



ECC Algorithm ***(Error Checking & Correction)***

Traduction Fr par Dr Lous

**Memory Product &
Technology Division**

1997.8.30

ECC Code Generation

- ◆ Le code ECC se compose de 3 octets par 256 octets
 - Réellement 22bit utilisé pour le code ECC par 2048bits (256 octets)
 - 22bit ECC code = 16bit parité de ligne + 6bit parité de colonne
- ◆ Table d'assignement des bits de données avec le code ECC

I/O 7	I/O 6		I/O 1	I/O 0
D(00000000,111)	D(00000000,110)		D(00000000,001)	D(00000000,000)
D(00000001,111)	D(00000001,110)		D(00000001,001)	D(00000001,000)
D(00000010,111)	D(00000010,110)	• • • •	D(00000010,001)	D(00000010,000)
D(11111110,111)	D(11111110,110)		D(11111110,001)	D(11111110,000)
D(11111111,111)	D(11111111,110)		D(11111111,001)	D(11111111,000)

1er Byte
2ème Byte
3ème Byte

255ème Byte
256ème Byte

◆ Table d'assignement du code ECC

I/O 7	I/O 6	I/O 5	I/O 4	I/O 3	I/O 2	I/O 1	I/O 0
P64	P64`	P32	P32`	P16	P16`	P8	P8`
P1024	P1024`	P512	P512`	P256	P256`	P128	P128`
P4	P4`	P2	P2`	P1	P1`	1	1

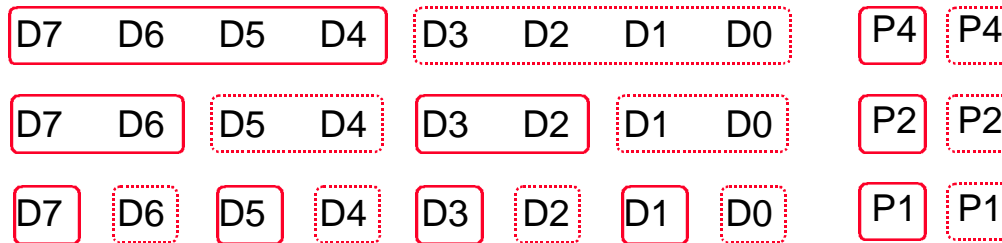
P8 ~ P1024 : Parité de ligne
P1 ~ P4 : Parité de colonne

Adresses des Bit changé ☰

=> (P1024,P512,P256,P128,P64,P32,P16,P8, P4,P2,P1)

Exemple Simple de génération de code ECC

◆ Génération de la parité (Dans le cas de 8 bits en entrée)



$$P4 = D7(+)D6(+)D5(+)D4 \quad P4' = D3(+)D2(+)D1(+)D0$$

$$P2 = D7(+)D6(+)D3(+)D2 \quad P2' = D5(+)D4(+)D1(+)D0$$

$$P1 = D7(+)D5(+)D3(+)D1 \quad P1' = D6(+)D4(+)D2(+)D0$$

* (+) est un XOR (Ou exclusif)

◆ Pour exemple (Dans le cas ou 1 bit serait faux)

Donnée d'origine : 1 0 1 0 1 0 1 0

↓

Données modifiées : 1 0 1 **1** 1 0 1 0

$$P4 = 1(+)0(+)1(+)0 = 0, \quad P4' = 1(+)0(+)1(+)0 = 0$$

$$P2 = 1(+)0(+)1(+)0 = 0, \quad P2' = 1(+)0(+)1(+)0 = 0$$

$$P1 = 1(+)1(+)1(+)1 = 0, \quad P1' = 0(+)0(+)0(+)0 = 0$$

$$P4 = 1(+)0(+)1(+)\mathbf{1} = \mathbf{1}, \quad P4' = 1(+)0(+)1(+)0 = 0$$

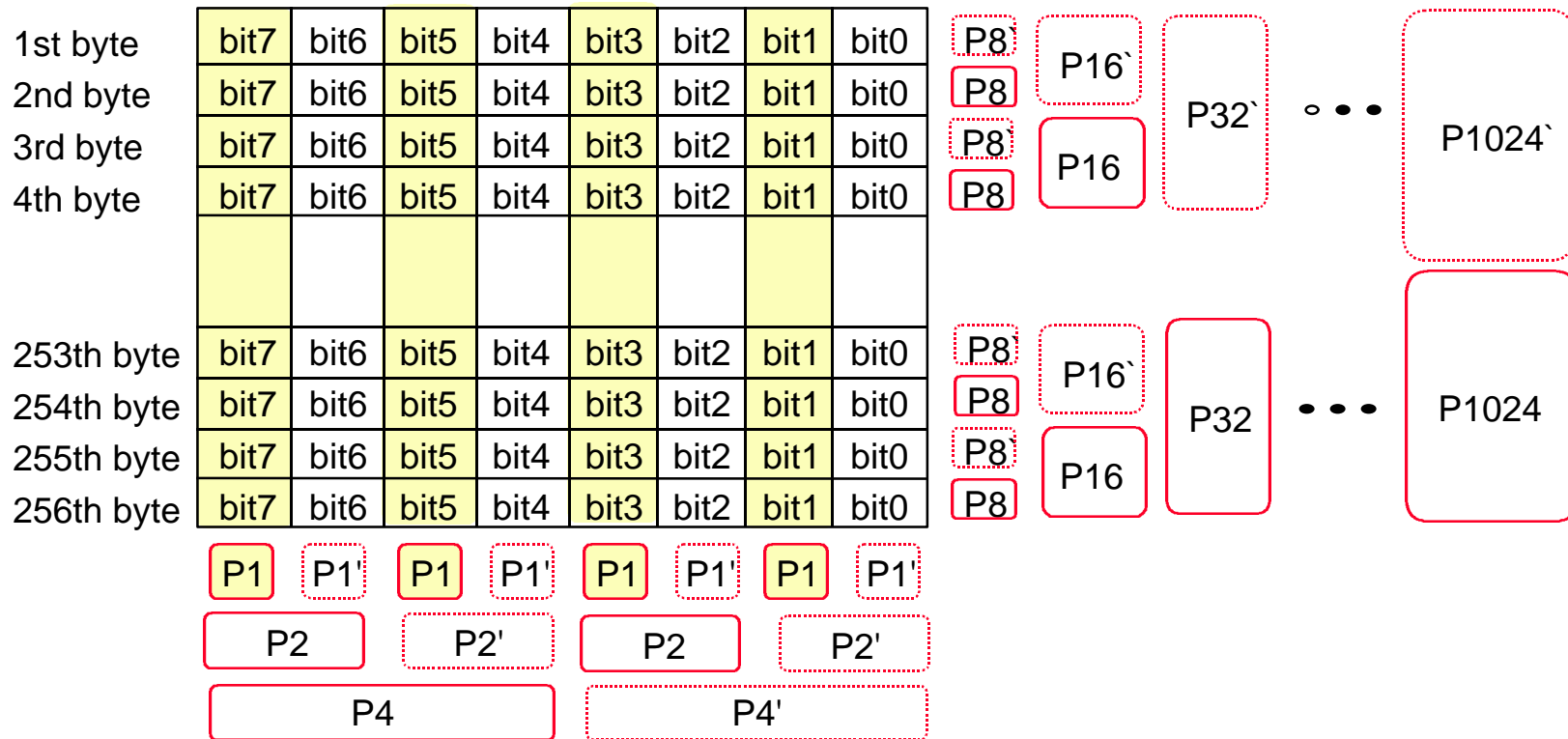
$$P2 = 1(+)0(+)1(+)0 = 0, \quad P2' = 1(+)\mathbf{1}(+)1(+)0 = \mathbf{1}$$

$$P1 = 1(+)1(+)1(+)1 = 0, \quad P1' = 0(+)\mathbf{1}(+)0(+)0 = \mathbf{1}$$

Ici, le bit faux a été identifié ($P4, P2, P1 = 100 = I/O4$)

Méthode de Génération du Code ECC (1)

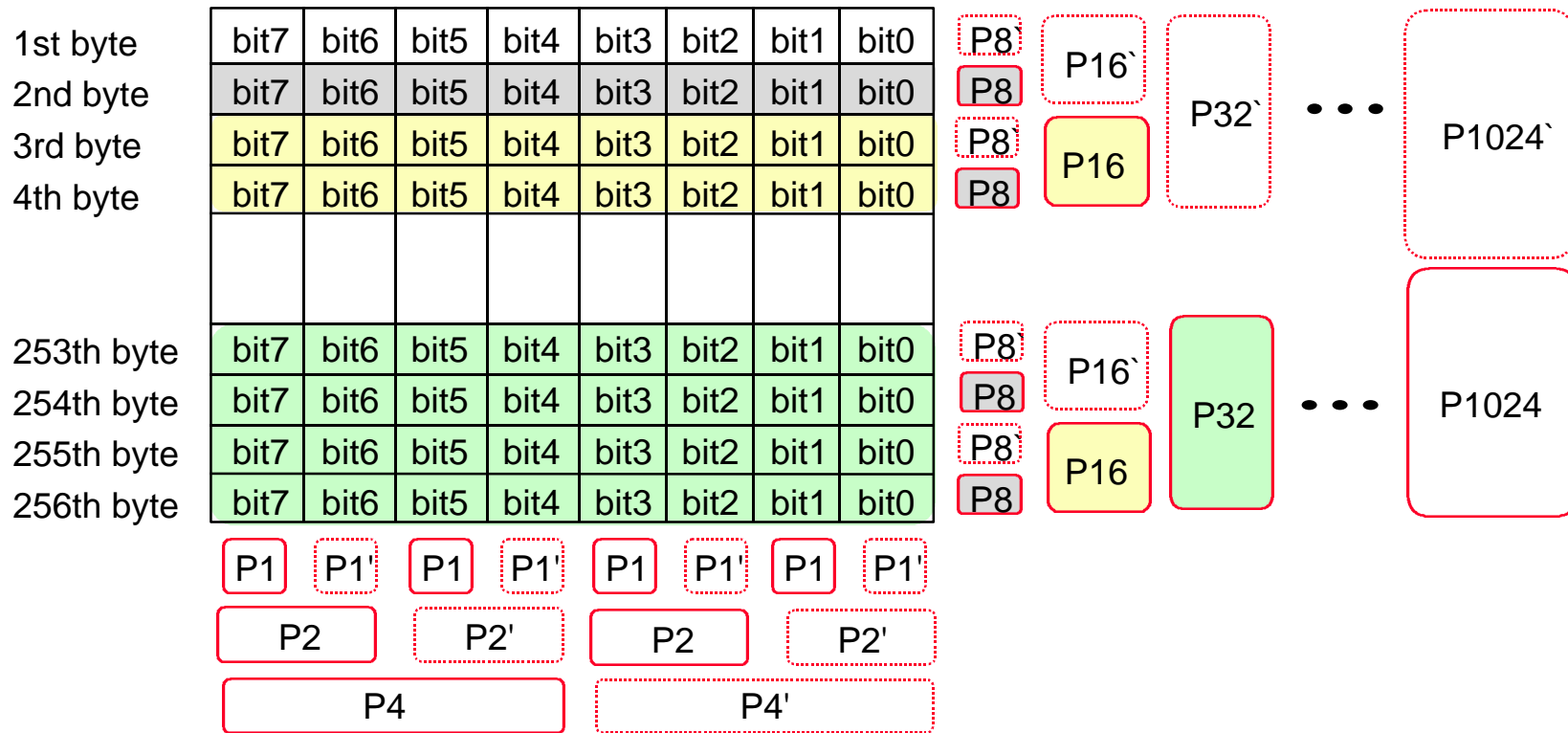
◆ Génération de la parité (dans le cas d'une entrée de 256 octets)



$P1 = \text{bit7}(+) + \text{bit5}(+) + \text{bit3}(+) + \text{bit1}(+) + P1$
 $P2 = \text{bit7}(+) + \text{bit6}(+) + \text{bit3}(+) + \text{bit2}(+) + P2$
 $P4 = \text{bit7}(+) + \text{bit6}(+) + \text{bit5}(+) + \text{bit4}(+) + P4$

Méthode de Génération du Code ECC (2)

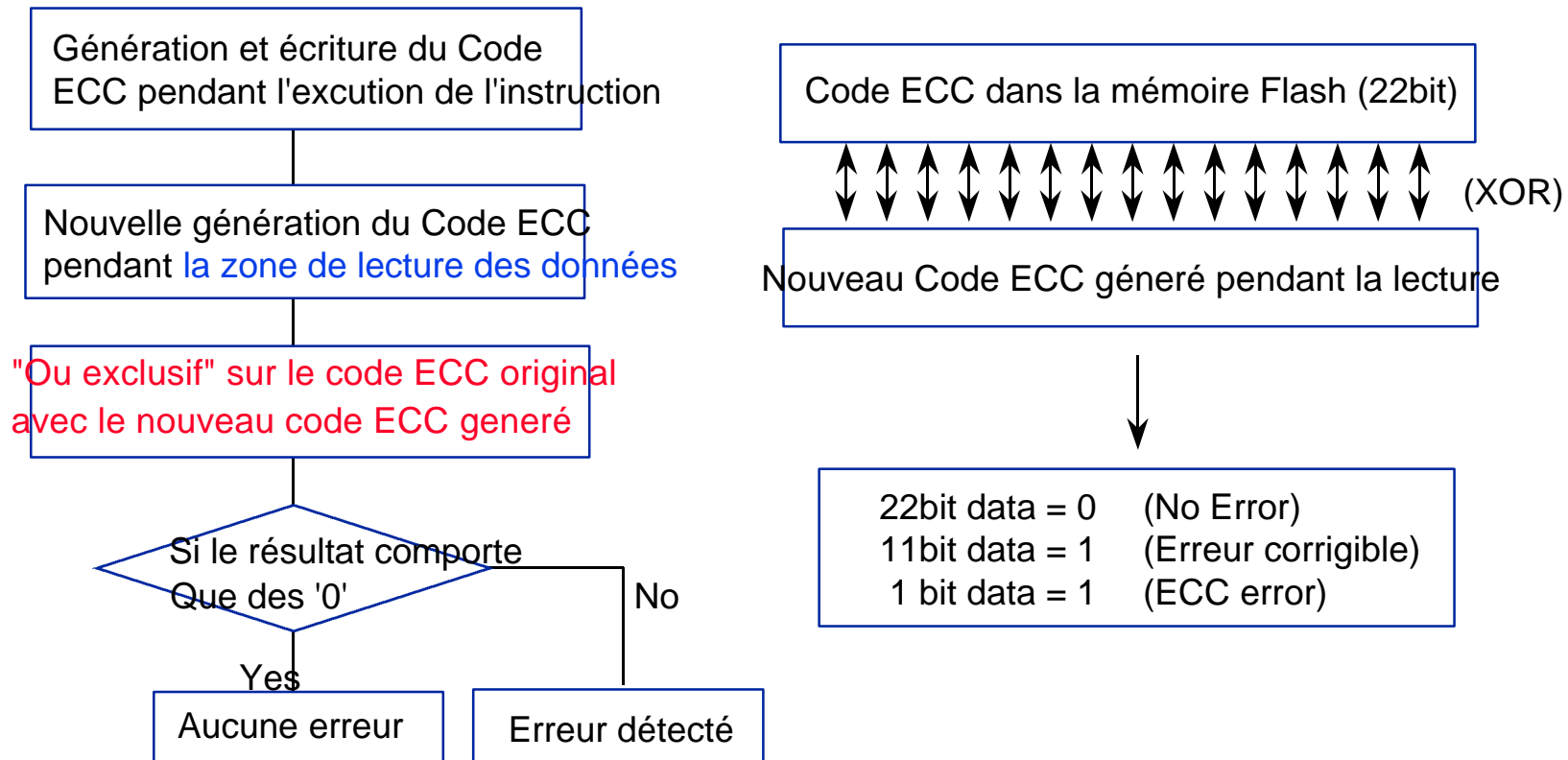
◆ Génération de la parité (dans le cas d'une entrée de 256 octets)



P8 = bit7(+)+bit6(+)+bit5(+)+bit4(+)+bit3(+)+bit2(+)+bit1(+)+bit0(+)+P8

Méthode de détection d'erreur

◆ Séquence de détection d'erreur



Résultat de la détection d'une erreur

◆ Aucune erreur

- Le résultat du "Ou exclusif" : Tout les codes ECC sont à `0`

◆ Erreur corrigible

- Le résultat du "Ou exclusif" : total 11bits sont à `1`
- Si dans le secteur (zone de ECC code) a 1 bit d'erreur, les parité de paire (ex. P8 & P8`) a 1&0 ou 0&1.

◆ ECC Error

- Le résultat du "Ou exclusif" : seulement 1bit est à `1`
- Quand le code ECC à lui même une erreur, on appelle ca une Erreur ECC.

◆ Erreur irréparable

- Le résultat du "Ou exclusif" : données aléatoire
- Quand la mémoire Flash a plus de 2 bits d'erreur, on ne peut plus corriger les données.

Correction d'erreur

- ◆ Pour une erreur corrigéable, nous pouvons corriger les données érronées.

- ◆ Exemple de correction d'erreur

 - Après l'exécution du "Ou exclusif", nous pouvons avoir le code ECC final.

Si le code ECC final P1024,P512,P256,P8,P4,P2,P1 is 01001100 110.

L'octet mauvais : (01001100)ème donnée (76ème)

Le mauvais bit : (110)ème bit (I/O6 | D6)

ECC Algorithm Executed Result Example

Original Data : 123456789

ECC code : 01000011 001

Changed Data : 12345**7**789

ECC code : 01000110 001

ECC code : 00000101 000 (5h 0h)

=>Fail bit is 6th byte 1st bit

	P1024	P1024'	P512	P512'	P256	P256'	P128	P128'	P64	P64'	P32	P32'	P16	P16'	P8	P8'	P4	P4'	P2	P2'	P1'	P1
Original ECC	0	0	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1
New ECC	0	1	1	0	0	1	0	1	0	1	1	0	1	0	0	1	0	1	0	1	1	0
Result of XOR	0	1	0	1	0	1	0	1	0	1	1	0	0	1	1	0	0	1	0	1	1	0

- Les données résultant du "Ou exclusif", chaque paire ont des données alternatives qui signifient l'erreur corrigible.
- Puisque nous connaissons l'adresse du mauvais bit, nous pouvons le corriger