode utilisée pour :
 - ↪ Les mémoires
 - ↪ Les circuits analogiques
 - ↪ Les circuits combinatoires et séquentiels
 - ↪ Les systèmes on chip « SOC »

☞ Architecture générale du BIST

- Générateur de vecteurs de test pseudo-aléatoire « PRPG »
- Analyseur de réponses
- Partie contrôle

C'est quoi le test intégré ? (suite)

☞ Avantages :

- Génération interne des vecteurs de test, amélioration de contrôlabilité
- Analyse interne des réponses, amélioration d'observabilité
- Test on-line et off-line
- Équipements de test simples ou pas d'équipement de test
- Test à la fréquence d'utilisation
 - ❖ Plus rapide
 - ❖ Plus précis
 - ❖ Permet le test réel des délais
- Hiérarchisation des tests
- Réutilisation des structures de test

C'est quoi le test intégré ? (suite)

☞ Inconvénients :

- Augmentation de la surface
 - ☞ Mémoire de 4096 mots de 128 bits de l'ordre de 13%
- Dégradation des performances
 - ☞ ROM de 256 Kbits de l'ordre de 6%
- Ajout de pins d'entrée/sortie

C'est quoi le test intégré ? (suite)

☞ Il existe deux types de BIST

➤ Off-line BIST

☞ Le test est effectué pendant le mode test

☞ Deux différents tests :

❖ Off-line fonctionnel : Pour le diagnostic

❖ Off-line structurel : Détection des défauts de fabrication

➤ On-line BIST

☞ Le test s'effectue durant le fonctionnement normal du circuit

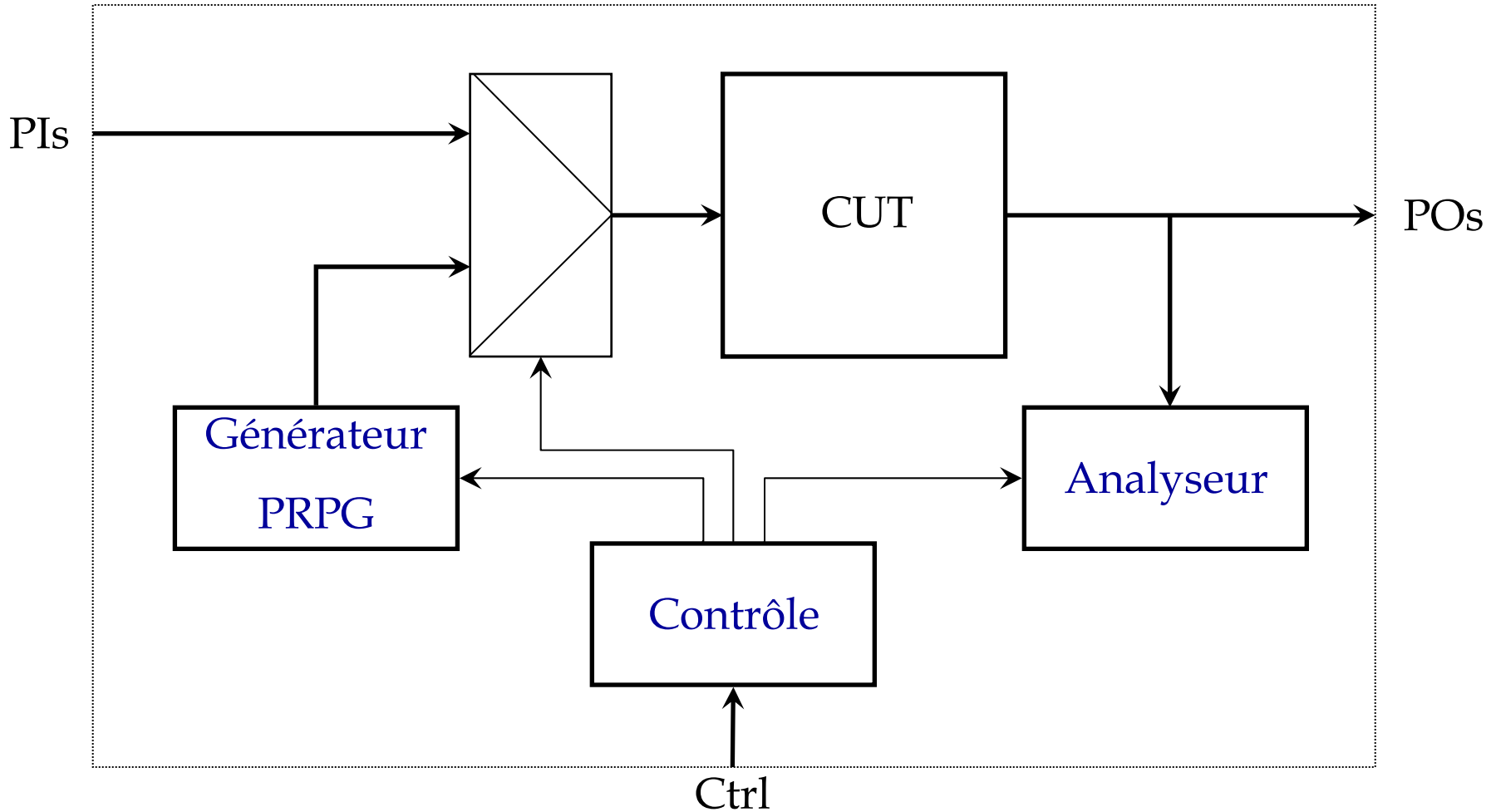
☞ Détecte les défauts pendant le fonctionnement normal du circuit

☞ Deux différents tests :

❖ On-line concurrent

❖ On-line non concurrent

Architecture générale du BIST



Méthodes de génération

☞ Méthodes de génération :

- Génération exhaustive/ pseudo-exhaustive
- Génération aléatoire/ pseudo-aléatoire
- Génération déterministe
- Génération fonctionnelle

☞ Techniques d'application des vecteurs

- Manuel
- ATE « Automatic Test Equipment »
- Interne (cas du BIST)

Techniques de génération pour le BIST

- ☞ Il existe plusieurs techniques de génération et d'application des vecteurs de test pour le BIST
 - Registres à décalage « LFSR »
 - ↳ Pour la génération aléatoire et exhaustive
 - Compteurs binaires
 - ↳ Pour la génération exhaustive
 - ROMs « Prestored TPG »
 - ↳ Pour la génération déterministe ou fonctionnelle
 - Techniques adaptatives
 - ↳ Pour la génération aléatoire pondérée dynamiquement
 - Automates cellulaires
 - ↳ Pour la génération aléatoire

« Linear Feedback Shift Register » LFSR

☞ Les registres à décalage à rebouclage linéaire « LFSR » sont des circuits logiques contenant les composants suivants :

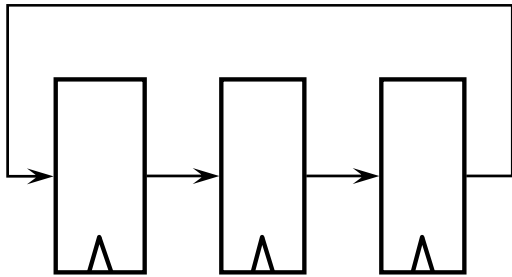
- 1- Bascules D « D Flip-Flops »
- 2- Additionneurs modulo-2 « XOR »

☞ Propriétés du LFSR :

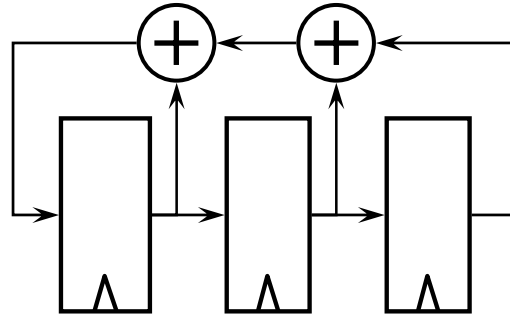
- Cyclique
 - ☞ Le circuit revient à son état initial pour un nombre suffisant de coups d'horloge
- Période
 - ☞ La période du LFSR est comprise entre 1 et $2^n - 1$, n est le nombre de bascules D (longueur du LFSR ou nombre d'étages)

Exemples de registres à décalage

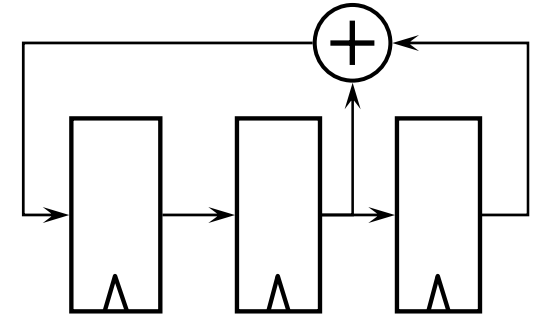
Cyclic SR



Feedback Shift Register



Maximal-Length FSR



S0 :	0	1	1
S1 :	1	0	1
S2 :	1	1	0

S0 :	0	1	1

S0 :	0	1	1
S1 :	0	0	1
S2 :	1	0	0
S3 :	1	1	0

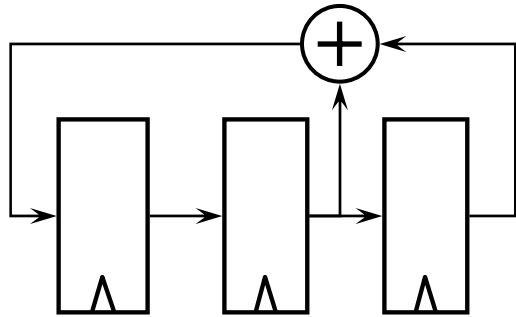
S0 :	0	1	1

S0 :	0	1	1
S1 :	0	0	1
S2 :	1	0	0
S3 :	0	1	0
S4 :	1	0	1
S5 :	1	1	0
S6 :	1	1	1

S0 :	0	1	1

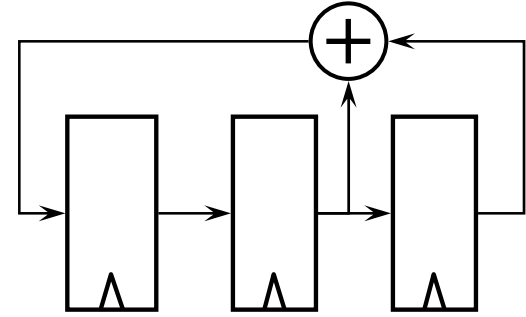
Exemples de registres à décalage (suite)

Maximal-Length FSR



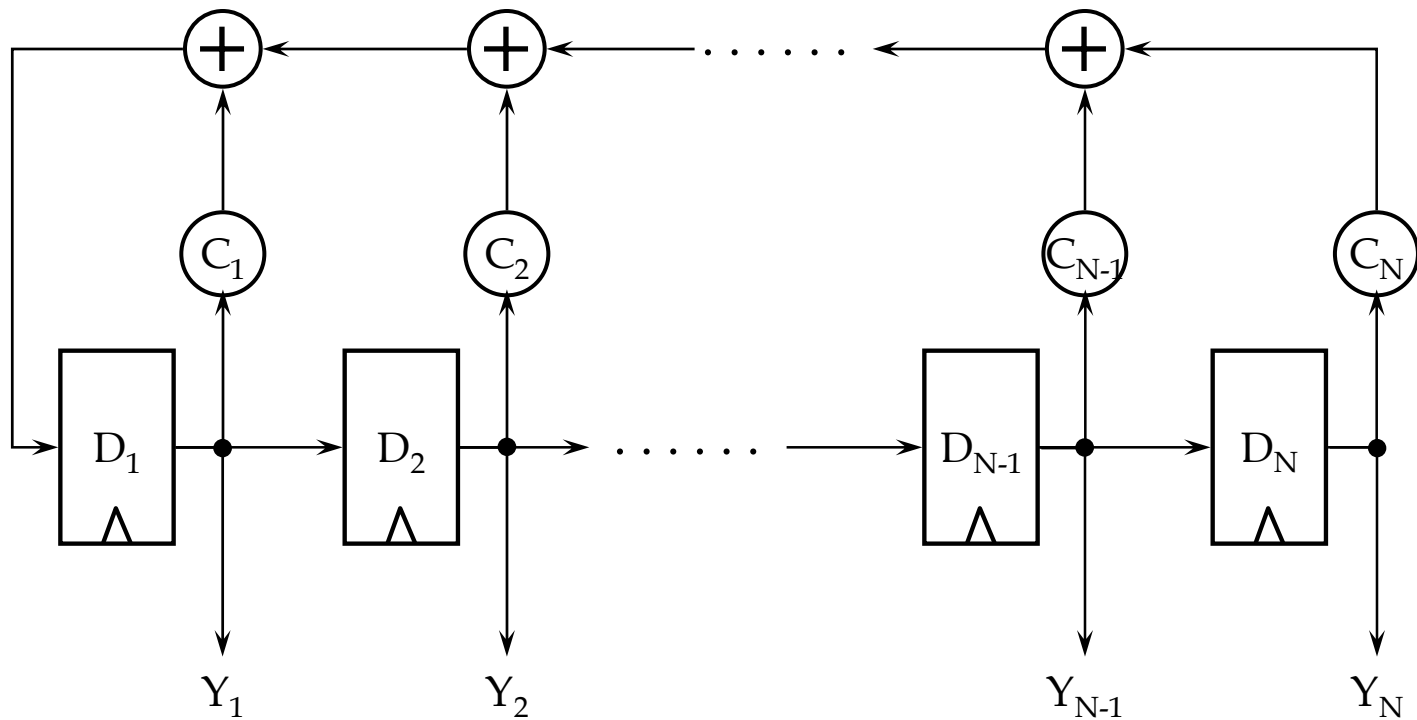
S0 :	1	1	1	} Cyclique de longueur $2^3 - 1$		
S1 :	0	1	1			
S2 :	0	0	1			
S3 :	1	0	0			
S4 :	0	1	0			
S5 :	1	0	1			
S6 :	1	1	0			
S0 :				1	1	1

Maximal-Length FSR



S0 :	0	0	0			
S0 :				0	0	0

Architecture d'un LFSR standard



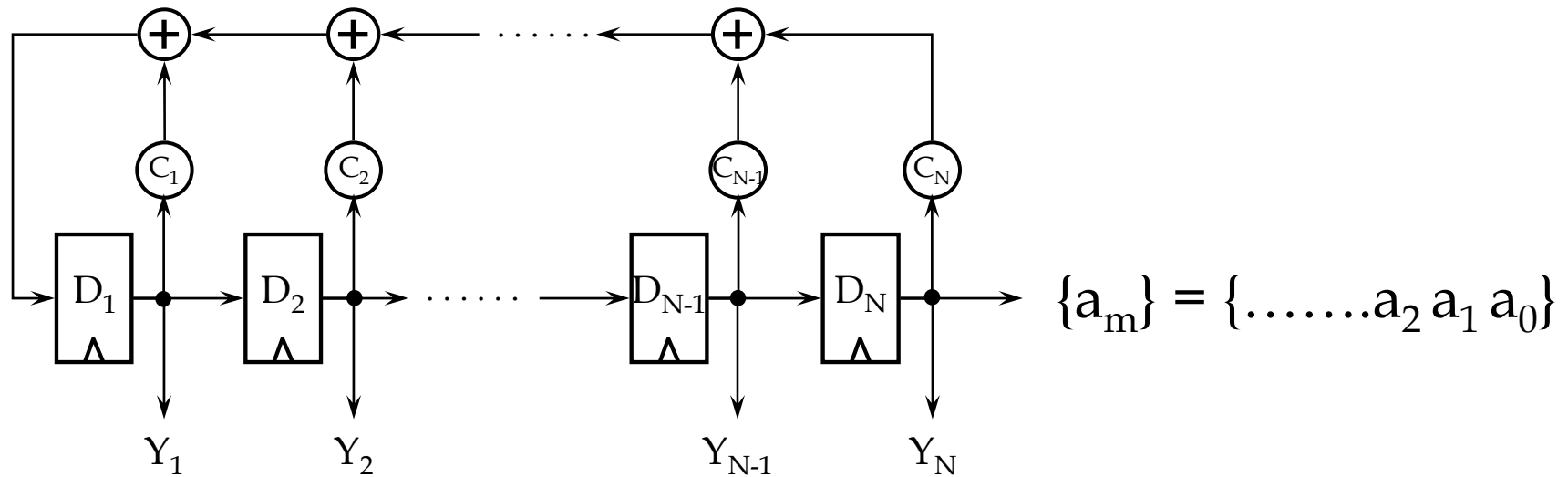
Architecture générale d'un LFSR standard

Architecture d'un LFSR standard (suite)

$$\begin{bmatrix} Y_1(t+1) \\ Y_2(t+1) \\ Y_3(t+1) \\ \cdot \\ \cdot \\ \cdot \\ Y_{n-1}(t+1) \\ Y_n(t+1) \end{bmatrix} = \begin{bmatrix} C_1 & C_2 & C_3 & C_4 & \dots & C_{n-2} & C_{n-1} & 1 \\ 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} Y_1(t) \\ Y_2(t) \\ Y_3(t) \\ \cdot \\ \cdot \\ \cdot \\ Y_{n-1}(t) \\ Y_n(t) \end{bmatrix}$$

Équation matricielle d'un LFSR standard

Fonction génératrice



☞ Pour un LFSR d'ordre n :

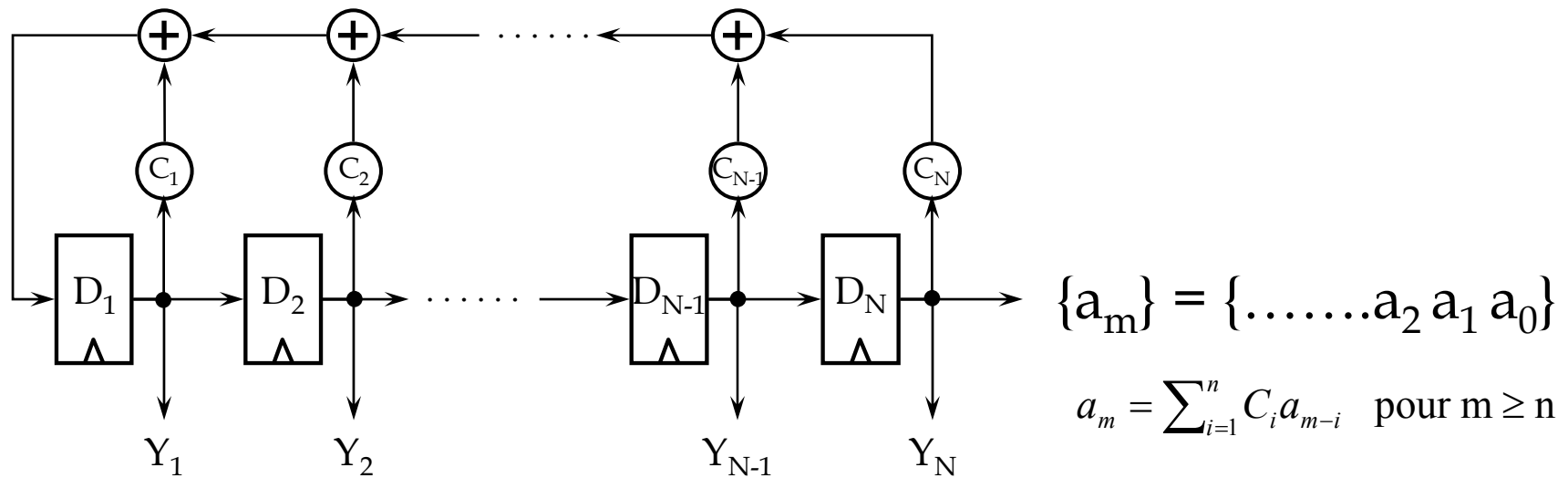
$a_0, a_1, a_2, \dots, a_{n-1}$: sont les valeurs initiales des registres

$$a_n = C_1 \cdot a_{n-1} \oplus C_2 \cdot a_{n-2} \oplus \dots \oplus C_n \cdot a_0$$

⋮

$$a_m = \sum_{i=1}^n C_i a_{m-i} \quad \text{pour } m \geq n$$

Fonction génératrice (suite)

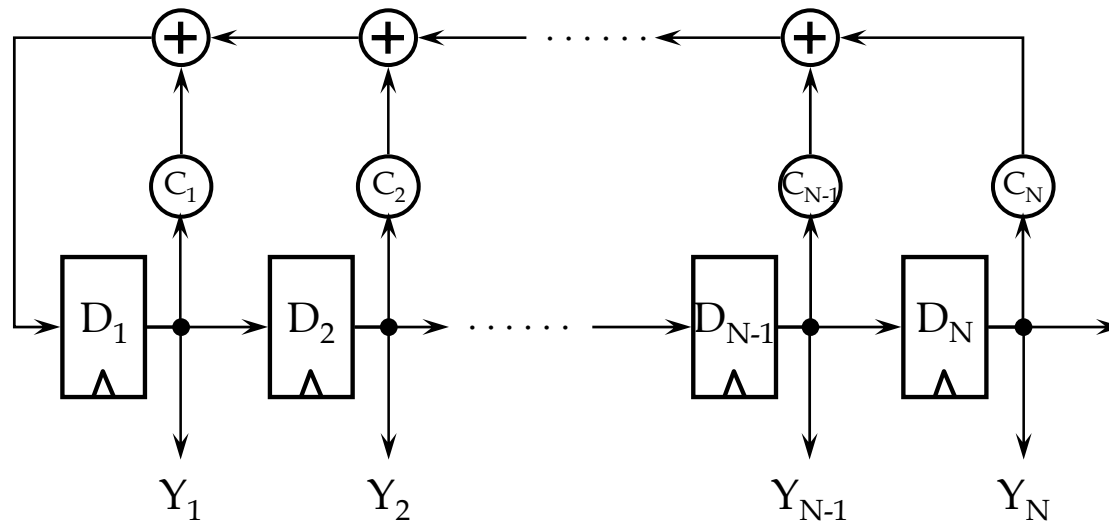


👉 On appelle fonction génératrice du LFSR :

$$G(X) = a_0 \oplus a_1 X \oplus a_2 X^2 \oplus \dots \oplus a_m X^m \oplus \dots$$

$$= \sum_{m=0}^{\infty} a_m X^m \quad (\text{dans la suite on utilisera } + \text{ au lieu de } \oplus)$$

Polynôme caractéristique

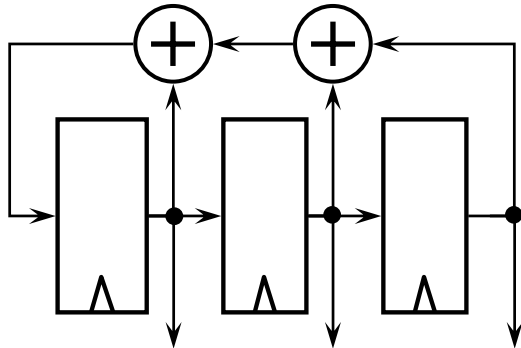


➡ On appelle polynôme caractéristique du LFSR

$$P(X) = 1 + \sum_{i=1}^n C_i X^i$$
$$= 1 + C_1 X + C_2 X^2 + \dots + C_{n-1} X^{n-1} + C_n X^n$$

Exemples de LFSR

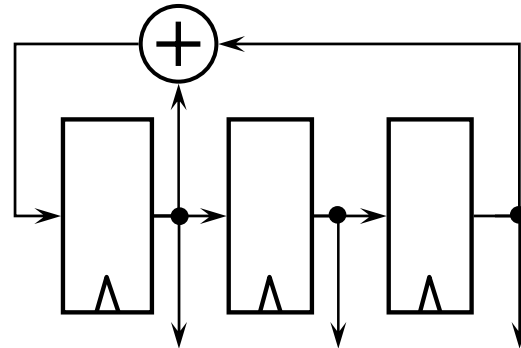
$$P(X) = 1 + X + X^2 + X^3$$



S0 :	0	1	1
S1 :	0	0	1
S2 :	1	0	0
S3 :	1	1	0

S0 :	0	1	1

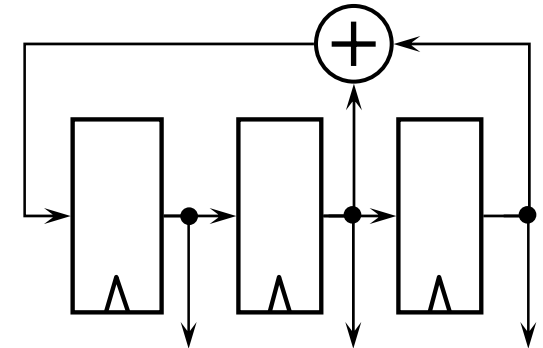
$$P(X) = 1 + X + X^3$$



S0 :	0	1	1
S1 :	1	0	1
S2 :	0	1	0
S3 :	0	0	1
S4 :	1	0	0
S5 :	1	1	0
S6 :	1	1	1

S0 :	0	1	1

$$P(X) = 1 + X^2 + X^3$$



S0 :	0	1	1
S1 :	0	0	1
S2 :	1	0	0
S3 :	0	1	0
S4 :	1	0	1
S5 :	1	1	0
S6 :	1	1	1

S0 :	0	1	1

Théorie des LFSRs

☞ On peut montrer que :

$$G(X) = \frac{\sum_{i=1}^n C_i X^i (a_{-i} X^{-i} + \dots + a_{-1} X^{-1})}{1 + \sum_{i=1}^n C_i X^i}$$

avec : $a_{-1}, a_{-2}, \dots, a_{-n}$ les valeurs initiales des registres

☞ Pour $a_{-1} = a_{-2} = \dots = a_{-n-1} = 0$ et $a_{-n} = 1$, on :

$$G(X) = \frac{1}{1 + \sum_{i=1}^n C_i X^i} = \frac{1}{P(X)}$$

Théorie des LFSRs (suite)

☞ Comme $G(X)$ est périodique de période p , alors on a :

$$G(X) = A(X) + X^p A(X) + \dots + X^{pi} A(X) + \dots$$

$$= A(X) \sum_{i=0}^{\infty} X^{pi} = \frac{A(X)}{1 - X^p}$$

$$\text{Comme } G(X) = \frac{1}{P(X)} \Rightarrow 1 - X^p = A(X)P(X)$$

☞ Si p est la période du LFSR, alors le polynôme caractéristique $P(X)$ divise $(1 - X^p)$

☞ Théorème :

- La période d'un LFSR de polynôme caractéristique $P(X)$ est le plus petit entier k tel que $P(X)$ divise $(1 - X^k)$

Polynômes primitifs

☞ Définition : Polynôme primitif

- C'est un polynôme caractéristique qui permet d'avoir la période maximum qui est $2^n - 1$

☞ Définition : Polynôme irréductible

- C'est un polynôme caractéristique qui est aussi premier, il n'est divisible que par 1 et lui-même, il est caractérisé par :
 - ❖ Il a un nombre impair de termes
 - ❖ S'il est de degré n , alors il divise $(1+X^k)$ avec $k=2^n-1$

☞ Théorème

- Un polynôme irréductible $P(X)$ est primitif si le plus petit entier positif k tel que $P(X)$ est un diviseur de $(1+X^k)$ est la valeur $k=(2^n-1)$ avec n le degré du polynôme $P(X)$

Polynômes primitifs (suite)

Valeurs de N	Polynômes primitifs
1,2,3,4,6,7,15,22	$1 + X + X^n$
5,11, 21, 29	$1 + X^2 + X^n$
10,17,20,25,28,31	$1 + X^3 + X^n$
9	$1 + X^4 + X^n$
23	$1 + X^5 + X^n$
18	$1 + X^7 + X^n$
8	$1 + X^2 + X^3 + X^4 + X^n$
12,14,16	$1 + X + X^3 + X^4 + X^n$
13	$1 + X + X^4 + X^6 + X^n$

Polynômes primitifs (suite)

☞ Les polynômes de degré 3 sont :

➤ $P(X)=1+X^3$

➤ $P(X)=1+X+X^3$ primitif

➤ $P(X)=1+X^2+X^3$ primitif

➤ $P(X)=1+X+X^2+X^3$

❖ $(1+X^3) = (1+X)(1+X+X^2)$ et $(1+X+X^2+X^3) = (1+X)(1+X^2)$

☞ Il existe plusieurs polynômes primitifs de degré n , mais il est avantageux d'utiliser celui qui a le moins de termes car il nécessite moins de portes xor.

Propriétés des générateurs LFSR

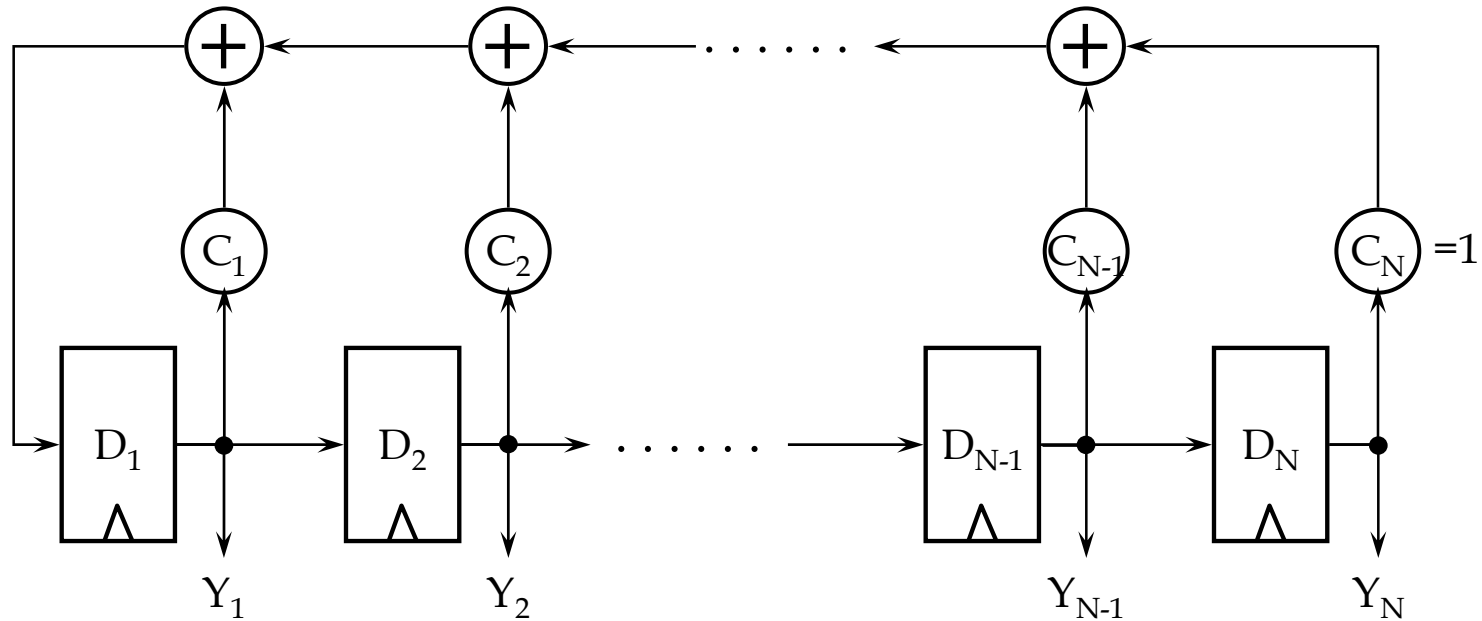
☞ La séquence des vecteurs générés par un LFSR ayant un polynôme primitif d'ordre n à les propriétés suivantes :

- ❖ La période de la séquence est $2^n - 1$
- ❖ Si on initialise le LFSR à une valeur différente de 0, le LFSR génère tous les vecteurs possibles (sauf 0) avant de revenir au vecteur initial
- ❖ La séquence de longueur $(2^n - 1)$ a 2^{n-1} 1s et $(2^{n-1} - 1)$ 0s
- ❖ La séquence de longueur $(2^n - 1)$ a $(2^{n-1} - 1)$ transitions

☞ Pour la génération pseudo-aléatoire, le LFSR est l'un des meilleurs générateurs

Types de LFSR

☞ Type 1 : LFSR standard « xor externes »



$$P(X) = 1 + \sum_{i=1}^n C_i X^i = 1 + C_1 X + C_2 X^2 + \dots + C_{n-1} X^{n-1} + C_n X^n$$

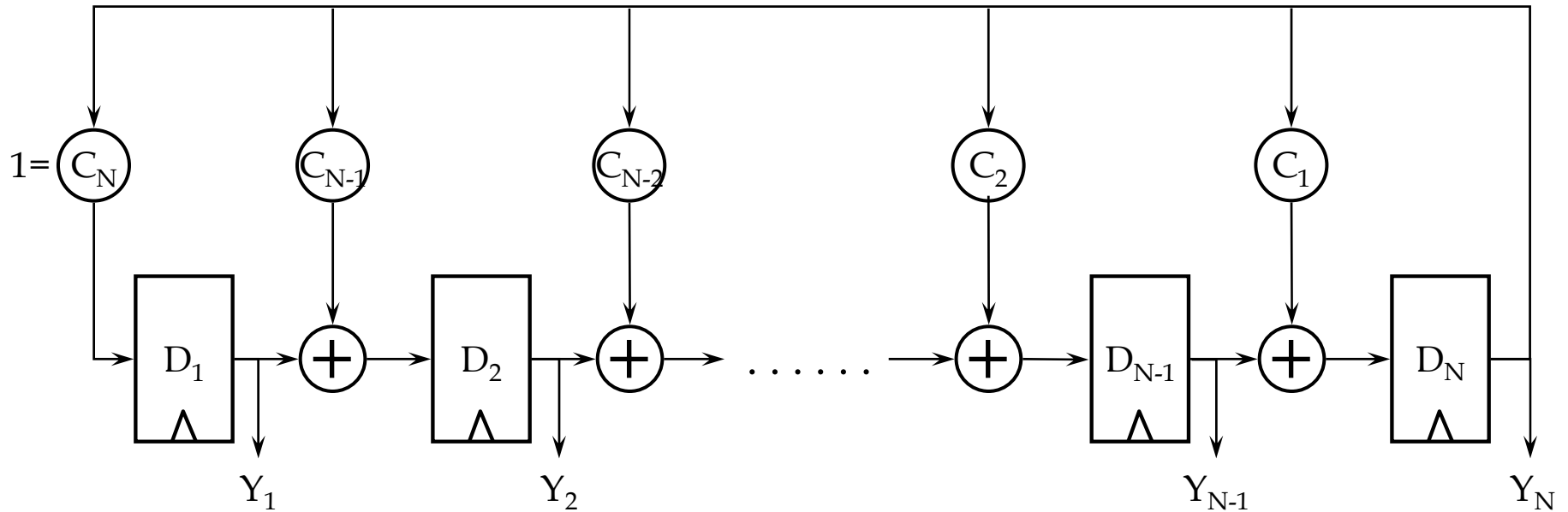
Types de LFSR (suite)

$$\begin{bmatrix} Y_1(t+1) \\ Y_2(t+1) \\ Y_3(t+1) \\ \cdot \\ \cdot \\ \cdot \\ Y_{n-1}(t+1) \\ Y_n(t+1) \end{bmatrix} = \begin{bmatrix} C_1 & C_2 & C_3 & C_4 & \dots & C_{n-2} & C_{n-1} & 1 \\ 1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} Y_1(t) \\ Y_2(t) \\ Y_3(t) \\ \cdot \\ \cdot \\ \cdot \\ Y_{n-1}(t) \\ Y_n(t) \end{bmatrix}$$

Équation matricielle d'un LFSR standard (XOR externe)

Types de LFSR (suite)

☞ Type 2 : LFSR dual « xor internes »



$$P(X) = 1 + C_1X + C_2X^2 + \dots + C_{n-1}X^{n-1} + C_nX^n$$

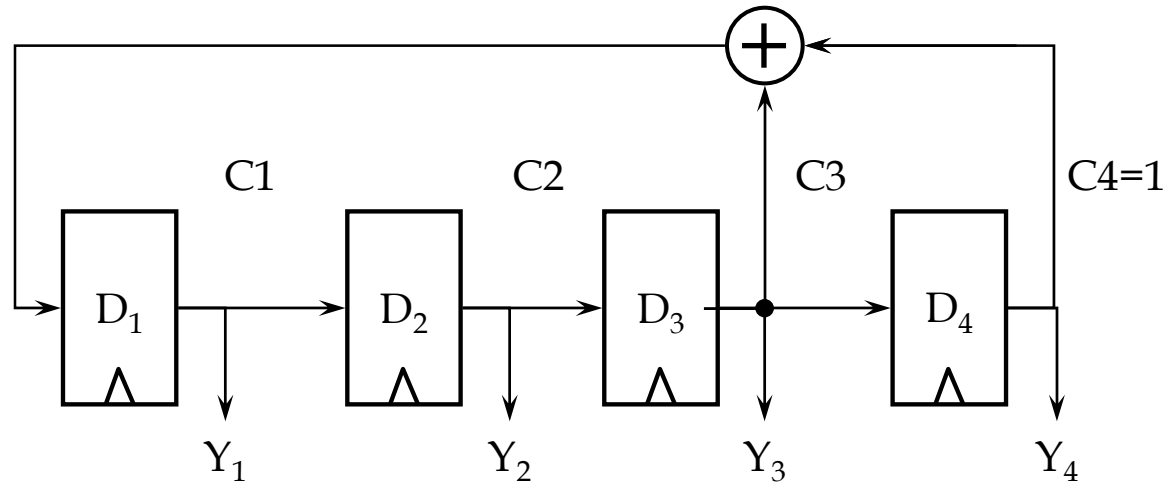
Types de LFSR (suite)

$$\begin{bmatrix} Y_1(t+1) \\ Y_2(t+1) \\ Y_3(t+1) \\ \cdot \\ \cdot \\ \cdot \\ Y_{n-1}(t+1) \\ Y_n(t+1) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & \dots & 0 & 0 & C_{n-2} \\ 0 & 1 & 0 & 0 & \dots & 0 & 0 & C_{n-1} \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 & C_2 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & C_1 \end{bmatrix} \times \begin{bmatrix} Y_1(t) \\ Y_2(t) \\ Y_3(t) \\ \cdot \\ \cdot \\ \cdot \\ Y_{n-1}(t) \\ Y_n(t) \end{bmatrix}$$

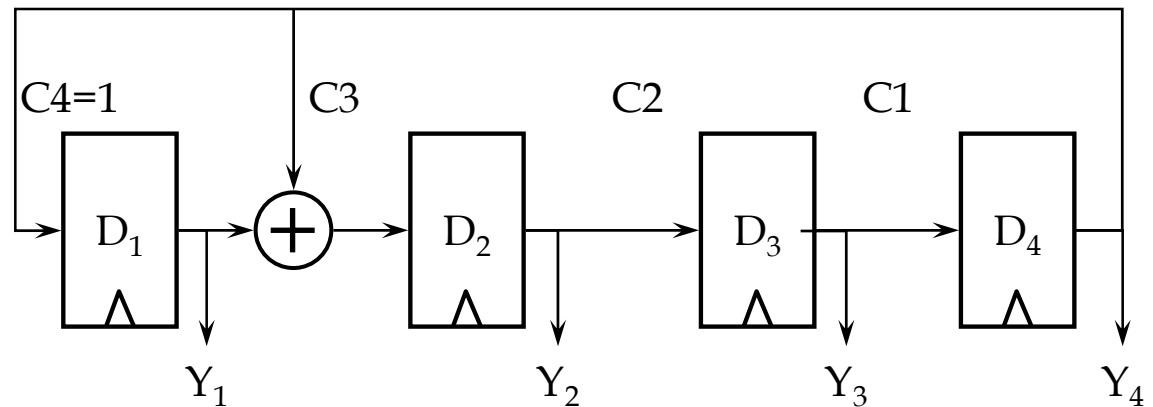
Équation matricielle d'un LFSR dual (XOR interne)

Types de LFSR (suite)

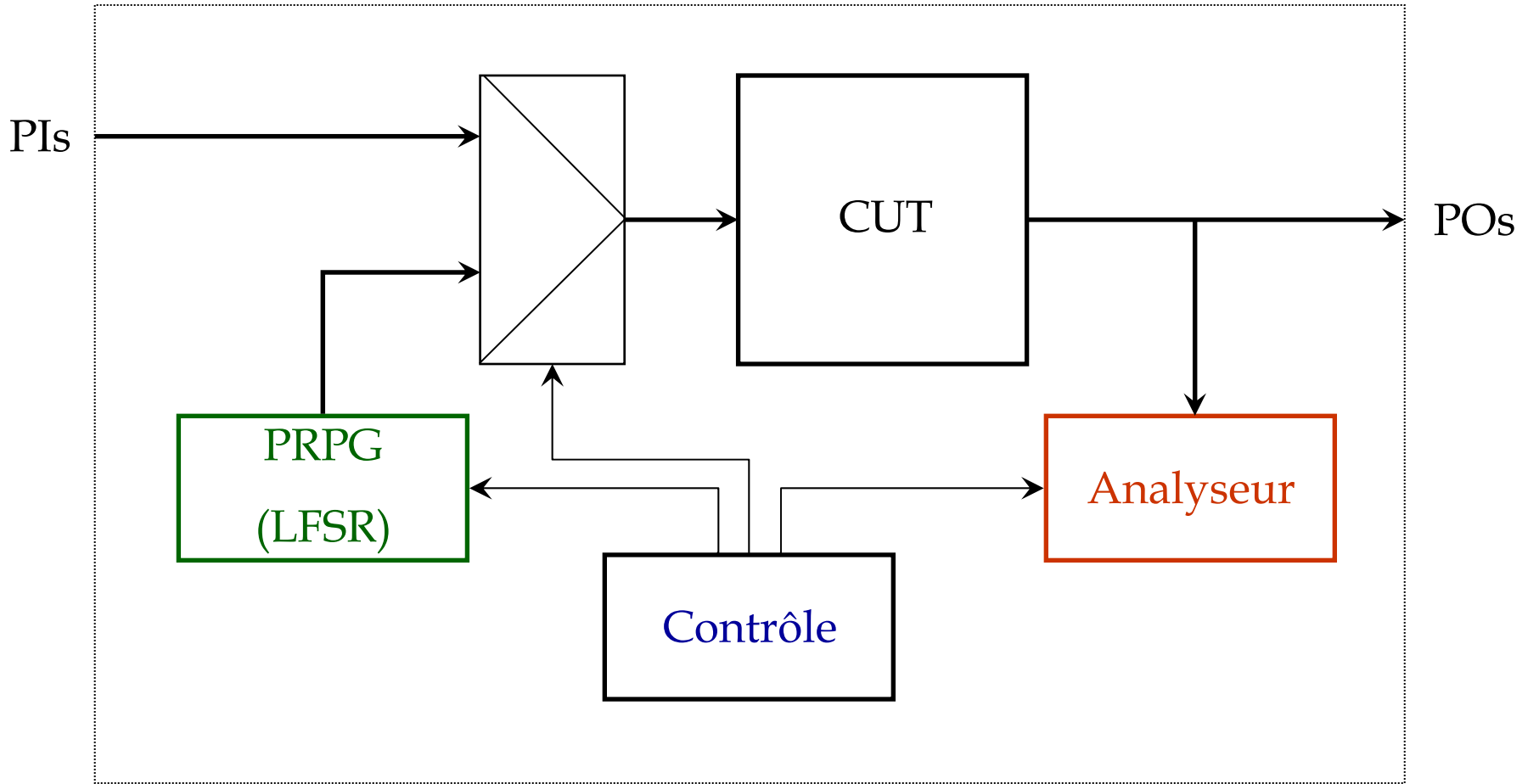
☞ LFSR
standard



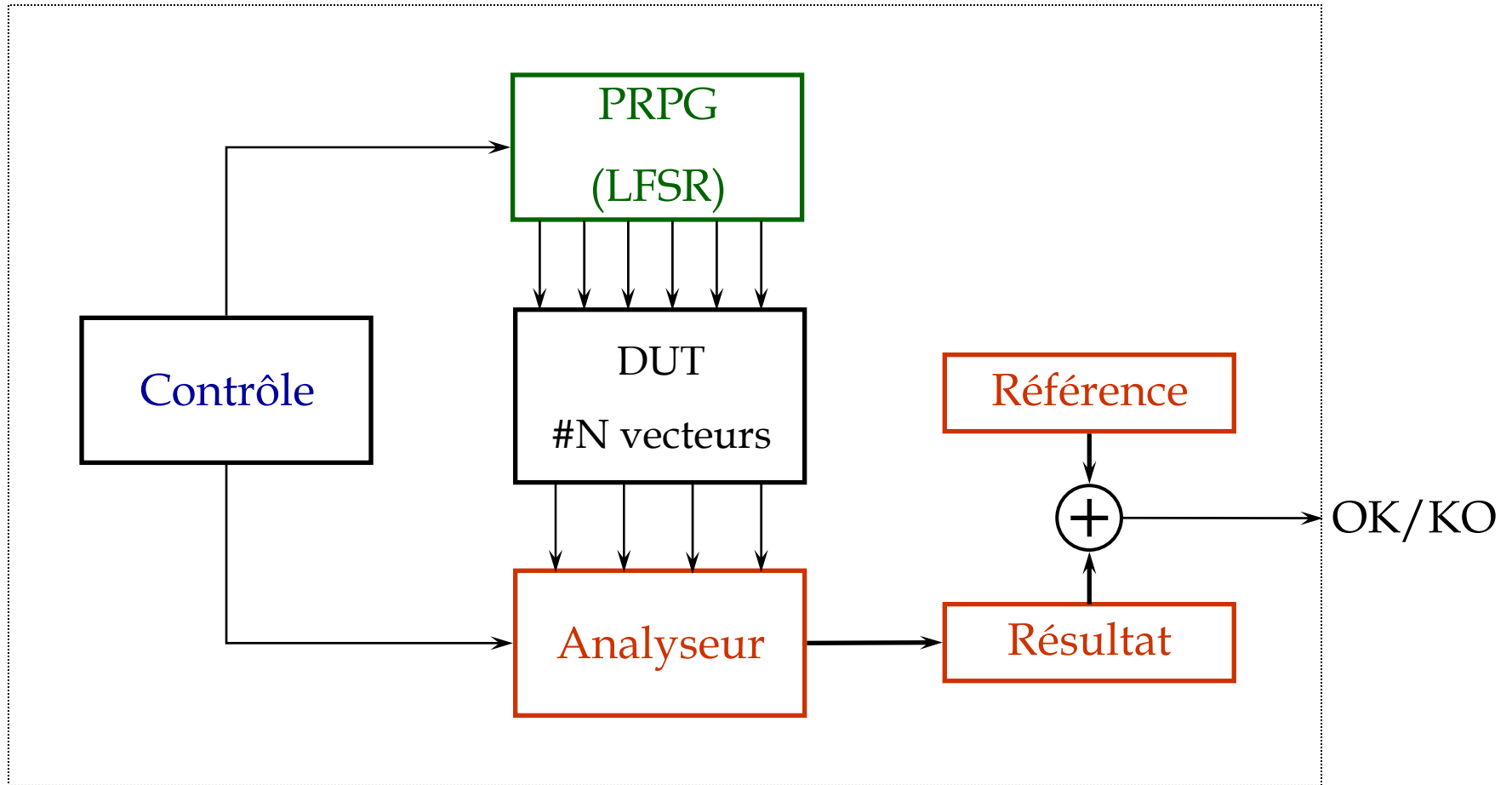
☞ LFSR dual



Architecture générale du BIST



Analyse des réponses



Analyse des réponses (suite)

👉 But

- Trouver un moyen de comparer les réponses du circuit sous test avec les réponses du circuit correct pour pouvoir détecter les circuits défectueux. Les paramètres importants à considérer sont :
 - ❖ L'algorithme utilisé doit être facile à implémenter
 - ❖ Le temps d'analyse des résultats doit être très rapide
 - ❖ Minimiser la longueur du résultat à stocker
 - ❖ Minimiser la perte d'information

👉 Problématique de l'analyse des réponses

- Compression (réversible) et stockage trop coûteux
- Compaction (irréversible) engendre le problème de masquage de fautes

Analyse des réponses (suite)

☞ Masquage de fautes

- Soit $C(m)$ la fonction de compression (compaction) utilisée pour n vecteurs
- On appelle « signature » le résultat de la fonction $C(m)$
- Une faute est masquée si la fonction $C(m)$ donne la même signature pour le circuit fautif et le circuit correct. Conséquences :
 - ❖ Pour le test, il est impossible de détecter la faute malgré que la séquence de test détecte la faute
 - ❖ Pour le diagnostic, il est impossible de différencier deux fautes qui ont la même signature

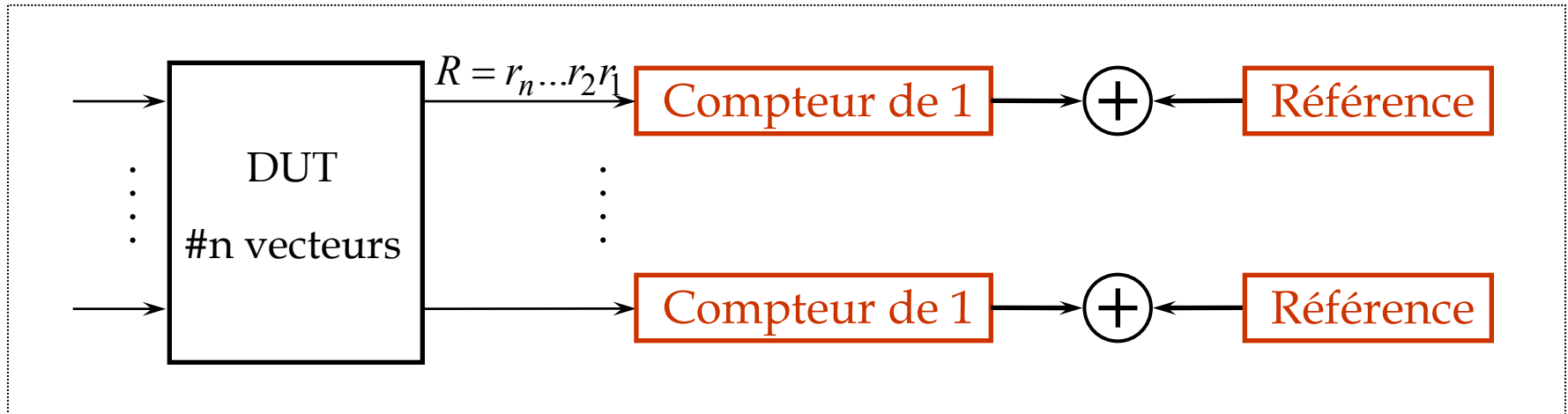
☞ Une des mesures importantes de la qualité d'une technique de compression est la probabilité de masquage de fautes

Analyse des réponses (suite)

☞ Il existe plusieurs techniques d'analyse des réponses qui sont :

- Contage de un « Ones counting »
- Contage de transitions « Transitions counting »
- Vérification de la parité « Parity checking »
- Compression cyclique « Cyclic code compression »
- Analyse de signatures « Signature analysis »
 - ↳ en utilisant un registre LFSR
 - ↳ en utilisant un registre MISR
 - ↳ en utilisant un registre BILBO

Contage de un

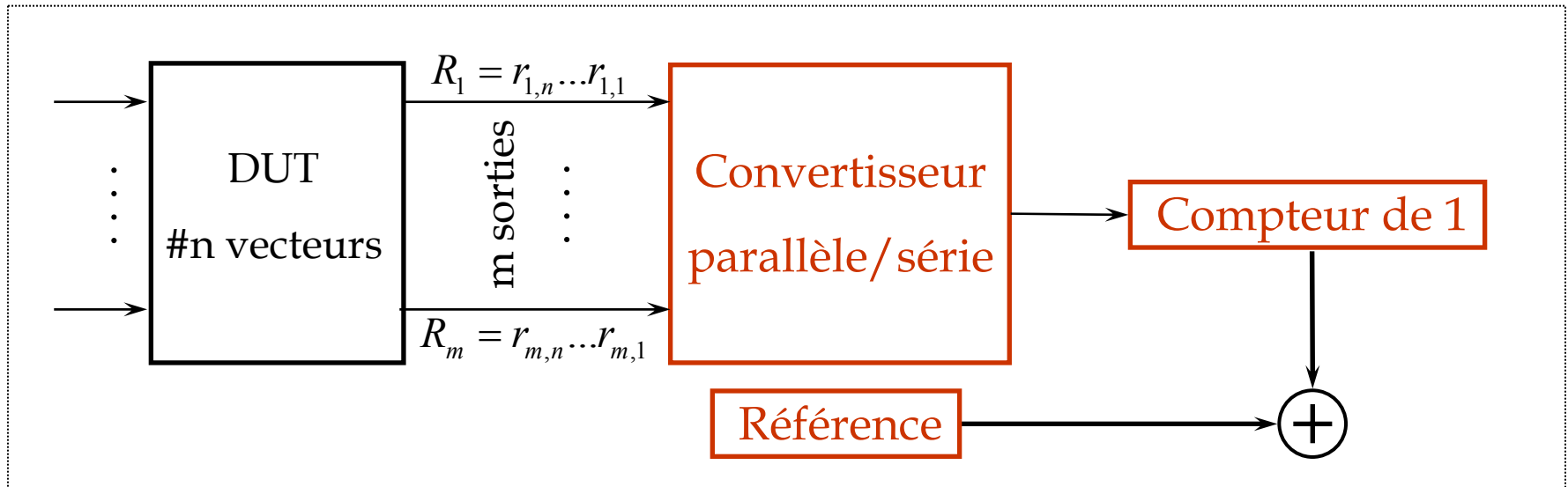


La fonction de compression est : $1C(R) = \sum_{i=1}^n r_i = r$ on a $0 \leq r \leq n$

Le nombre de séquence R' de n bits ayant $1C(R') = r$ est : $\binom{n}{r} = \frac{n!}{r!(n-r)!}$

La probabilité de masquage de fautes est : $P(n/r) = \frac{\binom{n}{r} - 1}{2^n - 1}$

Contage de un (suite)



La fonction de compression est : $1C(R) = \sum_{j=1}^m \sum_{i=1}^n r_{j,i} = r$ on a $0 \leq r \leq n.m$

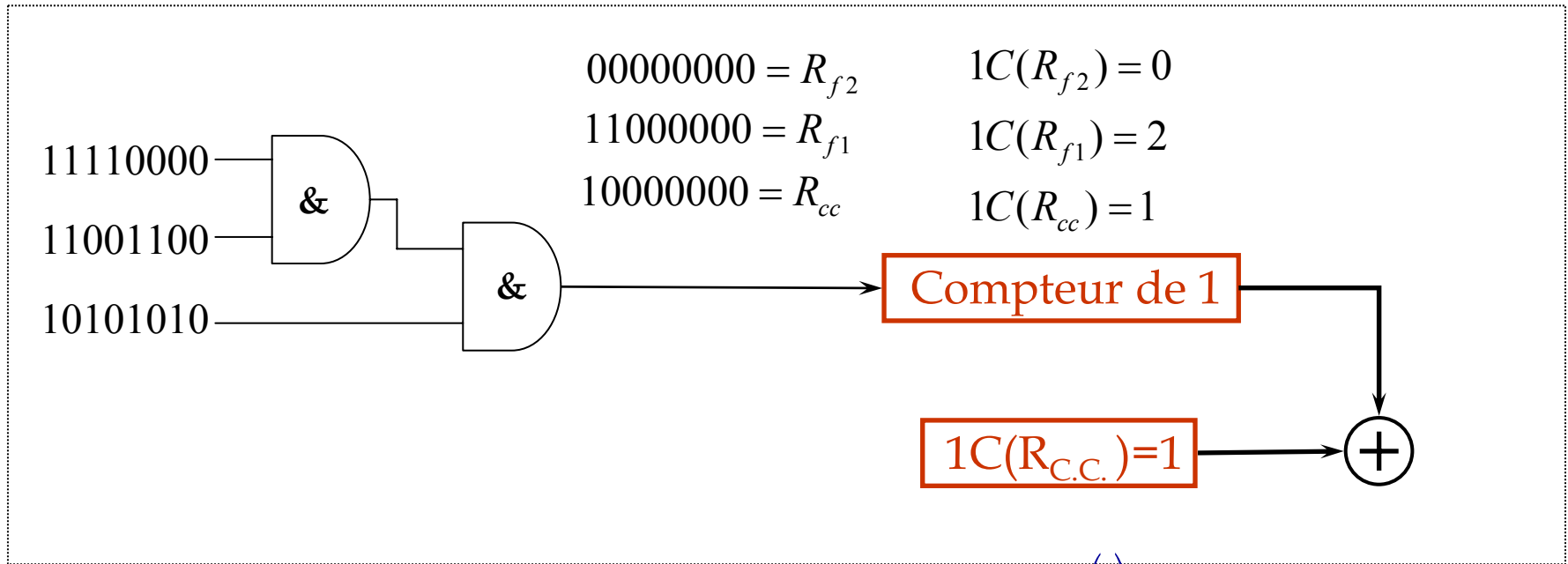
👉 Avantage :

- Diminue la surface additionnelle

👉 Inconvénients :

- Augmente le temps d'application des tests
- Augmente la probabilité de masquage de fautes

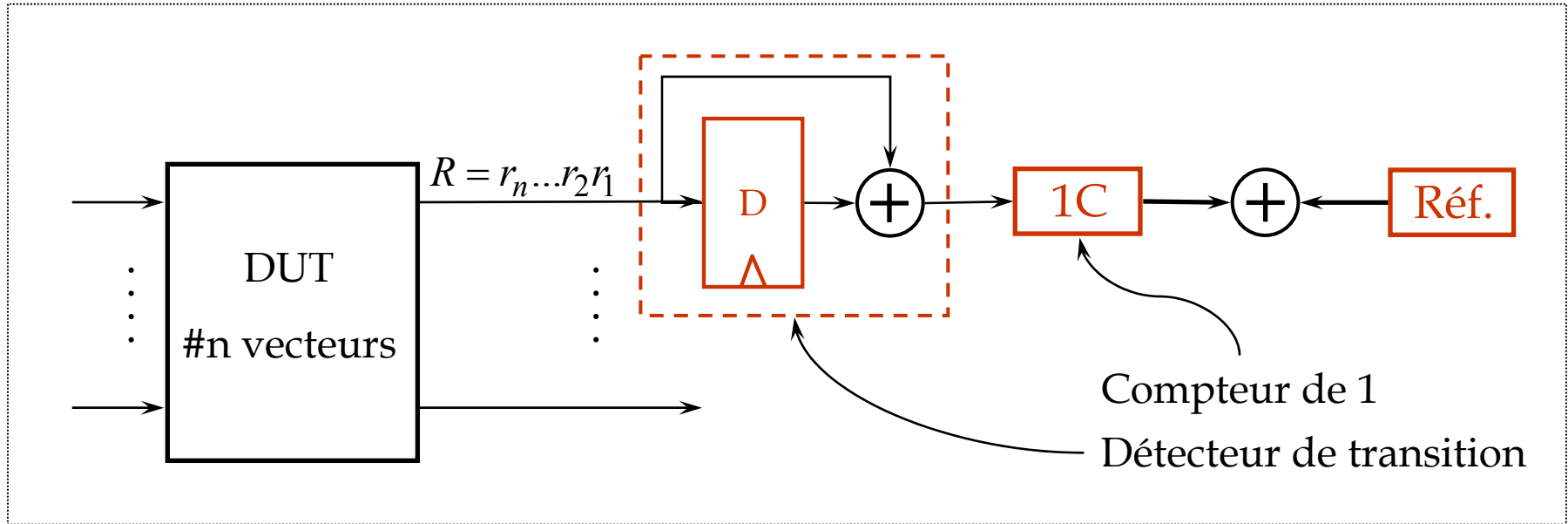
Contage de un (suite)



La probabilité de masquage de fautes est : $P(8/1) = \frac{\binom{8}{1}-1}{2^8-1} = \frac{7}{255} = 0,02745$

☞ On remarque que la probabilité de masquage est plus faible pour les valeurs extrêmes de $1C(R_{C.C.})$. Si $1C(R_{C.C.})=0$ ou 8 alors $P=0$

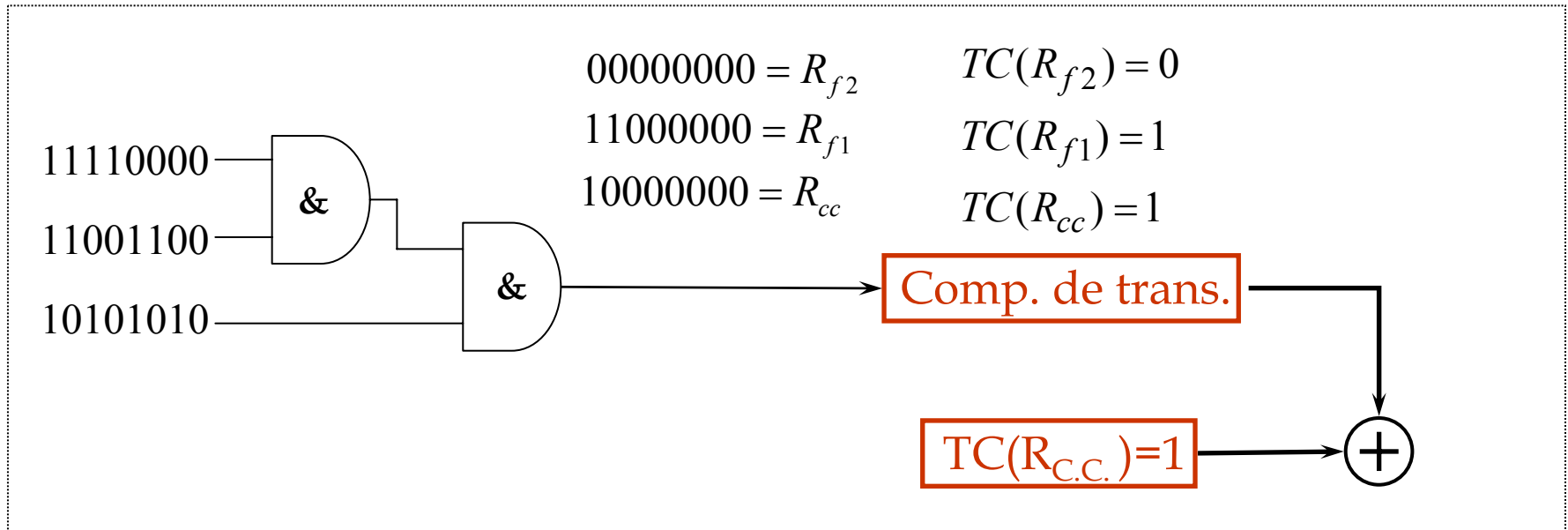
Contage de transitions



La fonction de compression est : $TC(R) = \sum_{i=1}^{n-1} (r_i \oplus r_{i+1}) = r$ on a $0 \leq r \leq (n-1)$

La probabilité de masquage de fautes est : $P(n/r) = \frac{2^{\binom{n-1}{r}} - 1}{2^n - 1}$

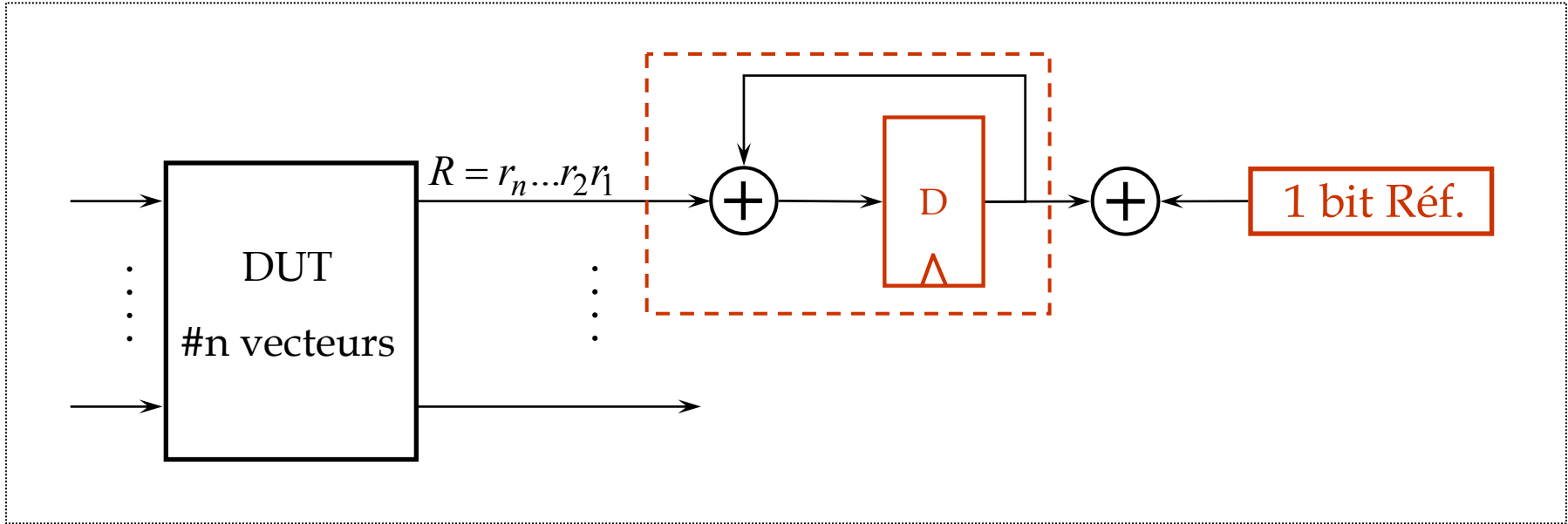
Contage de transitions (suite)



La probabilité de masquage de fautes est : $P(8/1) = \frac{2^{\binom{7}{1}} - 1}{2^8 - 1} = \frac{14}{255} = 0,055$

- ☞ On remarque que, dans ce cas, la probabilité de masquage est plus grande que pour la technique de comptage de 1.
- ☞ On ne peut pas détecter la faute f1

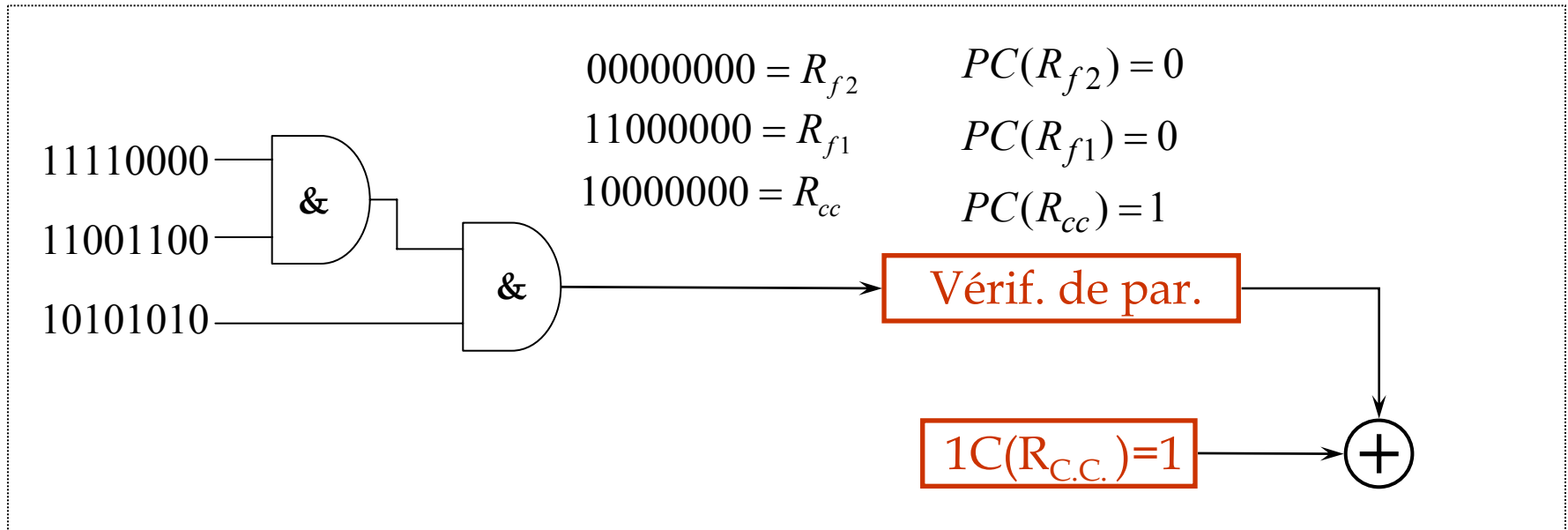
Vérification de la parité



La fonction de compression est : $PC(R) = r_0 \oplus r_1 \oplus \dots \oplus r_i$ $PC(R) = 0$ ou 1

La probabilité de masquage de fautes est : $P(n) = \frac{2^{n-1} - 1}{2^n - 1} \approx 0,5$

Vérification de la parité (suite)



La probabilité de masquage de fautes est : $P(8) = \frac{2^7-1}{2^8-1} = \frac{14}{255} = 0,498$

☞ On remarque que la probabilité de masquage est plus grande que pour la technique de comptage de transition, mais on détecte les deux fautes de l'exemple.

Analyse de signatures

- ☞ L'analyse de signature est une technique de compression basée sur l'utilisation des registres à décalage
 - La forme la plus simple de cette méthode consiste à utiliser un LFSR à une entrée
 - La signature est le contenu du LFSR après avoir entré le dernier bit de la séquence à analyser

Analyse de signatures (suite)

☞ Théorie des polynômes

Soit deux polynômes : $G(X)$ et $P(X)$

La division de $G(X)$ par $P(X)$ produit :

un quotient $Q(X)$

et un reste $R(X)$ on a donc :

$$\frac{G(X)}{P(X)} = Q(X) + \frac{R(X)}{P(X)} \quad \text{ou}$$

$$G(X) = Q(X)P(X) + R(X)$$

La fonction de division par $P(X)$ est réalisé
par un LFSR de polynôme caractéristique $P(X)$

☞ Exemple

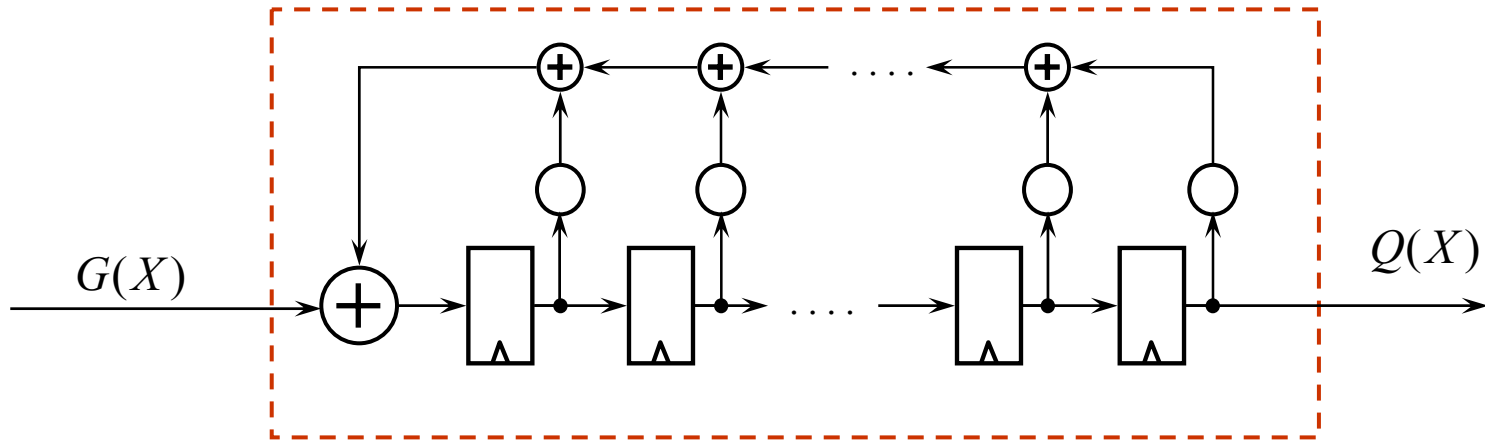
$$G(X) = X^7 + X^3 + X$$

$$P(X) = X^5 + X^3 + X + 1$$

$$\begin{array}{r|l}
 X^7 + X^3 + X & X^5 + X^3 + X + 1 \\
 \oplus X^7 + X^5 + X^3 + X^2 & \text{-----} \\
 \text{-----} & X^2 + 1 \\
 X^5 + X^2 + X & \\
 \oplus X^5 + X^3 + X + 1 & \\
 \text{-----} & \\
 X^3 + X^2 + 1 &
 \end{array}$$

$$Q(X) = X^2 + 1 \quad R(X) = X^3 + X^2 + 1$$

Analyse de signatures (suite)



Soit $P(X)$ le polynôme caractéristique du LFSR

$I(X) = 0$ l'état initial du LFSR

$R(X)$ l'état final du LFSR alors on a :

$$\frac{G(X)}{P(X)} = Q(X) + \frac{R(X)}{P(X)} \quad \text{ou}$$

$$G(X) = Q(X)P(X) + R(X)$$

Analyse de signatures (suite)

Exemple

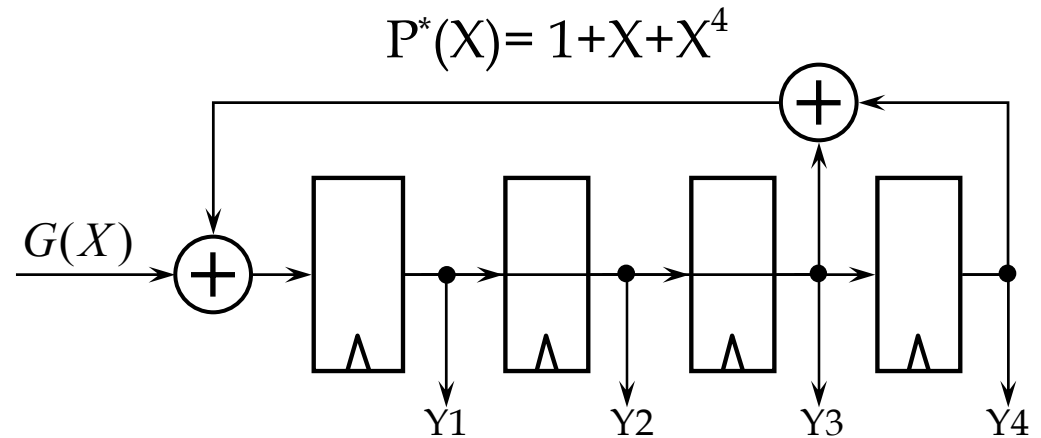
$$G(X) = X^7 + X^3 + X^2 + X \Rightarrow 10001110$$

$$P(X) = X^4 + X + 1$$

$$\begin{array}{r|l}
 X^7 + X^3 + X^2 + X & X^4 + X + 1 \\
 \oplus X^7 + X^4 + X^3 & \hline
 \hline
 X^4 + X^2 + X & X^3 + 1 \\
 \oplus X^4 + X + 1 & \\
 \hline
 X^2 + 1 &
 \end{array}$$

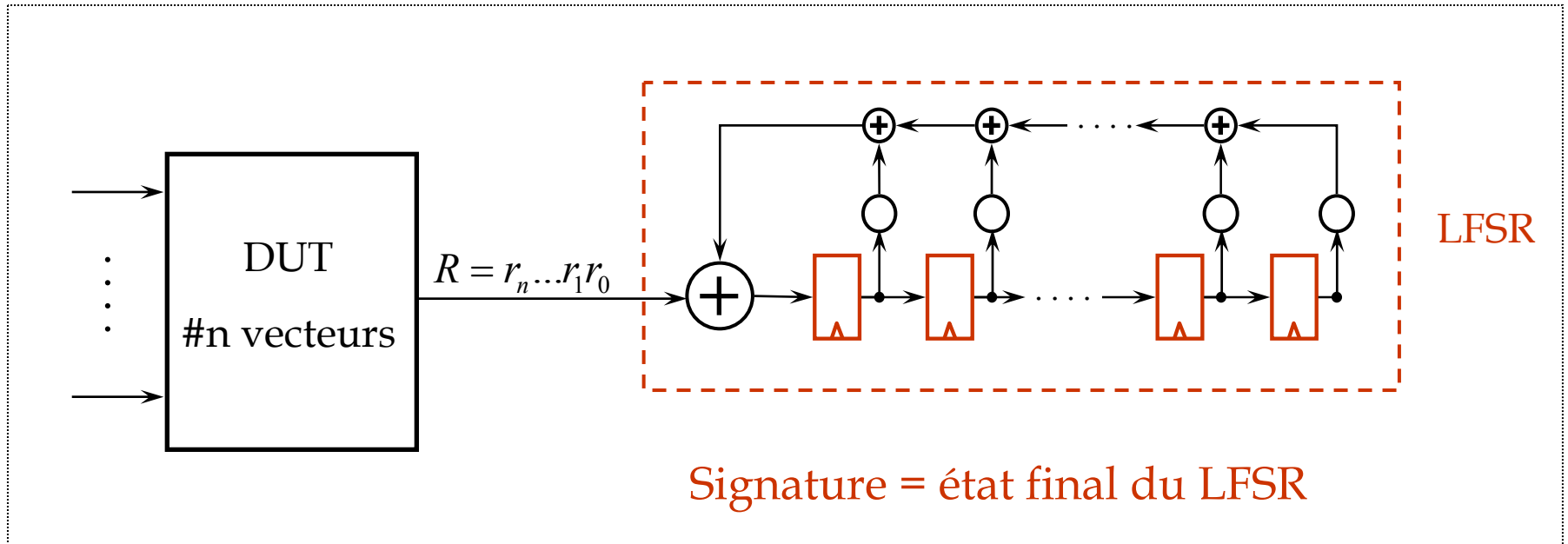
$$Q(X) = X^3 + 1 \Rightarrow 1001$$

$$R(X) = X^2 + 1 \Rightarrow 0101$$



G(X)	Y1	Y2	Y3	Y4	Q(X)
	0	0	0	0	
1	1	0	0	0	
0	0	1	0	0	
0	0	0	1	0	
0	1	0	0	1	
1	0	1	0	0	1
1	1	0	1	0	0
1	0	1	0	1	0
0	1	0	1	0	1

Analyse de signatures avec LFSR



La signature de la séquence R est : $R(X) = \text{état final du registre}$

La probabilité de masquage de fautes est : $P(n) = \frac{2^{n-m} - 1}{2^n - 1} \approx \frac{1}{2^m}$

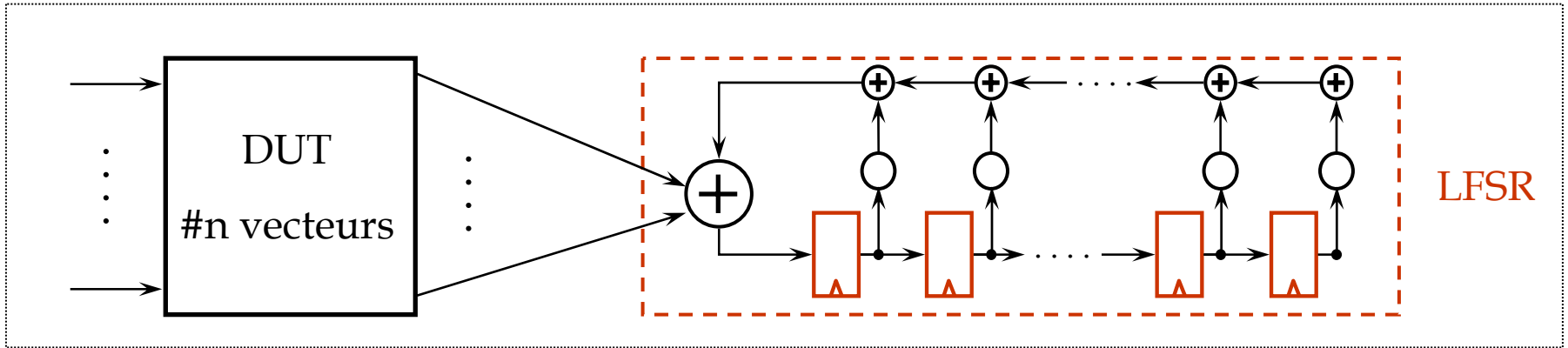
avec $m = \text{longueur du LFSR}$

Analyse de signatures avec LFSR (suite)

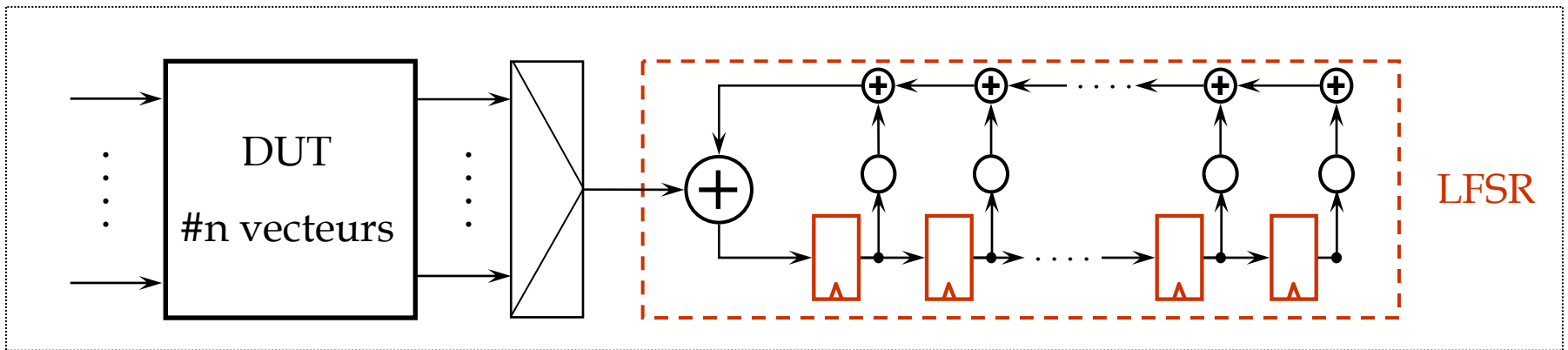
➤ Pour un circuit à plusieurs sorties $R_1 \dots R_m$ il existe plusieurs adaptations de la technique d'analyse de signatures avec un LFSR qui sont :

- 1- LFSR avec un multiplexeur
- 2- LFSR avec un XOR
- 3- LFSR à entrées multiples MISR

Analyse de signature (suite)

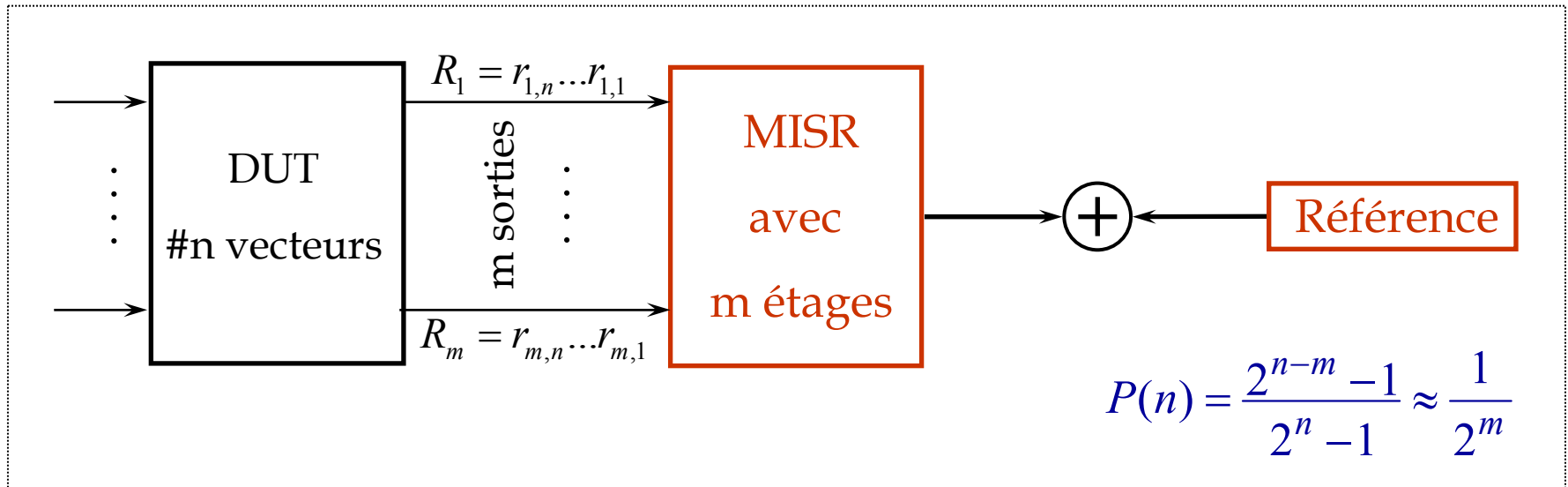


LFSR avec un XOR



LFSR avec un multiplexeur

Analyse de signature avec MISR



👉 Multiple Input Shift Register « MISR »

- On utilise un MISR pour analyser plusieurs sorties à la fois

👉 Avantages

- Diminue la surface additionnelles (par rapport à plusieurs LFSRs)
- Diminue le temps de test (par rapport au LFSR avec multiplexeur)

MISR (suite)

$$\begin{bmatrix} D_1(t+1) \\ D_2(t+1) \\ D_3(t+1) \\ \cdot \\ \cdot \\ \cdot \\ D_{m-1}(t+1) \\ D_m(t+1) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & \dots & 0 & 0 & C_{n-2} \\ 0 & 1 & 0 & 0 & \dots & 0 & 0 & C_{n-1} \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 & C_2 \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 & C_1 \end{bmatrix} \times \begin{bmatrix} D_1(t) \\ D_2(t) \\ D_3(t) \\ \cdot \\ \cdot \\ \cdot \\ D_{m-1}(t) \\ D_m(t) \end{bmatrix} \oplus \begin{bmatrix} R_1(t) \\ R_2(t) \\ R_3(t) \\ \cdot \\ \cdot \\ \cdot \\ R_{m-1}(t) \\ R_m(t) \end{bmatrix}$$

Équation matricielle d'un MISR

Built-In Logic Block Observer « BILBO »

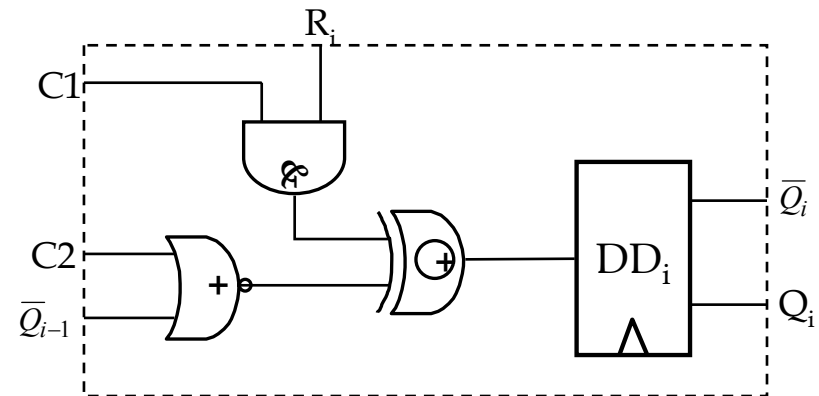
☞ Combine les registres avec scan, le générateur de vecteurs de test et l'analyseur de signature

☞ Quatre modes de fonctionnement

- Normal
- Générateur LFSR et analyseur de signature MISR
- SCAN et Reset

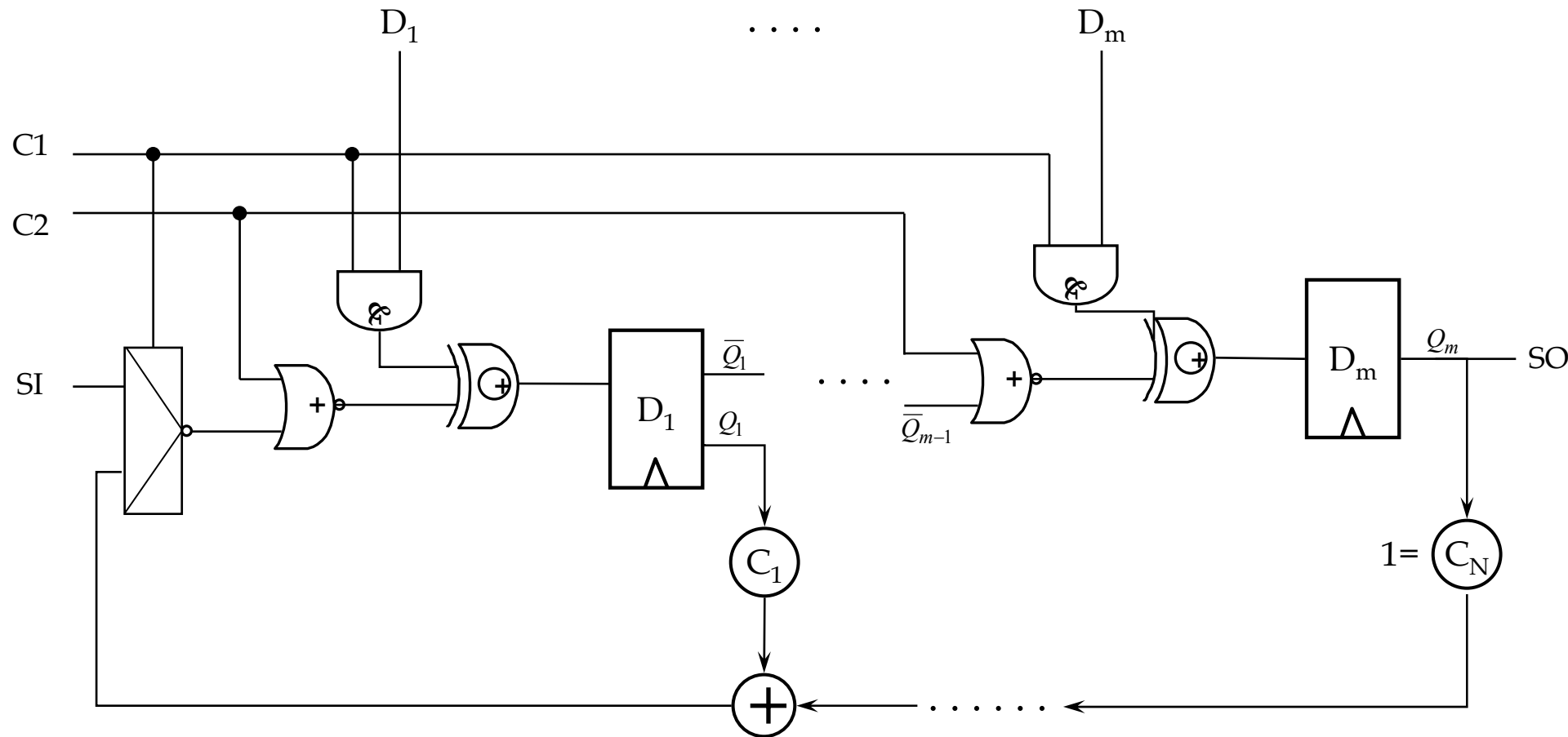
Mode		Fonction
C1	C2	
0	0	SCAN
0	1	Reset
1	0	LFSR/MISR
1	1	Normal

Modes de fonctionnement



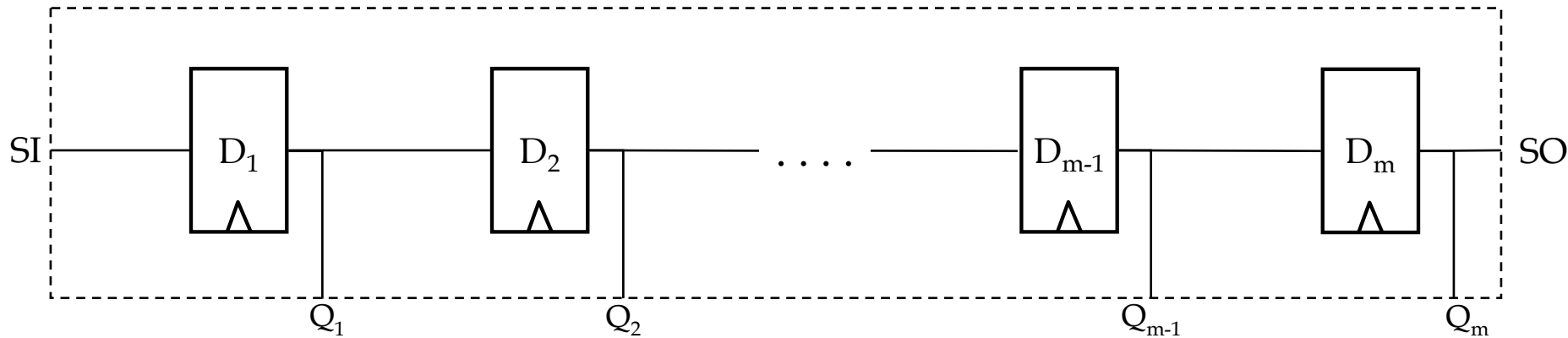
Cellule BILBO

Built-In Logic Block Observer (suite)

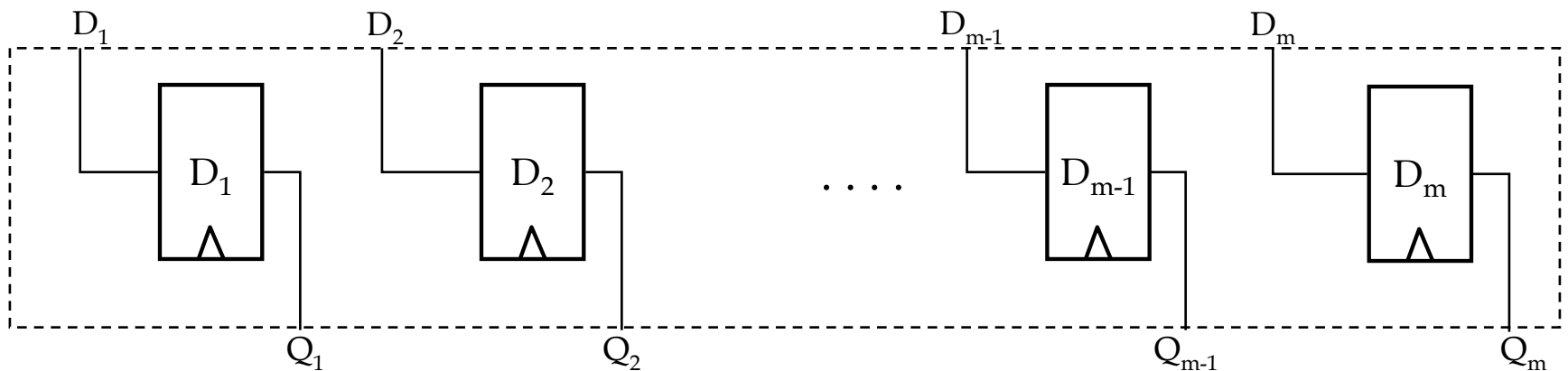


Built-In Logic Block Observer (suite)

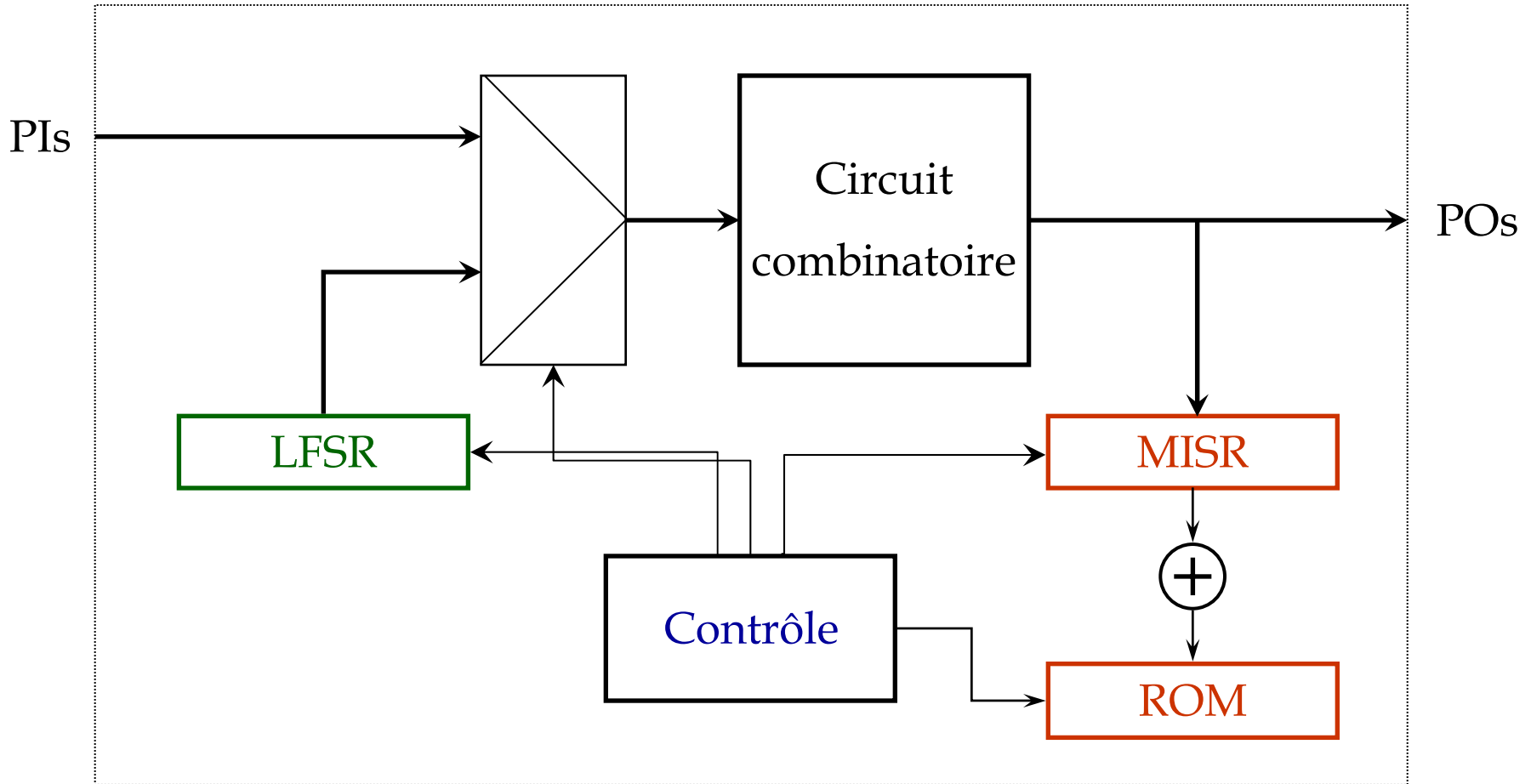
☞ $C1=C2=0$ (Scan)



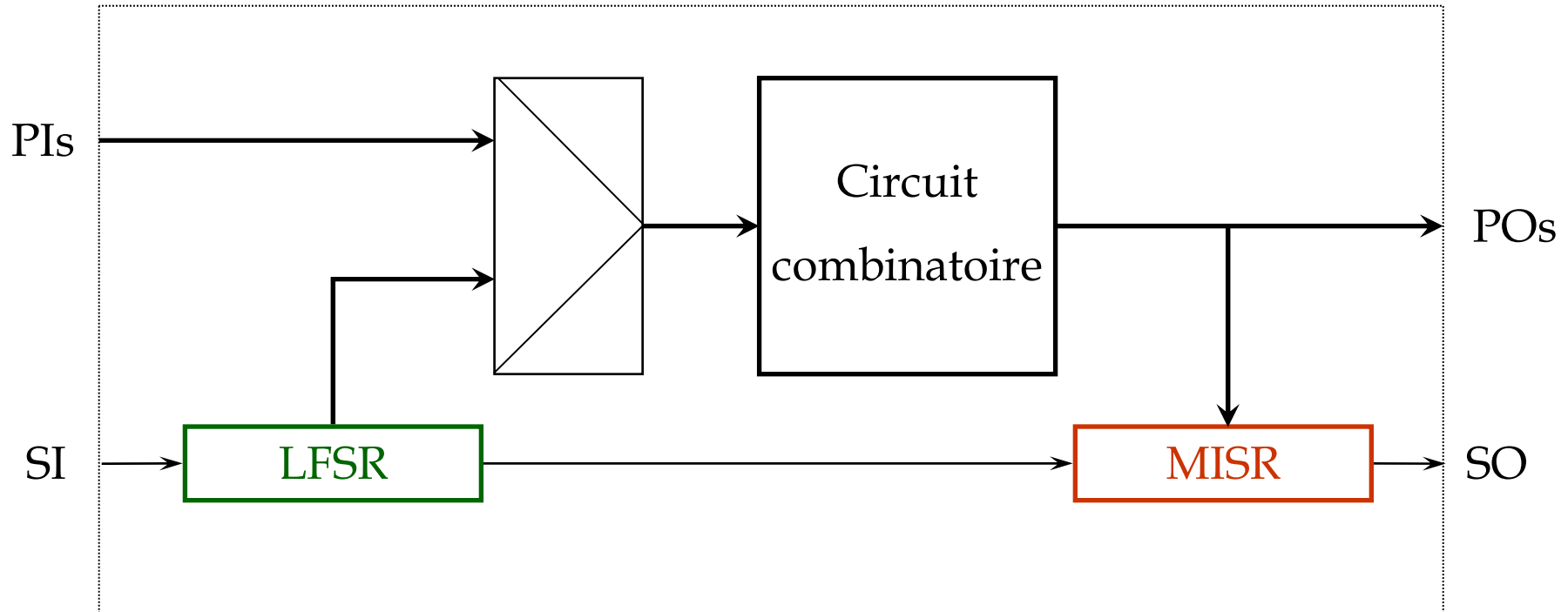
☞ $C1=C2=1$ (normal)



BIST avec LFSR et MISR



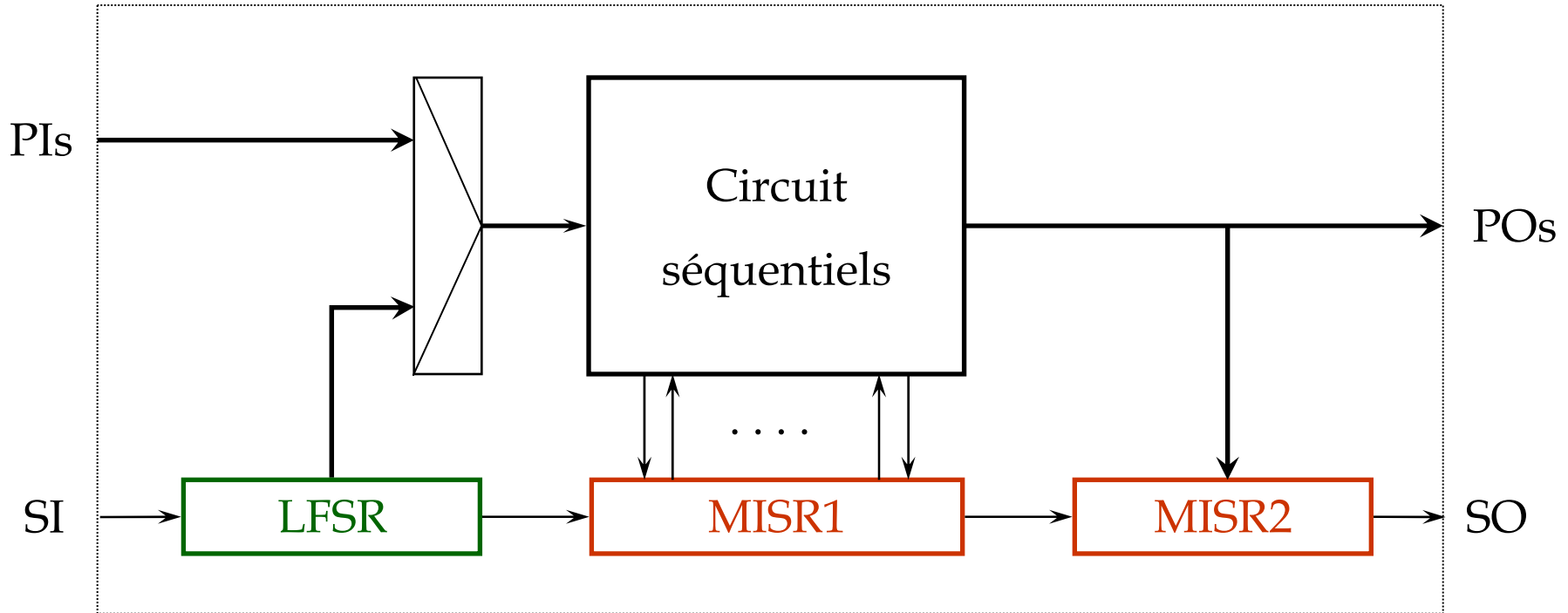
BIST avec LFSR et MISR et Scan-Path



Test intégré avec LFSR, MISR et Scan-Path

- ☞ On utilise le mode scan pour initialiser le LFSR et lire la signature finale du circuit

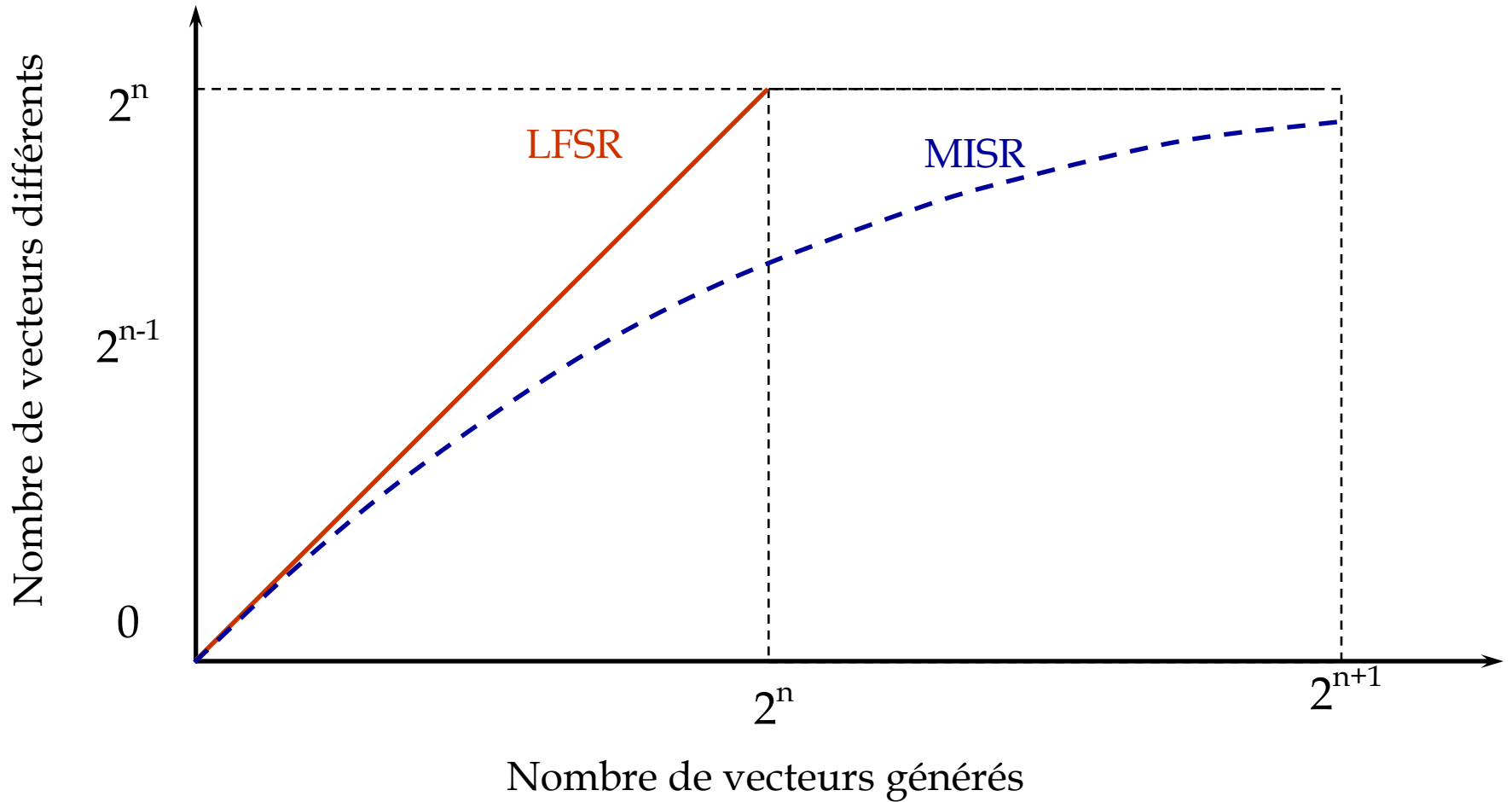
BIST pour les circuits séquentiels



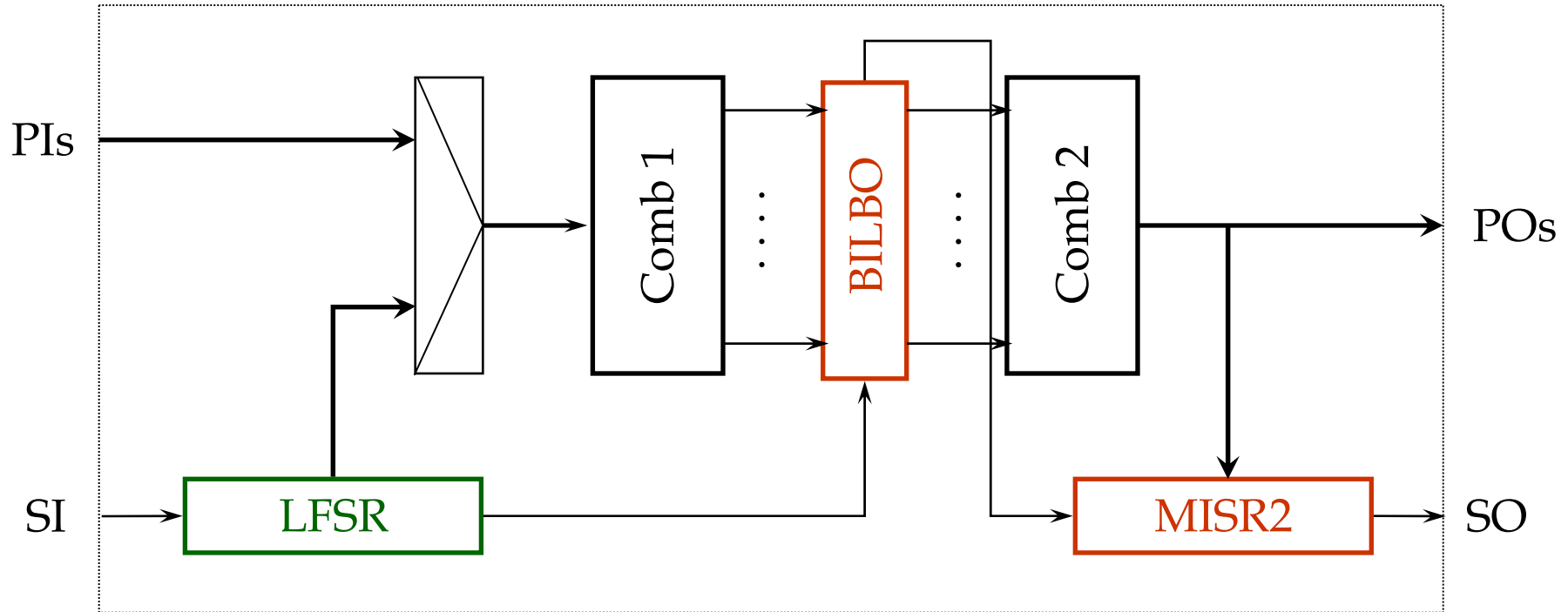
Test intégré avec LFSR et MISR pour les circuits séquentiels

- Le MISR1 est utilisé au même temps comme analyseur de signature et PRPG (générateur de vecteur de test pseudo-aléatoire)

MISR comme PRPG



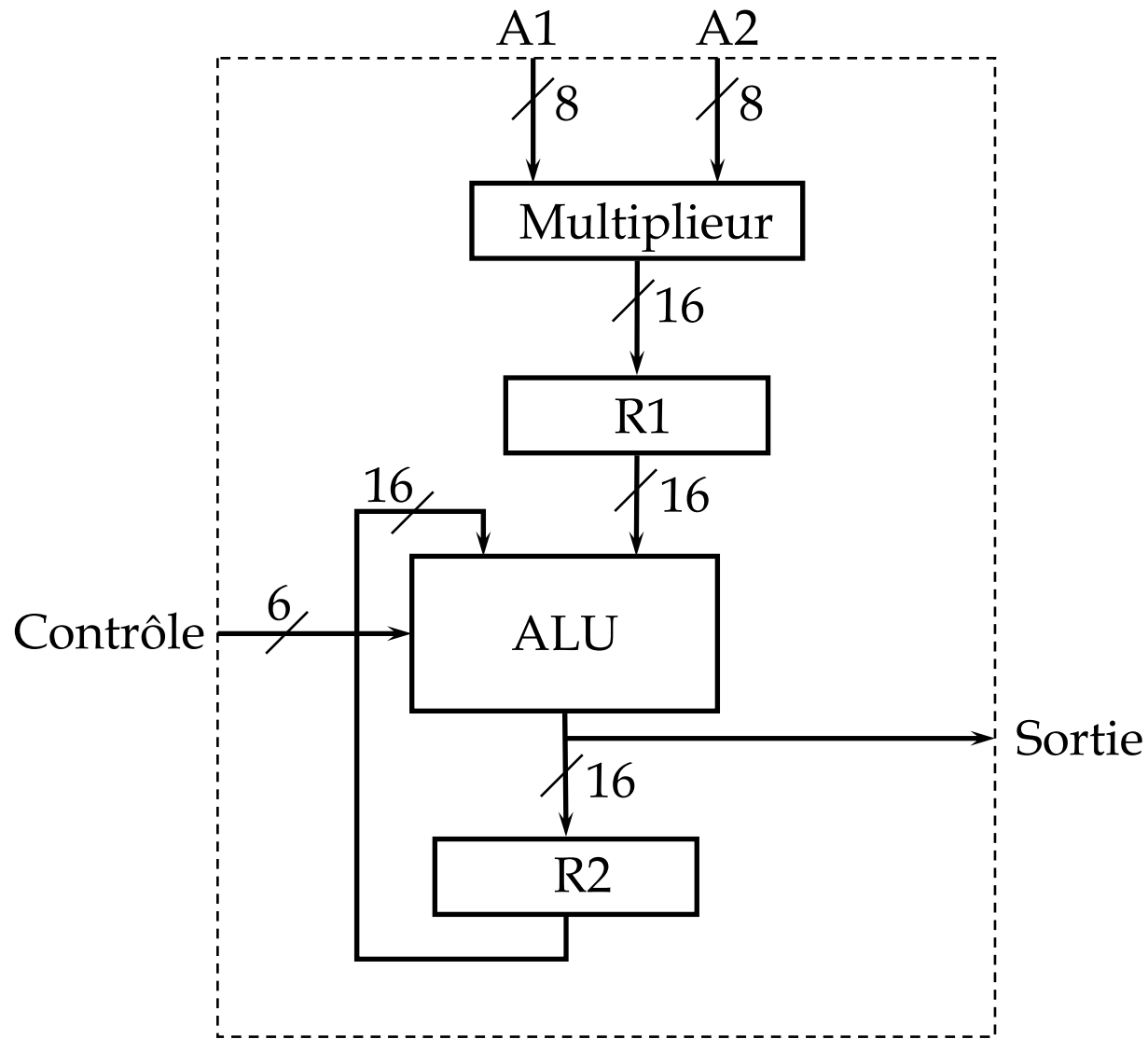
Test intégré avec BILBO



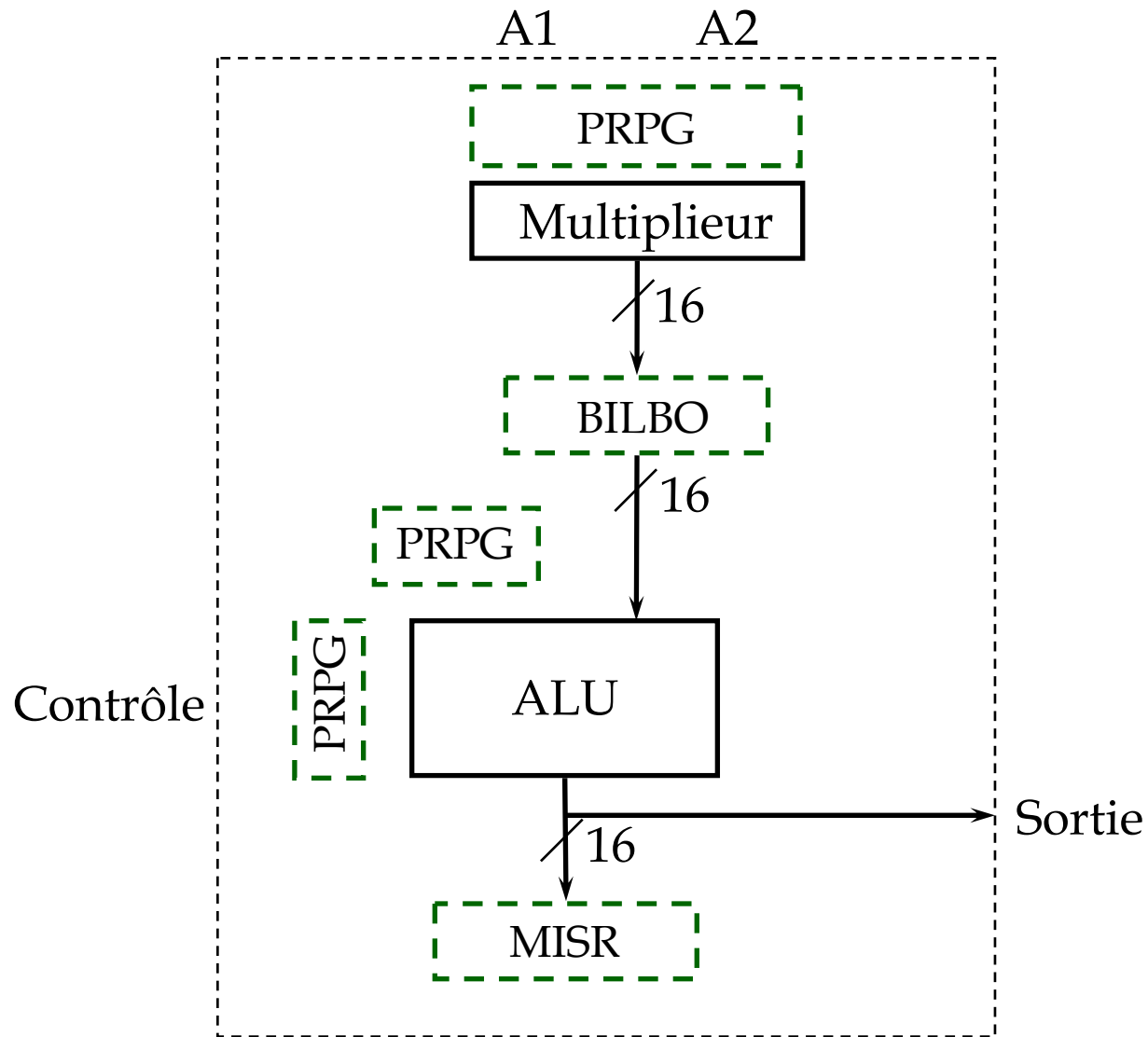
Test intégré avec BILBO pour les circuits séquentiels

- Le BILBO remplace les registres du circuit séquentiel, et est utilisé comme, générateur de vecteur de test LFSR et analyseur de signature MISR

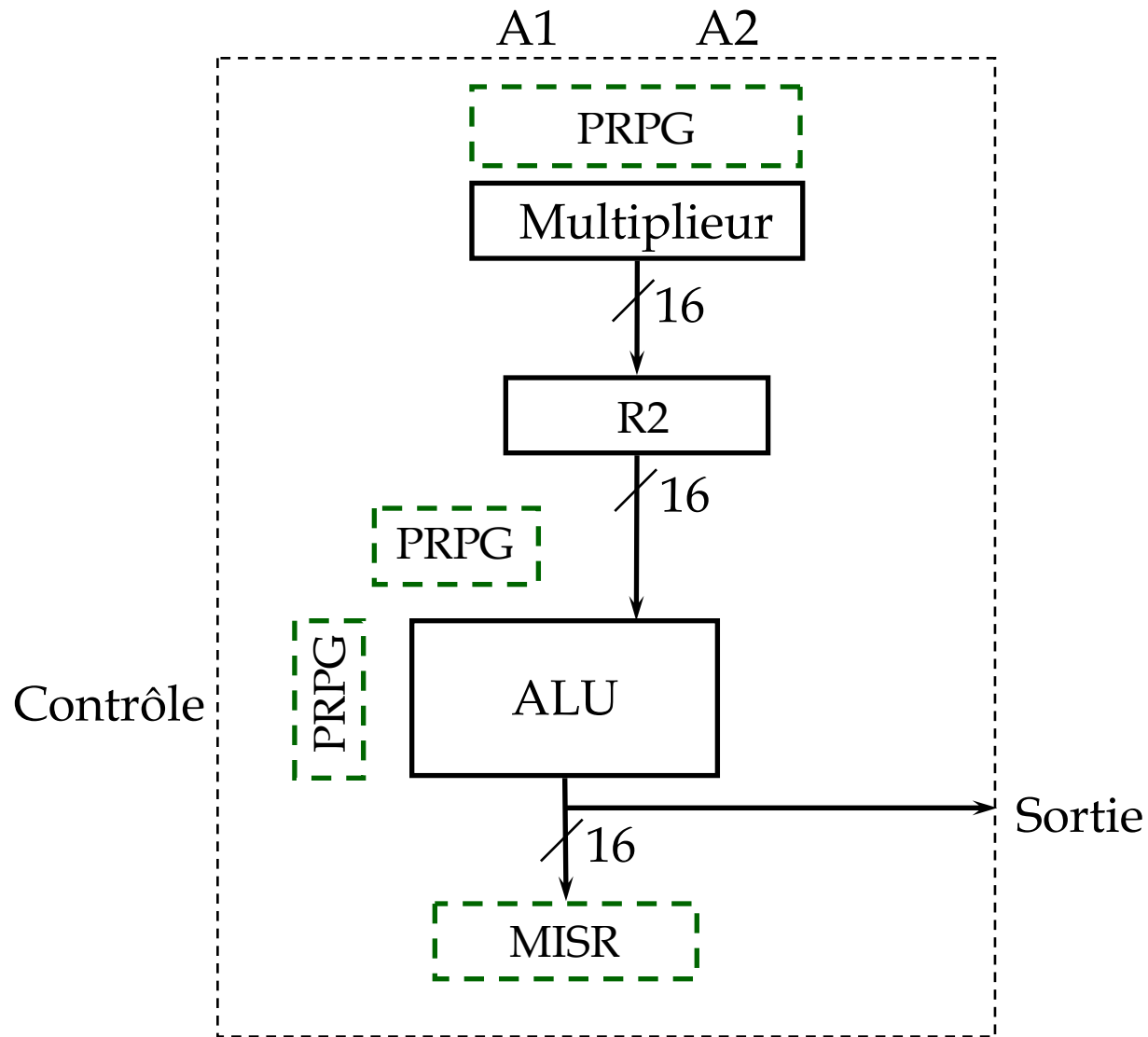
Exemple d'architecture de BIST



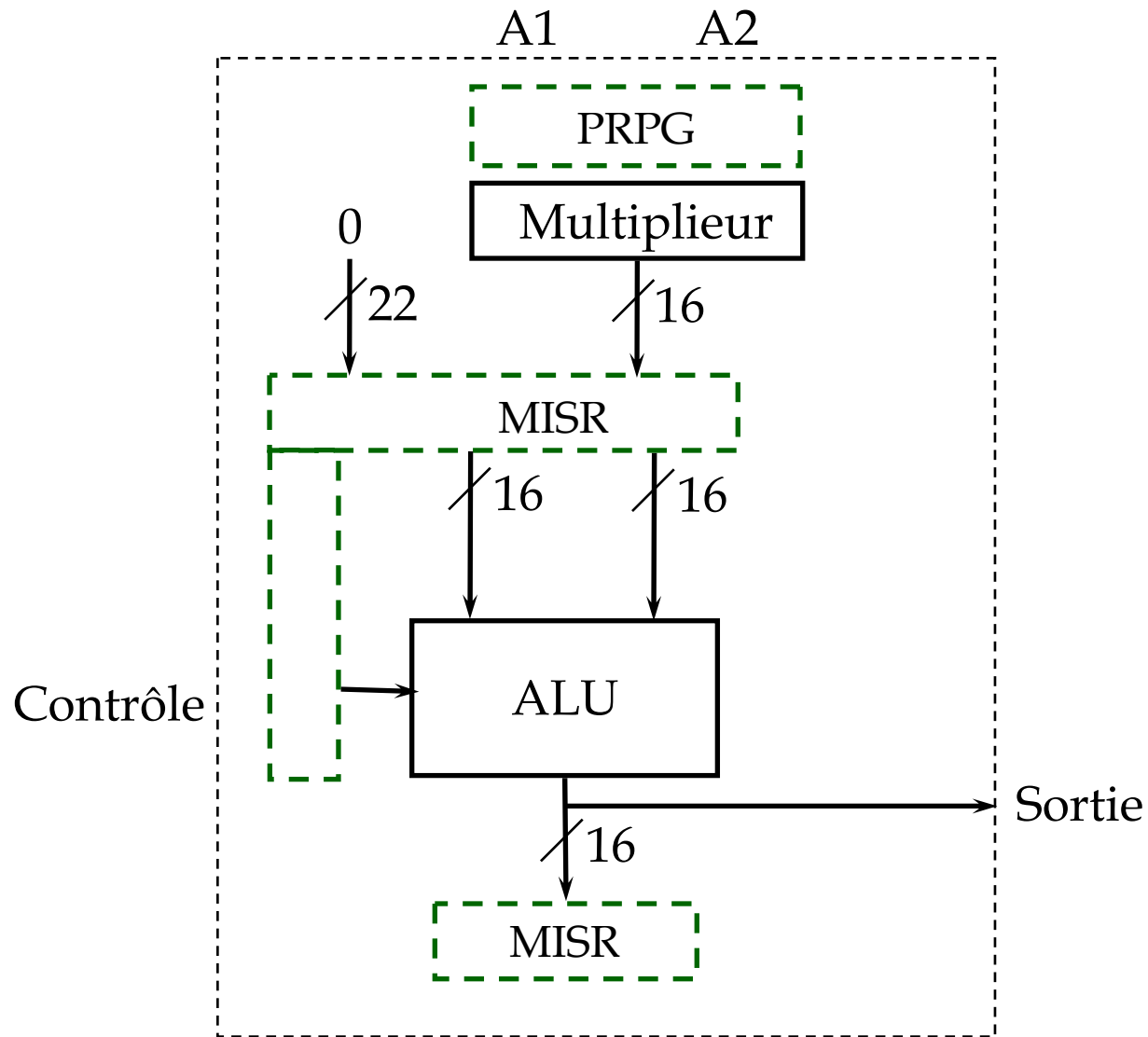
Exemple d'architecture de BIST : solution 1



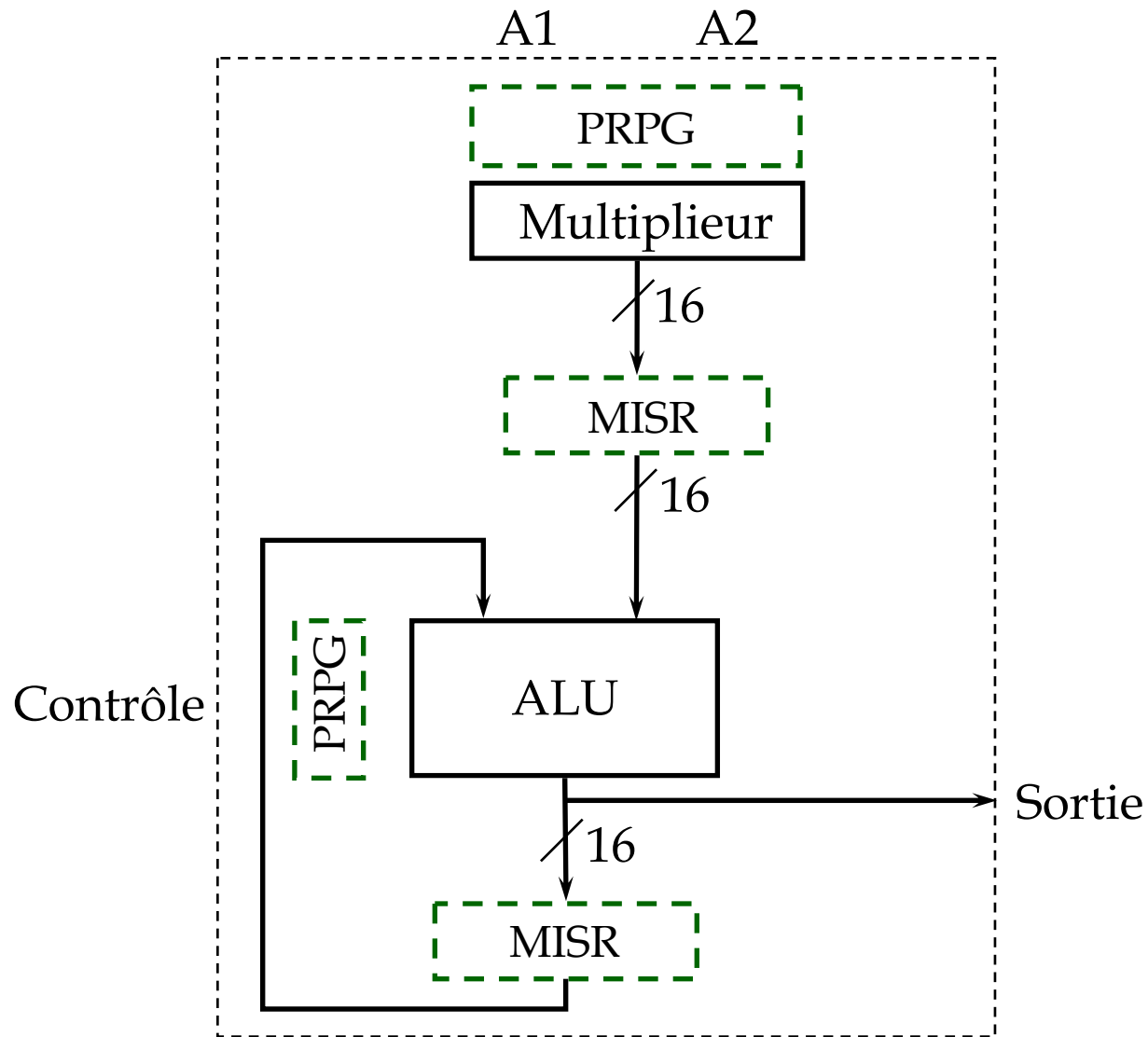
Exemple d'architecture de BIST : solution 2



Exemple d'architecture de BIST : solution 3



Exemple d'architecture de BIST : solution 4



Exemple d'architecture de BIST : comparaison

Solution	Nombre de vecteurs			Taux de couverture
	minimum	maximum	moyenne	
1	830	3619	2177	100%
2			>3000	64,5%
3	634	2531	1457	100%
4	721	2121	1378	100%

Conclusion

☞ Définition du BIST

- Avantages et inconvénients
- Architecture

☞ Les méthodes et techniques de génération des vecteurs de test

☞ Générateur LFSR

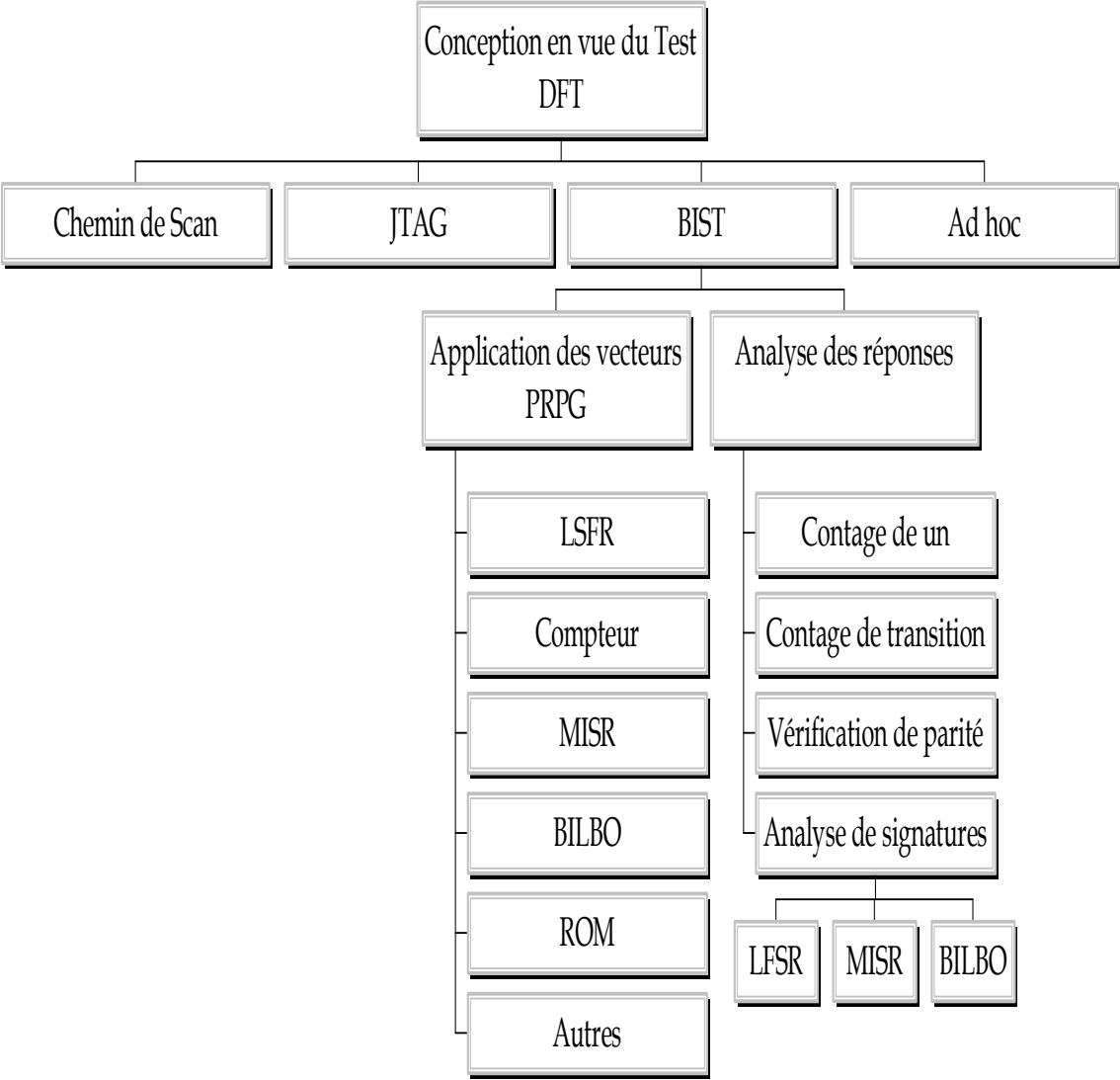
- Théorie et architecture

☞ Les techniques d'analyse des réponses

☞ Analyse de signature avec MISR

☞ Technique BILBO

Conclusion (suite)



Documentation

ATPG and BIST

- Kagaris, D.;
- The VLSI Handbook ; Ed. Wai-Kai Chen, CRC Press LLC, 2000.

CAD Tools for BIST/DFT and Delay Faults

- Tragoudas, S. ;
- The VLSI Handbook. Ed. Wai-Kai Chen, CRC Press LLC, 2000.

DS-LFSR: a BIST TPG for low switching activity

- Wang, S.; Gupta, S.K.;
- IEEE Transactions on Computer-Aided Design of Integrated
- Circuits and Systems, July 2002 Page(s): 842 -851

An analysis of the probabilistic behavior of linear feedback signature registers

- Ivanov, A.; Agarwal, V.K.;
- IEEE Transactions on Computer-Aided Design of Integrated
- Circuits and Systems, Oct. 1989, pp. 1074 -1088

Linear Feedback Shift Register v3.0

- Xilinx.

Questions

