

12)

DEMANDE DE BREVET D'INVENTION

A1

22) Date de dépôt : 13.03.98.

30) Priorité :

43) Date de mise à la disposition du public de la demande : 17.09.99 Bulletin 99/37.

56) Liste des documents cités dans le rapport de recherche préliminaire : *Se reporter à la fin du présent fascicule*

60) Références à d'autres documents nationaux apparentés :

71) Demandeur(s) : THOMSON MULTIMEDIA Société anonyme — FR.

72) Inventeur(s) : VIROULAUD CHRISTOPHE, PAUL PHILIPPE, DAOUDI MOHAMED et ISOLA PHILIPPE.

73) Titulaire(s) :

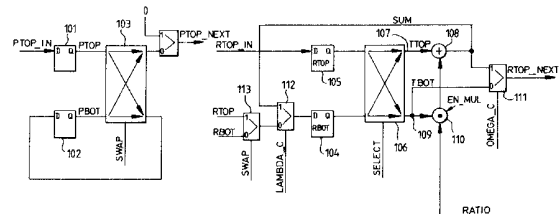
74) Mandataire(s) : THOMSON MULTIMEDIA.

54) DISPOSITIF DE CORRECTION D'ERREURS ET LECTEUR DE DISQUE OPTIQUE COMPORTANT UN TEL DISPOSITIF.

57) L'invention concerne un dispositif (1) de correction d'erreurs de type Reed-Solomon. Ce dispositif comprend un générateur de syndromes (2), des moyens (3) de détermination des coefficients des polynômes de localisation et d'évaluation d'erreur, des moyens (4) de détermination de la position et de la valeur d'erreurs, lesdits moyens (3) de détermination des coefficients étant aptes à mettre en oeuvre l'algorithme étendu d'Euclide.

Le dispositif est caractérisé en ce que les moyens (3) de détermination des coefficients sont agencés pour réaliser le produit du polynôme de localisation d'erreur ($\Lambda(z)$), et d'un polynôme d'effacement ($E(z)$) pour l'obtention d'un nouveau polynôme de localisation d'erreur ($\Psi(z)$).

L'invention concerne également un lecteur de disques optiques comprenant un tel dispositif.



L'invention concerne un dispositif de correction d'erreurs de type Reed-Solomon et un appareil de lecture optique comportant un tel dispositif. L'invention s'applique notamment dans le cadre de la correction d'erreur dans
5 des appareils de lecture de disques optiques ou magnéto-optiques.

La correction d'erreur de type Reed-Solomon ('RS') est utilisée dans de nombreuses applications. Une des applications les plus répandues est le disque compact ('CD'). Dans l'avenir, les codes RS seront également employés dans le cadre des disques dits 'DVD'.

10 Bien qu'ayant été développés dans les années 60, les codes RS n'ont été employés dans des appareils pour le grand public que depuis le début des années 80. Ceci est du en partie au fait que cette correction d'erreur exige une puissance de calcul relativement importante, donc coûteuse. La correction requiert par exemple la multiplication et la division de polynômes, ainsi que la
15 détermination des racines de polynômes. Dans ce cadre, il est intéressant de réduire la complexité des circuits mis en oeuvre dans cette correction, en vue d'un réduire le coût et d'en augmenter la fiabilité.

L'invention a pour objet un dispositif de correction d'erreurs de type
20 Reed-Solomon comportant :

- un générateur de syndromes,
- des moyens de détermination des coefficients des polynômes de localisation et d'évaluation d'erreur,
- des moyens de détermination de la position et de la valeur
25 d'erreurs, lesdits moyens de détermination des coefficients étant aptes à mettre en oeuvre l'algorithme étendu d'Euclide,
caractérisé en ce que

les moyens de détermination des coefficients sont agencés pour réaliser le produit du polynôme de localisation d'erreur et d'un polynôme
30 d'effacement pour l'obtention d'un nouveau polynôme de localisation d'erreur.

L'algorithme étendu d'Euclide nécessite la multiplication de polynômes. Selon l'invention, cette possibilité est mise à profit pour réaliser d'autres multiplications de polynômes nécessaires pour la correction d'erreur,
35 ce qui permet de simplifier les autres parties du dispositif de correction.

Selon un mode de réalisation de l'invention, les moyens de détermination des coefficients sont agencés pour réaliser le produit d'un polynôme de syndromes par le polynôme d'effacement.

5 L'invention a aussi pour objet un lecteur de disques optiques comportant le dispositif de correction.

D'autres avantages et caractéristiques de l'invention apparaîtront à travers la description d'un exemple de réalisation non limitatif décrit à l'aide des figures parmi lesquelles :

10 la figure 1 est un diagramme illustrant le codage RS d'un bloc de données de type CD,

la figure 2 est un diagramme illustrant le codage RS d'un bloc de données de type DVD,

15 la figure 3 est un diagramme bloc du dispositif conforme au présent exemple de réalisation,

la figure 4 est un diagramme bloc illustrant les échanges de données en mode dit mode 'DVD',

20 la figure 5 est un diagramme bloc illustrant les échanges de données en mode dit mode 'CD',

la figure 6 est un diagramme illustrant le stockage d'un paquet de données de type DVD dans une mémoire du dispositif conforme à l'invention,

la figure 7 est un diagramme illustrant l'organisation des données à l'intérieur d'une mémoire du dispositif conforme à l'invention,

25 la figure 8 est un schéma des deux registres de coefficients de polynômes utilisés par le résolveur d'équations des figures 3 à 5,

la figure 9 est un diagramme illustrant la connexion de cellules à registres dans un circuit résolveur d'équations utilisé dans le décodage Reed-Solomon,

30 la figure 10 est un diagramme bloc d'une cellule de la figure 9,

la figure 11 est un diagramme bloc de la cellule de la figure 10 dans une première configuration,

la figure 12 est un diagramme bloc de la cellule de la figure 10 dans une seconde configuration,

35 la figure 13 est un diagramme bloc de la cellule de la figure 10 dans une troisième configuration,

la figure 14 est un diagramme bloc de plusieurs cellules de la figure 10 placées en série selon une première configuration,

la figure 15 est un diagramme bloc de plusieurs cellules de la figure 10 placées en série selon une deuxième configuration,

5 la figure 16 est un diagramme bloc d'un lecteur de disque optique comportant un dispositif conforme à l'invention.

Le standard CD ('Compact Disc') préconise l'utilisation d'un code correcteur d'erreur Reed-Solomon à entrelacement croisé ("Cross Interleaved Reed-Solomon error-correction code" ou CIRC). La mise en oeuvre de ce code
10 comporte le partage de chaque échantillon digital de 16 bits à coder en deux symboles de 8 bits.

La figure 1 illustre plus particulièrement la structure d'un bloc de
15 données (trame) conformément au standard CD.

Les échantillons à coder sont stockés dans une mémoire vive et présentés à l'encodeur sous la forme de blocs de vingt-quatre symboles (correspondant à six échantillons de 16 bits pour chacun des deux canaux). Selon une première étape, ces symboles sont entrelacés, en retardant de deux
20 échantillons les échantillons pairs et en intervertissant les connexions vers la première partie de l'encodeur, dit encodeur C2. L'encodeur C2 détermine quatre symboles de parité de 8 bits (dits symboles Q) à partir du mot de code de 24 symboles, et insère les quatre symboles de parité parmi les autres symboles. En résultent 28 symboles. L'encodeur C2 est donc un encodeur
25 (28,24).

Les symboles sont transmis au second encodeur, appelé encodeur C2, à travers 28 lignes à retard de longueur inégale, multiples de quatre blocs, de façon à répartir les 28 symboles d'un mot sur 28 blocs différents. Ces 28 blocs seront ensuite entrelacés avec d'autres blocs, à raison d'un bloc parmi
30 les 28 blocs pour trois autres blocs. L'encodeur ajoute quatre symboles de parité de 8 bits supplémentaires (symboles P) à chaque bloc, portant le total à 32 symboles. L'encodeur C1 est donc un encodeur (32,28). Ces symboles sont utilisés, après un traitement supplémentaire consistant à inverser les symboles de parité et à retarder les symboles pairs, pour former un paquet de données.

35 A ces données s'ajoutent un certain nombre d'informations de contrôle, qui ne seront pas détaillées ici. Pour plus d'informations sur le codage des données, on se reportera entre autres au chapitre 9 de l'ouvrage 'Digital

Audio and Compact Disc Technology', troisième édition 1995, Sony Service Centre Europe, ou encore à l'ouvrage 'Reed-Solomon Codes and their applications', Wicker/Bhagarva, IEEE Press 1994, chapitre 4.

5 La correction des paquets de type CD nécessite donc deux passes à travers un décodeur Reed-Solomon, la première pour le décodage C1 et la seconde pour le décodage C2, avec un désentrelacement entre les deux passes.

10 Si la capacité de correction de C1 est dépassée (en fonction du nombre d'effacements et d'erreurs à corriger), les 28 symboles (32 moins les quatre symboles de parité P) sont marqués de drapeaux d'effacement ('erasure flags') et ne sont pas corrigés. Lors du désentrelacement, ces 28 symboles seront répartis sur 28 blocs.

15 Si la capacité de correction de C2 est dépassée, les 24 symboles (28 moins les quatre symboles de parité Q) sont marqués d'un drapeau et ne sont pas corrigés.

20 La correction d'erreur utilisée dans le cadre du DVD ('Digital Video/Versatile Disc') est différente: un bloc de 192 lignes de 172 symboles (un symbole = 1 octet) est associé verticalement à un code Reed-Solomon dit externe ('PO') composé de 16 lignes de 172 octets, puis horizontalement à un code Reed-Solomon dit interne ('PI') composé de 208 lignes de 10 octets. Le code interne est calculé à partir du bloc de données et à partir du code externe. L'ensemble donne un bloc de 208 lignes de 182 octets. Les 16 lignes du code externe sont ensuite insérées de manière entrelacée parmi les 192
25 lignes restantes.

La structure d'un paquet de données conforme au standard DVD est illustrée par la figure 2.

30 La figure 3 est un diagramme bloc du dispositif selon le présent exemple de réalisation.

Un circuit 1 comporte en série, pour les besoins de la correction d'erreur proprement dite, un générateur de syndromes 2, un résolveur d'équations 3 et un correcteur 4.

35 Une mémoire vive 5 externe au circuit 1, par exemple du type SRAM, sert à stocker certains paquets de données brutes à corriger et les données corrigées. Une mémoire vive interne 6, utilisée uniquement en mode

CD, est destinée à stocker des résultats de décodage intermédiaire. Ceci évite avantageusement un accès trop fréquent à la mémoire externe 5.

Le circuit 1 comporte en outre quatre multiplexeurs qui, selon leur état, définissent le mode de fonctionnement CD ou le mode de fonctionnement DVD, ainsi que les diverses étapes dans le cadre de chaque mode. Un premier multiplexeur 7 sélectionne, à l'entrée du générateur de syndromes 2, soit les données en provenance de la SRAM 5, soit les données en provenance de la mémoire 6. Un second multiplexeur 8 sélectionne à l'entrée de la mémoire 6 soit les données en provenance de la SRAM 5, soit les données entrantes en provenance de l'extérieur du circuit (ligne I sur la figure 1). Un troisième multiplexeur 9 sélectionne, à l'entrée du correcteur 4, soit des données en provenance de la SRAM 5, soit des données en provenance de la mémoire 6, soit des données en provenance du résolveur d'équations 3. Un quatrième multiplexeur 10 alimente la SRAM 5, soit par des données corrigées en provenance du correcteur 4, soit par des données entrantes (I).

La mémoire 5 est utilisée pour stocker les blocs de type DVD, les drapeaux d'erreur ou d'effacements associés à ces blocs avant ou après une passe de correction, les blocs C2 de type CD dans un but d'entrelacement avant correction, des drapeaux d'erreur qui y sont associés, les blocs C1 corrigés et diverses données de synchronisation.

Chaque passe de correction d'erreur Reed-Solomon effectuée par le circuit 1 peut être résumée comme suite, que ce soit en mode CD ou DVD.

Etape 1. Calcul des syndromes S_j , du polynôme d'effacement $E(z)$ et du syndrome modifié $T(z)$:

$$S(z) = \sum_{j=0}^{r-1} S_j z^j$$

où r est le nombre de symboles de parité dans un mot de code, et où

$$S_j = \sum_{i=0}^{n-1} r_i \alpha^{ij} \quad (1)$$

où n est le nombre de symboles d'un mot de code, où les coefficients r_i sont tels que $r(x) = r_{n-1} \cdot z^{n-1} + r_{n-2} \cdot z^{n-2} + \dots + r_0$ où $r(x)$ est un mot de code reçu, et où α^i est une racine de $g(z)$, polynôme générateur,

$$E(z) = \prod_{k=1}^e (1 - z\alpha^{Jk}) \quad (2)$$

5 où α^{Jk} est la position d'un effacement k et où e est le nombre d'effacements,

$$T(z) = S(z)E(z) \bmod z^{2t} \quad (3)$$

où t est un paramètre du code RS.

10

Remarque: le paramètre r dépend de la correction en cours. C'est le nombre de syndromes à calculer pour chaque mot de code. Il est égal au nombre de symboles de parité dans le mot de code à corriger: 4 et 4 pour les deux codages C1 et C2 du disque compact, 10 et 16 pour les codages interne et externe du DVD.

15

Etape 2. Mettre en oeuvre l'algorithme d'Euclide étendu pour calculer le polynôme de localisation d'erreur $\Lambda(z)$ et le polynôme d'évaluation d'erreur $\Omega(z)$, et calculer le nouveau polynôme de localisation d'erreur $\Psi(z)$,
 20 où

$$\Psi(z) = \Lambda(z)E(z) \bmod z^{2t} \quad (4)$$

Etape 3. Mettre en oeuvre l'algorithme de Chien pour déterminer les racines du polynôme de localisation d'erreur. Les racines de ce polynôme indiquent les positions des erreurs et des effacements dans les symboles reçus. Mettre en oeuvre l'algorithme de Forney pour déterminer les valeurs des erreurs et des effacements à partir des racines du polynôme de localisation d'erreur, de la dérivée du polynôme de localisation d'erreur et du polynôme d'évaluation d'erreur.

25

30

Etape 4. Vérifier le processus de décodage et corriger les symboles reçus.

35 Les algorithmes mentionnés sont, en soi, connus par ailleurs. On se référera notamment à l'ouvrage 'Reed-Solomon Codes and their Applications',

édité par S.B. Wicker et V.K. Bhagarva, IEEE Press 1994, en ce qui concerne des implémentations de l'algorithme d'Euclide et les algorithmes de Chien et de Forney, et de manière générale, la théorie mathématique des codes Reed-Solomon. Il existe d'ailleurs dans la littérature d'autres algorithmes pour
5 réaliser chacune des étapes.

Dans ce qui suit, on décrira dans un premier temps le fonctionnement du dispositif dans chaque mode, DVD ou CD, puis dans un deuxième temps, on s'attachera à décrire plus en détail l'implémentation
10 matérielle de certains aspects du décodeur Reed-Solomon, en particulier le résolveur d'équations 3.

- Décodage DVD

15 Le fonctionnement en mode DVD est illustré par la figure 4. Par rapport au schéma général de la figure 3, les parties du circuit 1 non concernées par le fonctionnement en mode DVD ont été effacées.

(a) Code interne (PI) - Première passe

20 Les données entrantes sont stockées dans un premier temps dans la SRAM 5, en transitant par le multiplexeur 10. La figure 6 montre la façon dont les blocs sont stockés dans la SRAM 5. Ces données sont composées de deux blocs de 208×182 octets. Chaque bloc est accompagné de 208×23 octets
25 d'effacement. Chaque bit d'effacement indique pour un symbole d'un bloc si ce symbole est à considérer comme sûr ou non. Cette information est utilisée ensuite au cours de la correction d'erreur Reed-Solomon. Ces bits d'effacement sont déterminés de façon connue en soi par un ensemble extérieur au circuit 1. Selon le présent exemple de réalisation, cet ensemble
30 extérieur vérifie la présence de la donnée dans une table de démodulation numérique.

Dans un premier temps, le circuit procède au décodage du code RS interne PI. Le bloc de données ou symboles, ainsi que les bits d'effacement
35 sont lus ligne par ligne dans la SRAM 5 et transférés au générateur de syndromes 2. Celui-ci détermine pour chaque ligne (chaque mot de code) le

polynôme $S(z)$ et le polynôme d'effacement $E(z)$, tels que définis précédemment.

Pratiquement, les syndromes S_j sont calculés de la façon suivante :

5

$$S_j = \sum_{i=0}^{n-1} r_i T^{ij}$$

où T est la matrice associée au polynôme générateur $G(z)$. Selon le présent exemple, $G(z) = z^8 + z^4 + z^3 + z^2 + 1$. Dans ce cas, la matrice T est :

10

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Les calculs des S_j syndromes sont effectués par le générateur 2, à l'aide de j registres rebouclés chacun sur leur entrée respective à travers un multiplicateur par T^n et un additionneur. Les syndromes S_j constituent les coefficients du polynôme $S(z)$.

15

Dans le cas du décodage DVD, chaque ligne transférée comporte donc 182 octets de données, chaque octet représentant un symbole, ainsi que 182 bits d'effacement, un pour chaque symbole. Le code interne PI étant constitué de 10 symboles de correction, 10 syndromes sont calculés. Le générateur 2 détermine également à partir des bits d'effacement les coefficients E_j du polynôme d'effacement $E(z)$ et les stocke dans un registre approprié.

20

S'il y a plus de dix effacements dans une ligne, la correction n'est pas possible et le mot de code est marqué comme erroné. L'état erroné d'un mot de code est inscrit dans la mémoire SRAM (Figure 6), sous la forme d'un bit ('Inner 1 Error Flag').

25

Les coefficients de $S(z)$, et le cas échéant de $E(z)$, sont transmis au résolveur d'équations 3.

Si un polynôme d'effacement $E(z)$ existe (au moins un effacement), alors le circuit 3 calcule le polynôme $T(z)$, produit de $E(z)$ par $S(z)$, puis met en oeuvre l'algorithme étendu d'Euclide pour déterminer en même temps le polynôme de localisation d'erreur $\Lambda(z)$ et le polynôme d'évaluation d'erreur $\Omega(z)$.

Si aucun effacement n'existe, alors la détermination de $\Omega(z)$ et de $\Lambda(z)$ est effectuée directement à partir de $S(z)$ (en ce qui concerne les calculs, on prendra dans ce cas $E(z)=1$).

Le circuit 3 détermine également, le cas échéant, $\Psi(z)$.

La structure et le fonctionnement du circuit 3 seront vus en détail plus loin.

Le correcteur 4 exécute de façon connue les algorithmes de Chien et Forney.

Ce correcteur disposera donc finalement des données suivantes :

- les positions des erreurs dans le mot de code traité,
- les valeurs de ces erreurs,
- le nombre d'effacements.

Les erreurs dans le mot de code sont corrigés en conséquence. Pour ce faire, le circuit correcteur 4 (figure 4) lit le mot de code à corriger dans la mémoire SRAM 5 et les positions et valeurs des erreurs à partir du circuit 3. Le correcteur 4 inscrit le mot de code corrigé dans la SRAM 5, à l'emplacement du mot de code initial. Les multiplexeurs 9 et 10 sont commandés en conséquence. Si la capacité du code RS est dépassée, le correcteur inscrit un drapeau d'indication d'erreur (Référence 'Inner 1 Error Flags' de la figure 6).

(b) Code externe (PO)

Le dispositif réalise ensuite une seconde correction sur la base du code PO. En fonction du résultat pour chaque mot de code vertical, un drapeau d'erreur est mis à une valeur appropriée (Référence 'Outer Error Flag' de la figure 6).

(c) Code interne (PI) - Seconde passe

Une fois la correction PO réalisée, une seconde correction horizontale est effectuée sur la base du code interne PI. Certains mots qui n'étaient pas corrigibles lors de la première passe peuvent l'être devenus suite à la correction de certains symboles de ces mots lors de la correction à base du code PO. Le cas échéant, des drapeaux d'erreur (Référence 'Inner 2 Error Flag' de la figure 6) sont mis à la valeur appropriée.

Cette dernière passe peut être optionnelle.

10

- Décodage CD

Le fonctionnement en mode CD est illustré par la figure 5. Par rapport au schéma général de la figure 3, les parties du circuit 1 non concernées par le fonctionnement en mode CD ont été effacées.

15

Dans un premier temps, les données à corriger (blocs C1 de 32 symboles) sont inscrites dans la mémoire vive interne 6 à travers l'entrée E. La figure 7 illustre la façon dont les données sont inscrites dans cette mémoire 6. La mémoire 6 comporte de l'espace pour trois blocs de 32 symboles chacun. Les blocs C1 sont entrelacés lorsqu'ils sont stockés dans la mémoire 6. Etant donné la nature de l'entrelacement, deux blocs sont nécessaires pour obtenir un mot de code C1. L'emplacement du troisième bloc est utilisé pour l'écriture du bloc C1 suivant, cette écriture ayant lieu pendant la correction d'un bloc précédent. 28 emplacements de la mémoire 6 sont réservés à un mot de code de 28 symboles (code C2). Quatre emplacements sont réservés à d'autres informations, ce qui donne à la mémoire une longueur de 128 mots. La mémoire possède une largeur de 9 bits, 8 bits étant destinés à stocker des symboles, le neuvième indiquant les effacements. Initialement, les bits d'effacement indiquent si les symboles stockés sont valides ou non valides.

20

25

Le générateur de syndromes 2 lit un premier mot de code désentrelacé dans la mémoire 6, ainsi que les bits d'effacement correspondants. Les traitements effectués par le circuit 2, ainsi qu'ensuite par le résolveur 3, sont similaires à ceux du mode DVD, étant entendu que les paramètres du code Reed-Solomon ne sont pas les mêmes. Le principe de la correction est cependant identique.

30

35

L'emplacement des erreurs et leur valeur est communiquée par le résolveur 3 au correcteur 4. Ce dernier reçoit de la RAM 6 tous les symboles du mot de code, en dehors des symboles de parité C1. Si une correction est nécessaire, alors le symbole erroné est remplacé par le symbole corrigé. Les symboles sont ensuite stockés dans la mémoire 5. Il est à noter que dans cette 'correction à la volée', tous les symboles transitent de la mémoire RAM 6 par le correcteur 4 vers la mémoire 5, tandis que dans le mode DVD, seuls les symboles à corriger transitent de la mémoire 5 à travers le correcteur et de nouveau vers la mémoire 5.

10 Cette architecture est basée sur la reconnaissance du fait que les mots de code en mode CD (28 ou 32 symboles) sont bien plus courts qu'en mode DVD (182 ou 208 symboles). Une correction à la volée est réalisée seulement sur les mots de code 'courts', et cela à partir de la mémoire rapide 6, car tous les symboles d'un mot de code doivent alors transiter par le correcteur. Une telle correction à la volée sur de longs mots de code exigerait plus de cycles de la part du correcteur 4, ainsi que la mise en oeuvre d'une mémoire 5 beaucoup plus rapide et par conséquent plus coûteuse si les mots de code à corriger de cette façon y étaient stockés.

20 Cette architecture permet par conséquent d'utiliser une mémoire 5 du commerce, celle-ci étant disposée hors du composant comportant le dispositif décrit ici, et de prévoir sur le composant lui-même une mémoire interne, de taille faible donc peu coûteuse. Ceci vaut pour le cas où la mémoire 5 est implémentée sur un composant autre que la mémoire 6.

25 Il est clair que l'évocation de mémoires 'externe' ou 'interne' n'exclut pas l'application de ce qui est décrit à un circuit intégré comportant à la fois la mémoire 5 et la mémoire 6. La mémoire 5 doit être vue comme une mémoire de masse, tandis que la mémoire 6 comme une mémoire cache.

30 L'entrelacement des symboles des mots de code C2 est tel que les 28 symboles d'un même mot de code sont répartis parmi 112 blocs de 32 symboles. A chaque correction de code C1 correspond une correction de code C2. Ainsi, le système ne sera correctement initialisé qu'après 112 trames. De plus, pour pouvoir effectuer le désentrelacement du code C2, après la correction de code C1, les symboles sont stockés dans la mémoire 5, de plus grande capacité que la mémoire 6 et sont relus pour être, d'une part, inscrits dans la mémoire 6 et, d'autre part, fourni directement au générateur de syndromes 2.

L'inscription dans la mémoire 6 est nécessaire pour la correction du mot de code par le correcteur 4 'à la volée', car comme mentionné à propos de la correction C1, dans le mode CD, le correcteur lit les mots de code à corriger non pas dans la mémoire SRAM 5, mais dans la mémoire vive 6.

Le processus de correction C2 est similaire à ce qui a déjà été exposé pour la correction C1. Suite à la correction à la volée, les symboles corrigés sont stockés dans la mémoire 5, sans toutefois les symboles de parité du code C2.

10

- Résolveur d'équations - Détermination de $\Omega(z)$ et de $\Lambda(z)$

La suite de la description concerne l'implémentation du résolveur d'équations.

15

La tâche principale du résolveur d'équations (dénommé aussi 'Key Equation Solver' en langue anglaise) est de déterminer les polynômes de localisation et d'évaluation d'erreur, respectivement $\Lambda(z)$ et $\Omega(z)$.

20

La détermination de $\Omega(z)$ et de $\Lambda(z)$ est effectuée en mettant en oeuvre l'algorithme étendu d'Euclide. On met en oeuvre pour cela deux registres de polynômes, respectivement RTOP et RBOT, illustrés par la figure 8.

L'algorithme d'Euclide étendu est le suivant :

25

Les polynômes $\Omega(z)$ et $\Lambda(z)$ satisfont l'équation :

$$\Lambda(z) \cdot S(z) = \Omega(z) \text{ mod } z^r$$

$$\text{deg } \Omega(z) < \text{deg } \Lambda(z)$$

30

(La variable 'r' est égale au nombre de symboles de parité d'un mot de code (capacité de correction du code), c'est à dire à 2t).

On prend comme valeurs initiales :

$$\Lambda_{n-1}(z) = 0$$

$$\Lambda_0(z) = 1$$

$$\Omega_{-1}(z) = Z^r$$

$$\Omega_0(z) = T(z)$$

(On prendra $\Omega_0(z) = S(z)$ dans le cas où aucun effacement n'est connu).

5

On réalise les itérations suivantes jusqu'à ce que

$$\text{deg ré} \Omega_n(z) < \text{deg ré} \Lambda_n(z)$$

$$\Omega_n(z) = \Omega_{n-2}(z) \bmod \Omega_{n-1}(z) \quad (\text{A})$$

10

$$\Lambda_n(z) = \Lambda_{n-1}(z) * Q_n(z) - \Lambda_{n-2}(z) \quad (\text{B})$$

$$Q_n(z) = \Omega_{n-2}(z) \text{div} \Omega_{n-1}(z) \quad (\text{C})$$

Les registres polynomiaux RTOP et RBOT comportent chacun $r+1$ registres (RTOP(k) et RBOT(k)) de 8 bits, appelés registres de symboles. Chacun de ces registres de symboles stocke un coefficient d'un des polynômes $\Lambda(z)$ ou $\Omega(z)$. Chaque registre RTOP et RBOT est séparé en une partie dite partie Λ et une partie dite Ω , de droite à gauche.

15

Initialement, le degré de $\Omega(z)$ est maximal et le degré de $\Lambda(z)$ est minimal. Dans les registres RTOP et RBOT, les degrés de $\Omega(z)$ augmentent vers la gauche, tandis que les degrés de $\Lambda(z)$ augmentent vers la droite, à partir du point de séparation entre les parties Λ et Ω .

20

Le tableau 1 illustre le contenu des registres en phase initiale de l'itération.

25

Rang de registre / cellule	r+1	r	...	2	1	0
RBOT	1	0	0	0	0	1
Pbot	0	0	0	0	0	1
RTOP	T_{r-1}	T_{r-2}	...	T_0	0	0
Ptop	0	0	...	0	1	1

Tableau 1

Les drapeaux Ptop et Pbot indiquent pour chaque registre de symbole si ce registre comporte un coefficient de $\Omega(z)$ (Ptop/bot=0) ou de $\Lambda(z)$ (Ptop/bot=1).

5 On définit par ailleurs 'DegTop' et 'DegBot' comme étant les degrés des parties Ω de RTOP et RBOT respectivement, ainsi que la variable 'ratio', qui désigne la rapport $RTOP(r+1)/RBOT(r+1)$.

L'algorithme étendu d'Euclide se traduit dans ce contexte par :

10

DegBot=r, DegTop=r-1

TANT QUE (DegTop>=t) FAIRE

SI ratio \neq 0 ET (DegTop<DegBot)

ECHANGER RTOP, RBOT

15

ECHANGER DegTop, DegBot

ECHANGER Ptop, Pbot

FIN SI

POUR k=r JUSQU'A 0

SI (Pbot(k)=Ptop(k)=1) ALORS

20

RBOT(k)=RBOT(k)+ratio*RTOP(k)

RTOP(k+1)=RTOP(k)

SINON SI (Pbot(k)=Ptop(k)=0) ALORS

RTOP(k+1)=RTOP(k)+ratio*RBOT(k)

SINON

25

RTOP(k+1)=RTOP(k)

FIN SI

Ptop(k+1)=Ptop(k)

FIN POUR

DegTop=DegTop-1

30

FIN TANT QUE

Lorsqu'il faut prendre en compte des effacements, la condition du TANT QUE devient $DegTop \geq m$, où $m = t + (\text{degré}E(z)/2)$.

35

L'indice k désigne une cellule d'ordre k et les registres qui y sont contenus.

On définit $r+1$ cellules, chaque cellule comprenant les deux registres de symboles de RTOP et RBOT ayant le même rang. Chaque cellule doit également effectuer les opérations définies ci-dessus.

5 La figure 9 est un schéma illustrant la connexion des $r+1$ cellules pour former les registres RTOP et RBOT. Le rectangle en tirets identifie une cellule de base, qui comprend les différents registres, multiplexeurs, additionneurs et le multiplieur nécessaire pour l'implémentation de l'algorithme.

10 La figure 10 est un diagramme fonctionnel d'une cellule.

La cellule comporte un registre à 1 bit 101 destiné à stocker le drapeau Ptop, ainsi qu'un registre à 1 bit 102, destiné à stocker le drapeau Pbot. Ces registres sont constitués chacun d'une bascule D commandée de façon appropriée. Un circuit d'inversion 103 possède deux entrées, connectées
15 respectivement aux sorties non inversées Q des registres 101 et 102 (signaux Ptop et Pbot). Ce circuit d'inversion comporte deux sorties, dont la première, référencée Ptop_NEXT, est connectée à l'entrée du registre équivalent au registre 101 de la cellule suivante, et dont la seconde sortie reboucle sur l'entrée D du registre 102. Selon la valeur d'un signal de commande SWAP, Pbot est transmis à la première sortie et Ptop à la seconde, ou inversement.
20

La cellule comporte également deux registres à 8 bits, référencés 104 et 105, faisant partie respectivement du registre RBOT et RTOP. Ces registres sur 8 bits sont formés par exemple de 8 registres à 1 bit connectés en parallèle. Les sorties respectives des registres 104 et 105 sont connectées
25 chacune à une entrée d'un circuit inverseur 106, commandé par un signal SELECT.

Le circuit inverseur 106, à l'image du circuit inverseur 103, connecte chaque entrée soit à l'une, soit à l'autre de ses sorties, selon l'état du signal SELECT. La première sortie 107 du circuit inverseur 106 est reliée à une
30 entrée d'un additionneur 108. La seconde sortie 109 du circuit inverseur 106 est reliée à un multiplieur 110, qui multiplie la valeur présente à la sortie 109 par la variable 'ratio'. La sortie du multiplieur 110 est reliée à une entrée de l'additionneur 108.

Le registre 105 reçoit sur son entrée D un signal RTOP_IN, qui
35 correspond à la sortie du registre à 8 bits de RTOP de la cellule précédente.

Le signal de sortie de la cellule vers la prochaine cellule est référencé RTOP_NEXT. Cette sortie est constituée par la sortie d'un

multiplexeur 11, qui sélectionne, selon la valeur d'un signal de commande référencé OMEGA_C, soit la sortie de l'additionneur 108, soit la seconde sortie 109 du circuit inverseur 106.

En dernier lieu, l'entrée du registre RBOT 104 est reliée par l'intermédiaire d'un second multiplexeur 112, selon la valeur d'un signal de commande référencé LAMBDA_C, soit à la sortie de l'additionneur 108, soit à la sortie d'un troisième multiplexeur 113. Ce troisième multiplexeur sélectionne, selon la valeur du signal de commande SWAP, soit la contenu du registre 104 (RBOT), soit celui du registre 105 (RTOP).

10

Les signaux de commande sont élaborés de la façon suivante :

$$\text{LAMBDA_C} = \text{Pbot AND Ptop}$$

$$\text{OMEGA_C} = \text{NOT (Pbot OR Ptop)}$$

15

$$\text{SELECT} = \text{NOT (SWAP XOR OMEGA_C)}$$

$$\text{EN_MUL} = (\text{LAMBDA_C OR OMEGA_C})$$

20

La figure 11 représente une cellule dans le cas où Ptop et Pbot sont égaux à 0. Les deux registres 104 et 105 de la cellule contiennent alors des coefficients de $\Omega(z)$. Cette configuration de la cellule est par conséquent dite configuration Omega.

De manière similaire, les figures 12 et 13 représentent une cellule dans les cas respectifs où Ptop=Pbot=1 et Ptop est différent de Pbot, et sont appelées configurations Lambda et Mixte.

25

<u>Type de cellule</u>	<u>SWAP = 0</u>	<u>SWAP = 1</u>
Cellule Lambda	$\text{Pbot}_{t+1}(k) = \text{P}_{t\text{bot}}(k)$ $\text{Ptop_NEXT}_{t+1}(k) = \text{Ptop}_t(k)$ $\text{RBOT}_{t+1}(k) = \text{RBOT}_t(k) + \text{RTOP}_t(k) * \text{ratio}$ $\text{RTOP_NEXT}_{t+1}(k) = \text{RTOP}_t(k)$	$\text{Pbot}_{t+1}(k) = \text{P}_{t\text{top}}(k)$ $\text{Ptop_NEXT}_{t+1}(k) = \text{Pbot}_t(k)$ $\text{RBOT}_{t+1}(k) = \text{RTOP}_t(k) + \text{RBOT}_t(k) * \text{ratio}$ $\text{RTOP_NEXT}_{t+1}(k) = \text{RBOT}_t(k)$
Cellule Omega	$\text{Pbot}_{t+1}(k) = \text{P}_{t\text{bot}}(k)$ $\text{Ptop_NEXT}_{t+1}(k) = \text{Ptop}_t(k)$ $\text{RBOT}_{t+1}(k) = \text{RBOT}_t(k)$ $\text{RTOP_NEXT}_{t+1}(k) = \text{RTOP}_t(k) + \text{RBOT}_t(k) * \text{ratio}$	$\text{Pbot}_{t+1}(k) = \text{P}_{t\text{top}}(k)$ $\text{Ptop_NEXT}_{t+1}(k) = \text{Pbot}_t(k)$ $\text{RBOT}_{t+1}(k) = \text{RTOP}_t(k)$ $\text{RTOP_NEXT}_{t+1}(k) = \text{RBOT}_t(k) + \text{RTOP}_t(k) * \text{ratio}$

Cellule	$P_{bot_{t+1}}(k)=P_{t,bot}(k)$	$P_{bot_{t+1}}(k)=P_{t,top}(k)$
Mixte	$P_{top_NEXT_{t+1}}(k)=P_{top_t}(k)$	$P_{top_NEXT_{t+1}}(k)=P_{bot_t}(k)$
	$RBOT_{t+1}(k)=RBOT_t(k)$	$RBOT_{t+1}(k)=RTOP_t(k)$
	$RTOP_NEXT_{t+1}(k)=RTOP_t(k)$	$RTOP_NEXT_{t+1}(k)=RBOT_t(k)$

Tableau 2

Les parties Lambda réalisent la multiplication de l'équation (B), tandis que les parties Omega réalisent la division de l'équation (C).

5

Si l'on prend comme exemple le polynôme $S(z)$ tel que :

$$S(z) = \alpha^{14}z^5 + \alpha^{10}z^4 + \alpha^3z^3 - \alpha^7z^2 - \alpha^9z + \alpha^{12},$$

alors les conditions initiales sont :

$$\Lambda_{n-1}(z) = 0$$

$$\Lambda_0(z) = 1$$

$$\Omega_{-1}(z) = Z^6$$

$$\Omega_0(z) = S(z)$$

10

Les tableaux 3 à 10 illustrent le contenu des cellules à la fin de chaque itération de l'algorithme.

Rang	7	6	5	4	3	2	1	0
RBOT	1	0	0	0	0	0	0	1
Pbot	0	0	0	0	0	0	0	1
RTOP	α^{14}	α^{10}	α^3	α^7	α^9	α^{12}	0	0
Ptop	0	0	0	0	0	0	1	1

Tableau 3 (Conditions initiales)

(DegBot=6 - DegTop=5 - ratio= α^{14} - SWAP=1)

15

Rang	7	6	5	4	3	2	1	0
RBOT	α^{14}	α^{10}	α^3	α^7	α^9	α^{12}	0	1
Pbot	0	0	0	0	0	0	1	1
RTOP	1	0	0	0	0	0	0	1
Ptop	0	0	0	0	0	0	0	1

Tableau 4 (Après SWAP)

(DegBot=5 - DegTop=6 - ratio= α - SWAP=0)

20

Rang	7	6	5	4	3	2	1	0
RBOT	α^4	α^{10}	α^3	α^7	α^9	α^{12}	0	α
Pbot	0	0	0	0	0	0	1	1
RTOP	α^{11}	α^4	α^8	α^{10}	α^{13}	0	1	0
Ptop	0	0	0	0	0	0	1	1

Tableau 5 (Itération 1)

(DegBot=5 - DegTop=5 - ratio= α^{12} - SWAP=0)

Rang	7	6	5	4	3	2	1	0
RBOT	α^4	α^{10}	α^3	α^7	α^9	α^{12}	α^{12}	α
Pbot	0	0	0	0	0	0	1	1
RTOP	α^3	α^2	α^2	1	α^9	1	0	0
Ptop	0	0	0	0	0	1	1	1

5

Tableau 6 (Itération 2)

(DegBot=5 - DegTop=4 - ratio= α^3 - SWAP=1)

Rang	7	6	5	4	3	2	1	0
RBOT	α^3	α^2	α^2	1	α^9	1	0	0
Pbot	0	0	0	0	0	1	1	1
RTOP	α^4	α^{10}	α^3	α^7	α^9	α^{12}	α^{12}	α
Ptop	0	0	0	0	0	0	1	1

Tableau 7 (Après SWAP)

(DegBot=4 - DegTop=5 - ratio= α^{11} - SWAP=0)

10

Rang	7	6	5	4	3	2	1	0
RBOT	α^3	α^2	α^2	1	α^9	1	α^8	α^{12}
Pbot	0	0	0	0	0	1	1	1
RTOP	α^9	α^8	α^8	α^6	α^{12}	α^{12}	α	0
Ptop	0	0	0	0	0	1	1	1

Tableau 8 (Itération 3)

(DegBot=4 - DegTop=4 - ratio= α^6 - SWAP=0)

Rang	7	6	5	4	3	2	1	0
RBOT	α^3	α^2	α^2	1	α^9	α^{14}	α^{11}	α^{12}
Pbot	0	0	0	0	0	1	1	1
RTOP	0	0	0	α^{11}	α^{12}	α	0	0
Ptop	0	0	0	0	1	1	1	1

Tableau 9 (Itération 4)

(DegBot=4 - DegTop=3 - ratio=0 - SWAP=0)

15

Rang	7	6	5	4	3	2	1	0
RBOT	α^3	α^2	α^2	1	α^9	α^{14}	α^{11}	α^{12}
Pbot	0	0	0	0	0	1	1	1
RTOP	0	0	α^{11}	α^{12}	α	0	0	0
Ptop	0	0	0	1	1	1	1	1

Tableau 10 (Itération 5)

(DegBot=4 - DegTop=2 - ratio=0 - SWAP=0)

5 On obtient donc finalement :

$$\Lambda(z) = \alpha^{12} z^2 + \alpha^{11} z + \alpha^{14}$$

$$\Omega(z) = \alpha^{11}$$

10 Les coefficients de $\Lambda(z)$ sont stockés dans la partie Lambda de RBOT, tandis que les coefficients de $\Omega(z)$ sont stockés dans la partie Omega de RTOP.

Après le calcul de $\Lambda(z)$ et de $\Omega(z)$, il faut ajuster les coefficients de $\Omega(z)$ en rebouclant RTOP(r+1) sur RTOP(2) et en effectuant $m=t+(\text{degré}E(z)/2)$ rotations.

15

Détermination de T(z) - syndrome modifié:

20 Avantageusement, le résolveur d'équations est également utilisé pour multiplier les polynômes E(z) et S(z) pour le calcul du syndrome modifié T(z). Le résolveur d'équations dispose déjà des multiplieurs nécessaires, car la détermination du polynôme $\Lambda(z)$ requiert une multiplication de polynômes. On met à profit ces ressources pour ce qui est de la multiplication $E(z)*S(z)$, ce qui simplifie la constitution du générateur 2.

25 Le degré maximal de S(z) est $2t-1=r-1$, tandis que celui de E(z) est $2t=r$ par construction (si le degré de E(z) est supérieur à $2t$, le mot de code est marqué comme non corrigible). Le degré maximal de T(z) est donc $r+1$, ce qui correspond exactement au nombre de cellules du résolveur d'équations.

30 La figure 14 illustre la configuration d'une cellule pour la mise en oeuvre d'une multiplication. Les registres Ptop et Pbot servent à déterminer le contenu des registres RTOP(k) et RBOT(k). La configuration des cellules est

assez proche de celle illustrée par le diagramme de la figure 12 ($P_{top}=P_{bot}=1$), c'est à dire la configuration Lambda. Chaque registre RBOT reboucle sur son entrée à travers l'additionneur 108 associé, tandis que l'entrée de chaque registre RTOP est connecté à la sortie du registre RTOP de la cellule précédente. En dernier lieu, l'ensemble des multiplieurs des cellules sont connectés à une même entrée E, en vue de l'introduction des coefficients en série d'un des deux polynômes à multiplier. Dans le cadre du présent exemple, ce sont les coefficients de $E(z)$ qui sont introduits par ce biais, mais on pourrait également introduire les coefficients de $S(z)$. L'entrée E servait dans la configuration utilisée pour la détermination de $\Lambda(z)$ et de $\Omega(z)$ pour fournir la valeur de 'ratio' à chaque cellule.

Une autre différence avec la configuration Lambda de la figure 12 est que l'état du circuit inverseur 106 demeure identique tout au long de la multiplication.

15

Le principe de fonctionnement est le suivant :

Dans un premier temps, les coefficients de $S(z)$ sont stockés dans les registres RTOP. Un '0' est présenté à l'entrée du premier registre RTOP(0).

Dans un second temps, les coefficients E_i de $E(z)$ sont présentés par ordre d'indice croissant à l'entrée E, à raison d'un coefficient par cycle d'horloge.

Le nombre de cycles d'horloge nécessaires pour effectuer la multiplication est égal au degré de $E(z)$.

A titre d'exemple, le tableau 11 donne le contenu de chacun des registres dans le cas d'une multiplication de $S(z)$ et de $E(z)$ où $S(z)$ est de degré 3 et $E(z)$ de degré 3.

<u>Cycle d'Horloge</u>	<u>Cellule</u>	4	3	2	1
	<u>Registre</u>				
	RBOT	0	0	0	0
	RTOP	S_3	S_2	S_1	S_0
<u>1</u>	RBOT	S_3E_0	S_2E_0	S_1E_0	S_0E_0
	RTOP	S_2	S_1	S_0	0
<u>2</u>	RBOT	$S_3E_0+S_2E_1$	$S_2E_0+S_1E_1$	$S_1E_0+S_0E_1$	S_0E_0
	RTOP	S_1	S_0	0	0
<u>3</u>	RBOT	$S_3E_0+S_2E_1+S_1E_2$	$S_2E_0+S_1E_1+S_0E_2$	$S_1E_0+S_0E_1$	S_0E_0
	RTOP	S_0	0	0	0
<u>4</u>	RBOT	$S_3E_0+S_2E_1+S_1E_2+S_0E_3$	$S_2E_0+S_1E_1+S_0E_2$	$S_1E_0+S_0E_1$	S_0E_0
	RTOP	0	0	0	0

Tableau 11

On voit que les registres RTOP(k) forment un registre à décalage servant uniquement à stocker les coefficients de S(z). Le registre à décalage est réalisé en choisissant correctement l'état des inverseurs 106. Les registres RBOT(k) servent d'accumulateurs respectifs pour chaque coefficient du polynôme T(z).

Chaque registre RBOT(k) contient donc finalement le coefficient T_k , où $T(z) = \sum_{k=1}^r T_k z^k$.

10 Détermination du nouveau polynôme de localisation d'erreur $\Psi(z)$

Avantageusement, le résolveur d'équations est également utilisé pour multiplier les polynômes E(z) et $\Lambda(z)$ pour le calcul de $\Psi(z)$. Ce dernier polynôme est calculé une fois $\Lambda(z)$ déterminé par le résolveur d'équations. Une fois que l'algorithme étendu d'Euclide a été mis en oeuvre, les coefficients de $\Lambda(z)$ sont stockés dans la partie Lambda de RBOT. Les autres registres RTOP(k) et RBOT(k) sont mis à zéro.

La configuration des cellules pour réaliser la multiplication est illustrée par le diagramme de la figure 15.

Les registres RTOP(k) sont connectés en série à travers les additionneurs des cellules. La valeur '0' est présentée à l'entrée du premier registre RTOP(0). La sortie de chaque registre RBOT(k) est connectée au multiplieur de sa cellule k. Les valeurs stockées dans les registres RBOT(k) ne changent pas pendant la présente multiplication. Comme dans la configuration de la figure 14, les entrées des multiplieurs autres que celles connectées à la sortie des registres RBOT(k) sont interconnectées (référence E).

Le degré de $\Lambda(z)$ est égal au nombre d'erreurs. Le degré de E(z) est égal au nombre d'effacements.

Le tableau 12 donne le contenu des cellules après chaque cycle d'horloge.

Cycle	Cellule	...	7	6	5	4	3	2	1	0
	Registre									
	RBOT	0	0	0	0	0	0	Λ_0	Λ_1	Λ_2
	RTOP	0	0	0	0	0	0	0	0	0
<u>1</u>	RBOT	0	0	0	0	0	0	Λ_0	Λ_1	Λ_2
	RTOP	0	0	0	0	0	$\Lambda_0 E_0$	$\Lambda_1 E_0$	$\Lambda_2 E_0$	0
<u>2</u>	RBOT	0	0	0	0	0	0	Λ_0	Λ_1	Λ_2
	RTOP	0	0	0	0	$\Lambda_0 E_0$	$\Lambda_1 E_0 + \Lambda_0 E_1$	$\Lambda_2 E_0 + \Lambda_1 E_1$	$\Lambda_2 E_1$	0
<u>3</u>	RBOT	0	0	0	0	0	0	Λ_0	Λ_1	Λ_2
	RTOP	0	0	0	$\Lambda_0 E_0$	$\Lambda_1 E_0 + \Lambda_0 E_1$	$\Lambda_2 E_0 + \Lambda_1 E_1 + \Lambda_0 E_2$	$\Lambda_2 E_1 + \Lambda_1 E_2$	$\Lambda_2 E_2$	0
<u>4</u>	RBOT	0	0	0	0	0	0	Λ_0	Λ_1	Λ_2
	RTOP	0	0	$\Lambda_0 E_0$	$\Lambda_1 E_0 + \Lambda_0 E_1$	$\Lambda_2 E_0 + \Lambda_1 E_1 + \Lambda_0 E_2$	$\Lambda_2 E_1 + \Lambda_1 E_2 + \Lambda_0 E_3$	$\Lambda_2 E_2 + \Lambda_1 E_3$	$\Lambda_2 E_3$	0

Tableau 12

Pendant le calcul de $\Psi(z)$, les multiplieurs restant actifs sont ceux des cellules pour lesquelles $P_{bot}(k)=1$.

5 Bien que le présent exemple de réalisation concerne la correction Reed-Solomon, la configuration de deux mémoires décrite peut être appliquée dans tous les cas où des paquets de données de longueur sensiblement différente doivent être corrigés.

10 En dernier lieu, la figure 16 représente un lecteur 200 de disque optique 201 comportent le dispositif conforme à l'invention. Ce lecteur comporte un bloc optique 202 comportant de manière connue un laser, des moyens de focalisation, des détecteurs photoélectriques. Le lecteur comporte également un moteur 203 commandé par une unité d'asservissement 204. Les données lues par le bloc optique sont transmises à un démultiplexeur 205. Les données de contrôle sont transmises à un microprocesseur 206 commandant un affichage 207, l'asservissement de la focalisation 208 et le suivi de piste 209. Un clavier 210 permet à l'utilisateur de commander le lecteur.

15 Les paquets de données CD/DVD proprement dits sont transmis par le démultiplexeur 205 au dispositif de décodage 1 relié à la mémoire 5 et à une unité d'interpolation 211 qui fait office d'interface avec des convertisseurs numérique/analogique non illustrés.

Bien sûr, si un autre algorithme que l'algorithme étendu d'Euclide est utilisé, et que cet algorithme nécessite aussi des multiplications de polynômes, l'invention peut également s'appliquer.

Revendications

1. Dispositif (1) de correction d'erreurs de type Reed-Solomon comportant :

- 5 - un générateur de syndromes (2),
 - des moyens (3) de détermination des coefficients des polynômes de localisation et d'évaluation d'erreur,
 - des moyens (4) de détermination de la position et de la valeur d'erreurs, lesdits moyens (3) de détermination des coefficients étant aptes à
10 mettre en oeuvre l'algorithme étendu d'Euclide,
 caractérisé en ce que
 les moyens (3) de détermination des coefficients sont agencés pour réaliser le produit du polynôme de localisation d'erreur ($\Lambda(z)$), et d'un polynôme d'effacement ($E(z)$) pour l'obtention d'un nouveau polynôme de localisation
15 d'erreur ($\Psi(z)$).

2. Dispositif selon la revendication 1, caractérisé en ce que les moyens (3) de détermination des coefficients sont agencés pour réaliser le produit d'un polynôme de syndromes ($S(z)$) par le polynôme d'effacement
20 ($E(z)$).

3. Dispositif selon l'une des revendications 1 ou 2, caractérisé en ce que les moyens (3) de détermination des coefficients comportent $r+1$ cellules en série, r étant le nombre maximum de symboles redondants, chaque cellule
25 comportant un premier registre (RTOP(k), 105) pour le stockage d'un coefficient de polynôme, un second registre (RBOT(k), 104) pour le stockage d'un coefficient de polynôme, un circuit (106) pour connecter le premier registre à une première entrée d'un additionneur (108) et le second registre à une première entrée d'un multiplieur (109) ou inversement, l'entrée du premier
30 registre (105) étant connectée à une sortie d'une cellule précédente, l'entrée du second registre (104) étant connectée à des moyens de multiplexage (112, 113) dont les entrées sont connectées à la sortie du premier registre et du second registre de la cellule précédente et à la sortie de l'additionneur (108), la sortie de la cellule étant constituée par un multiplexeur (111) dont les entrées
35 sont connectées à la sortie de l'additionneur et à la sortie du circuit (106) reliée à la première entrée du multiplieur, une seconde entrée de l'additionneur (108)

étant connectée à la sortie du multiplieur (110), une seconde entrée du multiplieur (110) étant apte à recevoir un coefficient de multiplication.

5 4. Dispositif selon la revendication 3, caractérisé en ce que les secondes entrées des multiplieurs des cellules sont connectées.

10 5. Dispositif selon les revendications 2 à 4, caractérisé en ce que pour la multiplication des polynômes d'effacement ($E(z)$) et de syndromes ($S(z)$), la sortie du premier registre (RTOP(k), 105) d'une cellule est connectée au multiplieur (110), la sortie du second registre (RBOT(k), 104) est connectée à l'additionneur (108), la sortie de la cellule étant constituée par la première entrée du multiplieur, l'entrée du second registre (RBOT(k), 104) étant connectée à la sortie de l'additionneur (108).

15 6. Dispositif selon la revendication 5, caractérisé en ce que les coefficients d'un des deux polynômes à multiplier sont stockés dans les premiers registres des cellules (RTOP(k), 105) en phase d'initialisation, tandis que les coefficients de l'autre polynôme sont présentés un à un à la seconde entrée du multiplieur (110).

20 7. Dispositif selon les revendications 1, 3 et 4, caractérisé en ce que pour la multiplication des polynômes de localisation d'erreur ($\Lambda(z)$) et d'effacement ($E(z)$), le premier registre (RTOP(k), 105) d'une cellule est connecté à la sortie de l'additionneur de la cellule précédente (k-1), la sortie du premier registre étant connectée à la première entrée de l'additionneur (108), la sortie du second registre (RBOT(k), 104) étant connectée à la première entrée du multiplieur.

30 8. Dispositif selon la revendication 7, caractérisé en ce que les coefficients du polynôme de localisation ($\Lambda(z)$) sont stockés dans les seconds registres (RBOT(k)) des cellules en phase d'initialisation, le contenu de ces second registres n'évoluant pas lors de la multiplication, les coefficients du polynôme d'effacement ($E(z)$) étant présentés un à un aux secondes entrées des multiplieurs (110) des cellules.

35 9. Lecteur (200) de disques optiques (201) caractérisé en ce qu'il comporte un dispositif selon l'une des revendications précédentes.

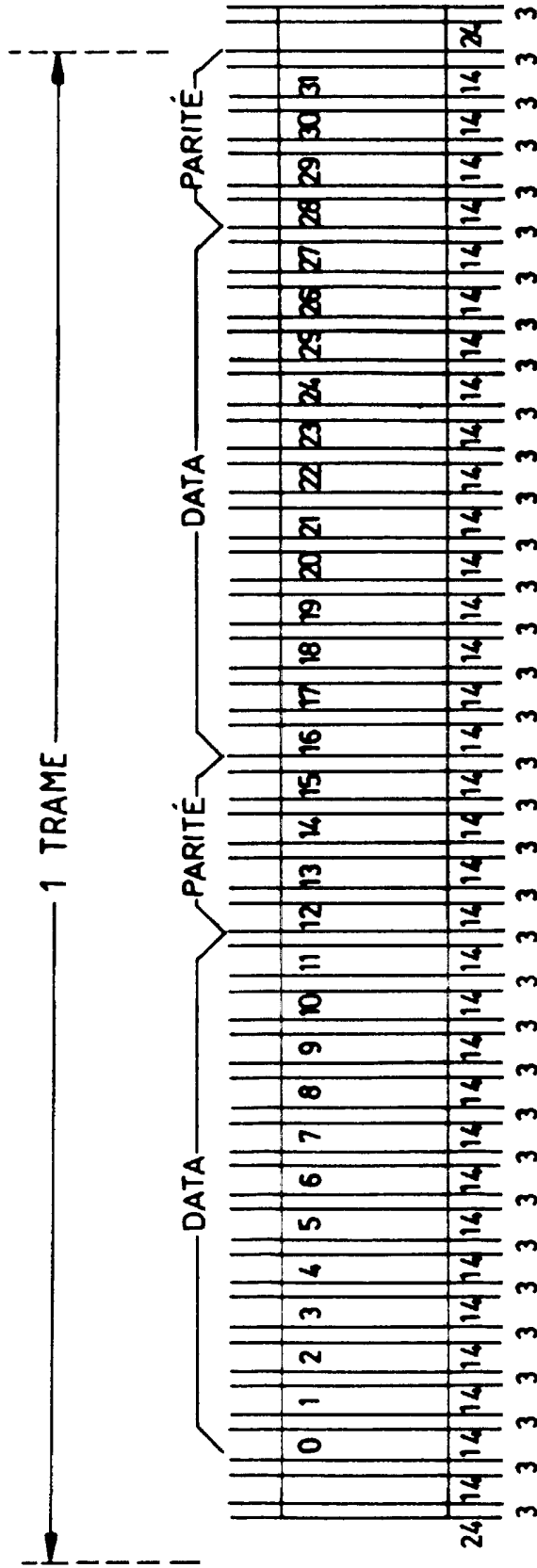


FIG.1

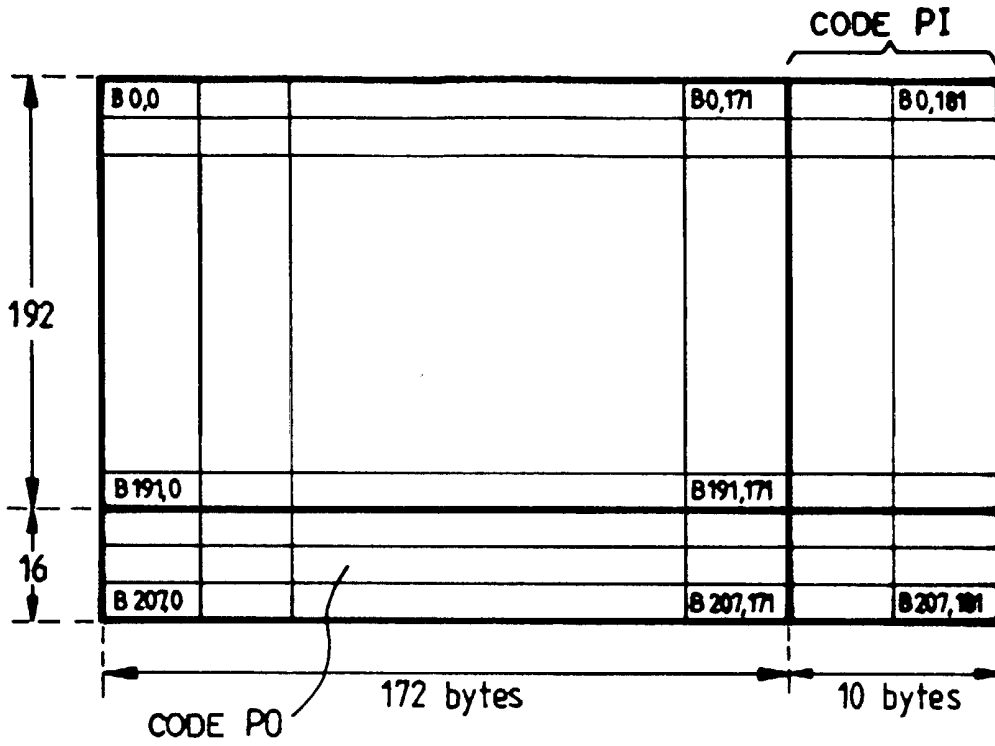


FIG. 2

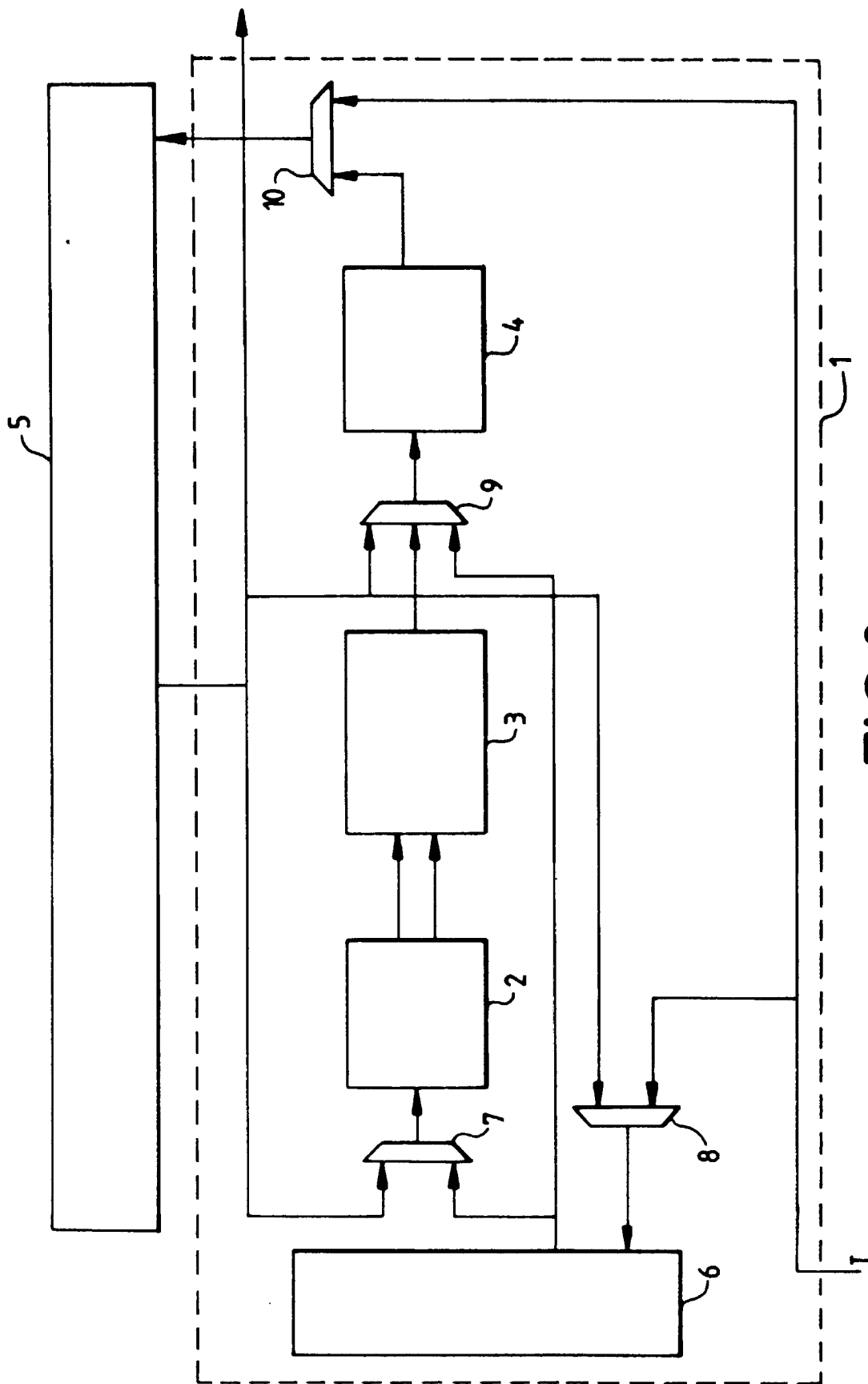


FIG.3

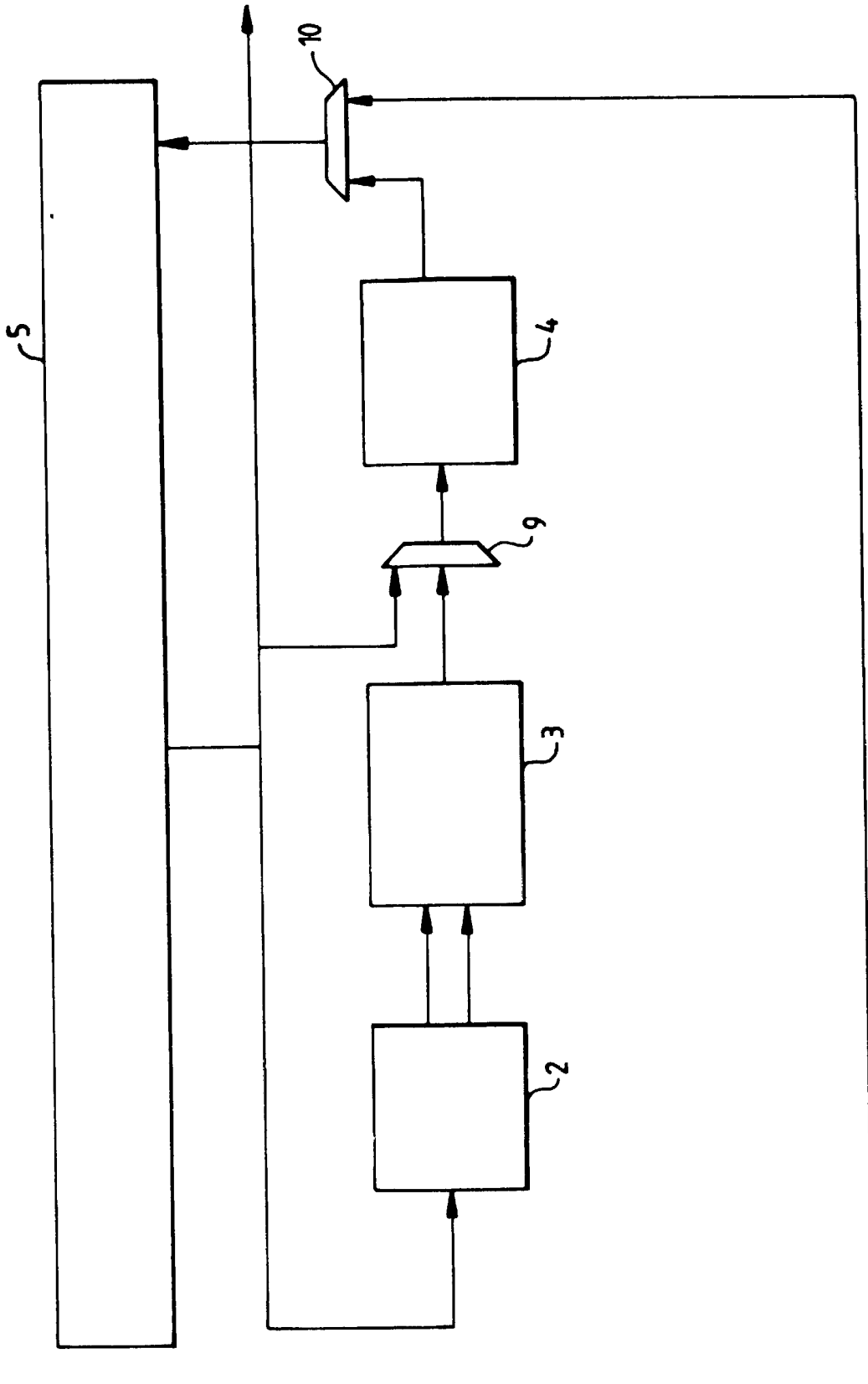


FIG.4

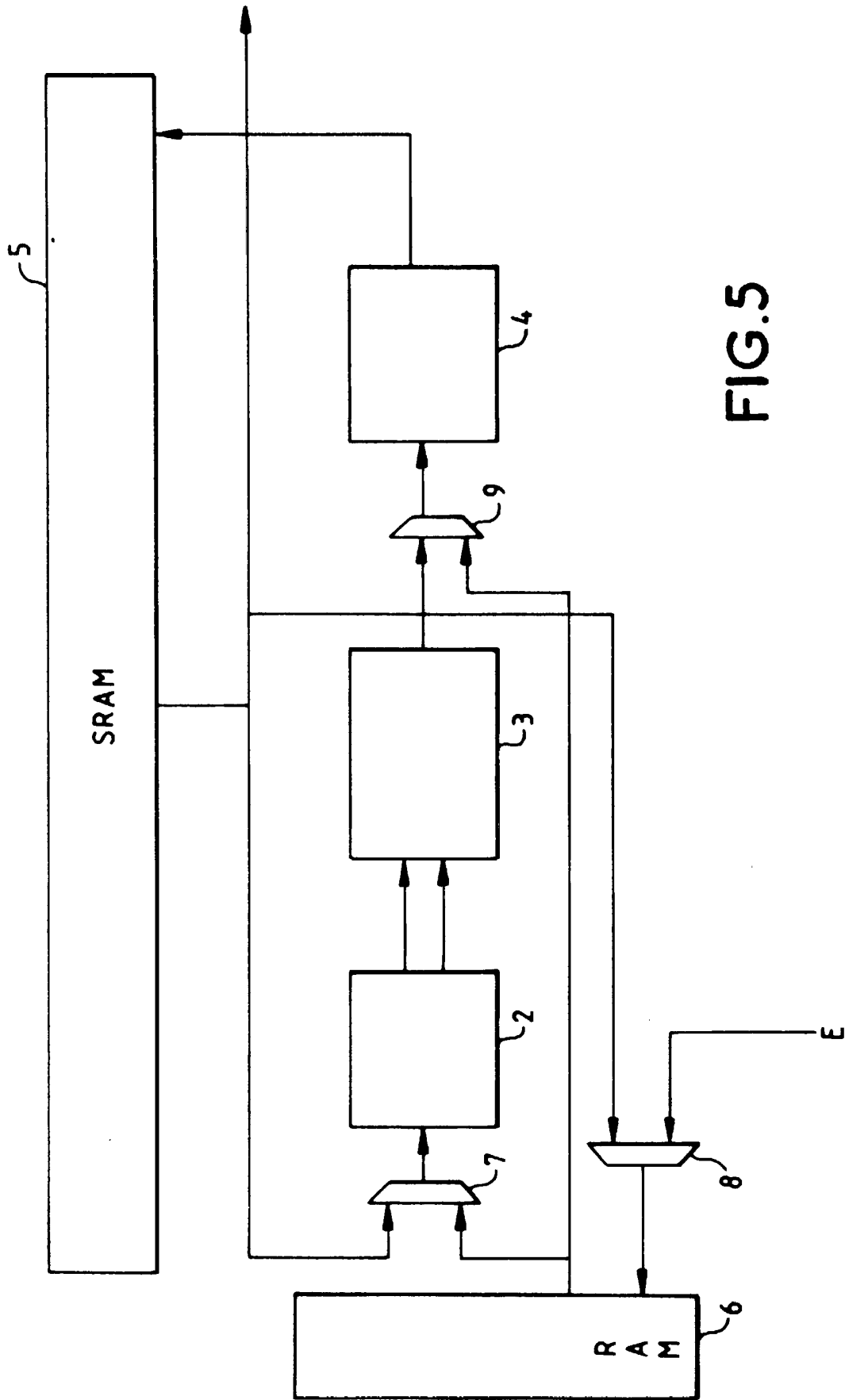


FIG. 5

FIG. 6

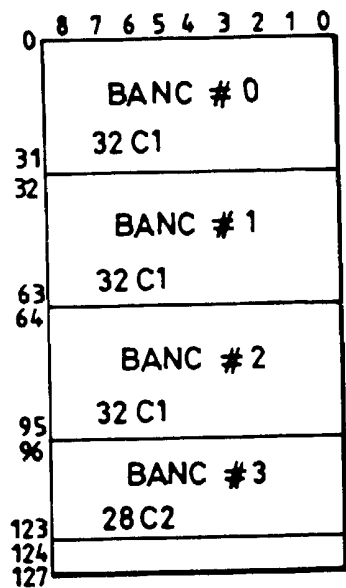
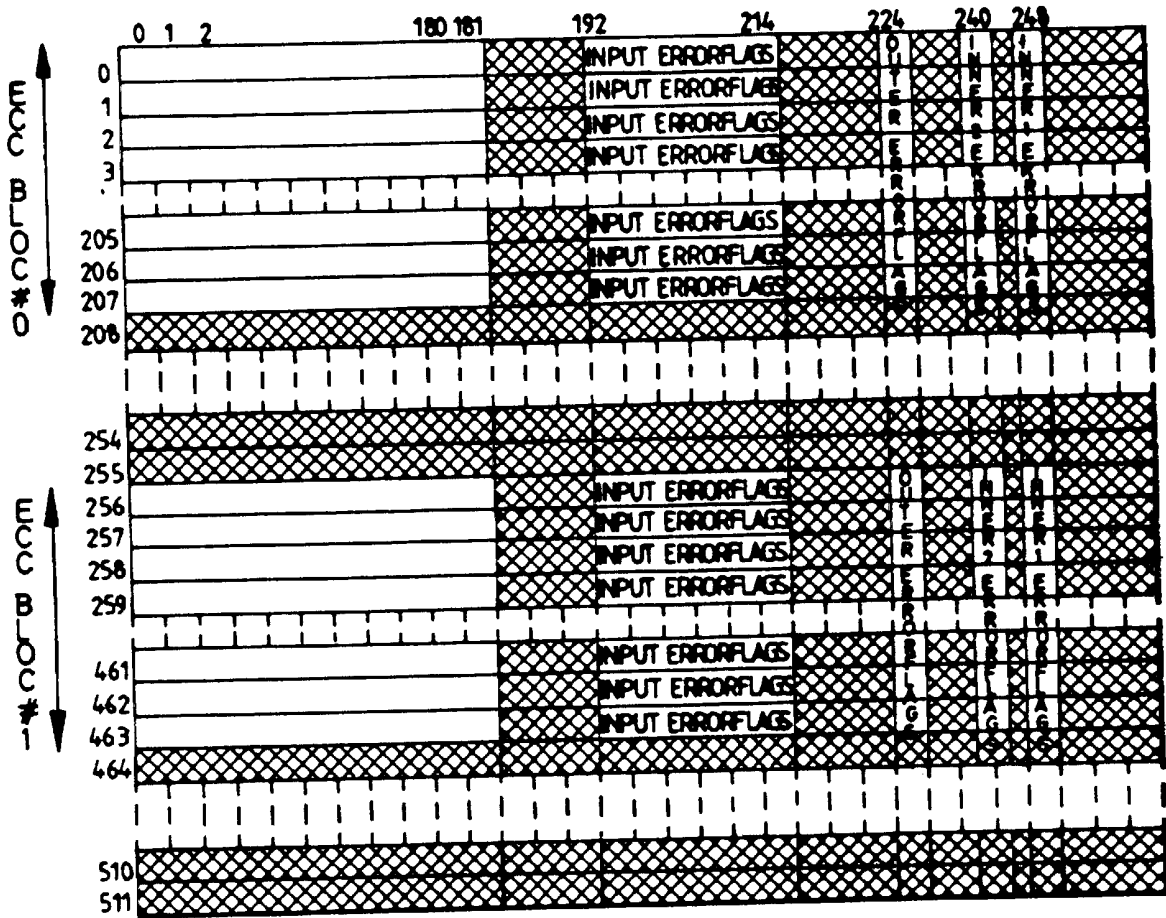


FIG. 7

4 OCTETS:
 124: "SUBCODE ODD FRAME"
 125: "S1/S0 REGISTER ODD FRAME"
 126: "SUBCODE EVEN FRAME"
 127: "S1/S0 REGISTER EVEN FRAME"

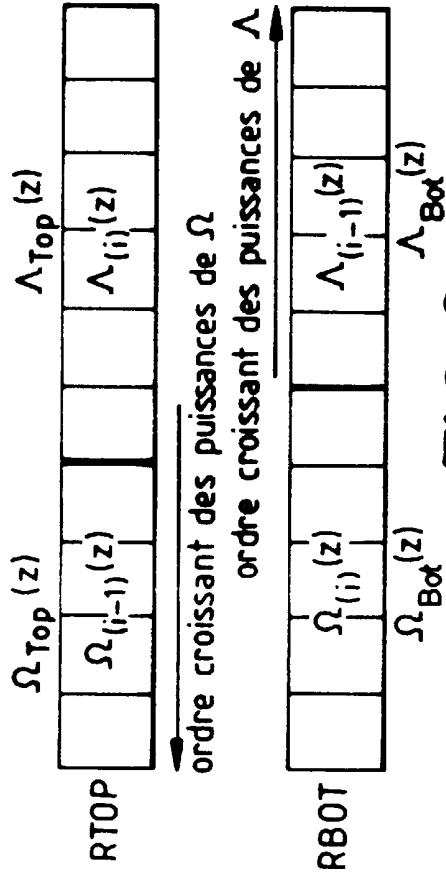


FIG. 8

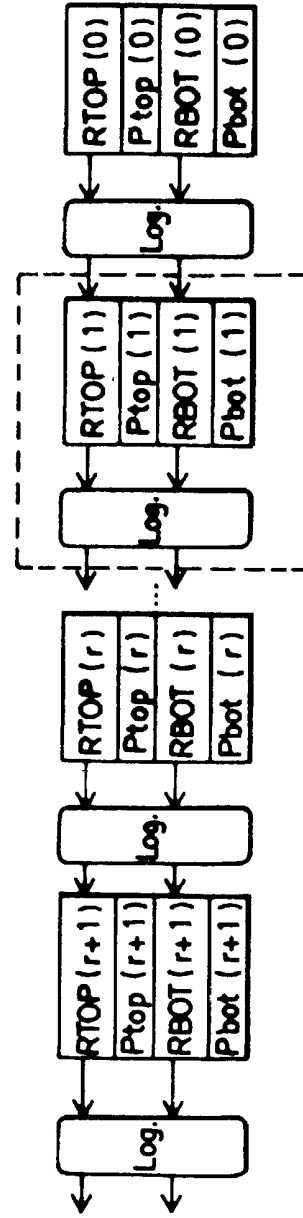
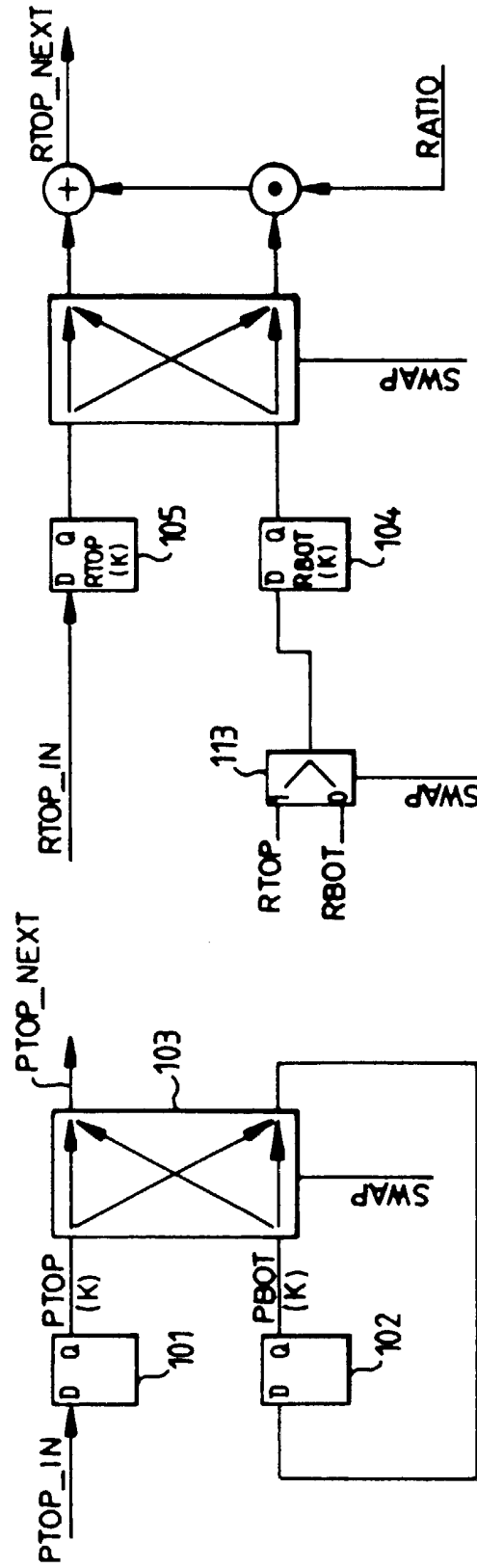


FIG. 9



$P_{TOP} = P_{BOT} = 0$

FIG.11

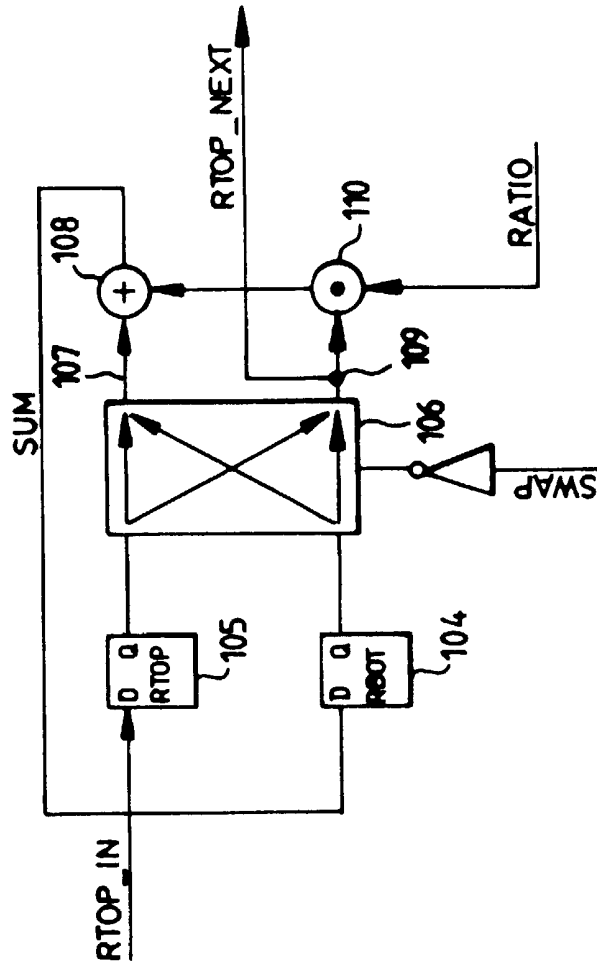
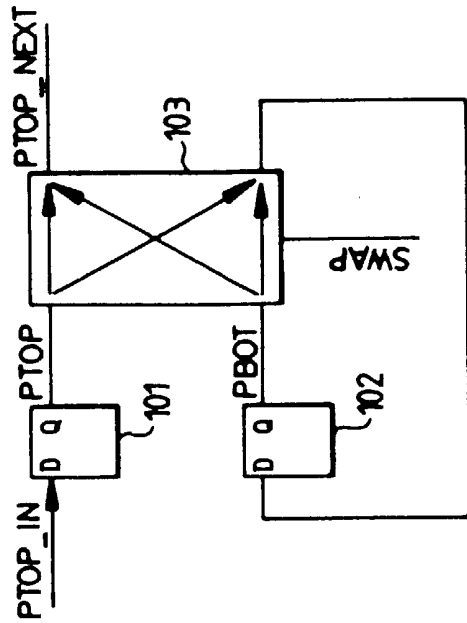


FIG.12



$PTOP = PBOT = 1$

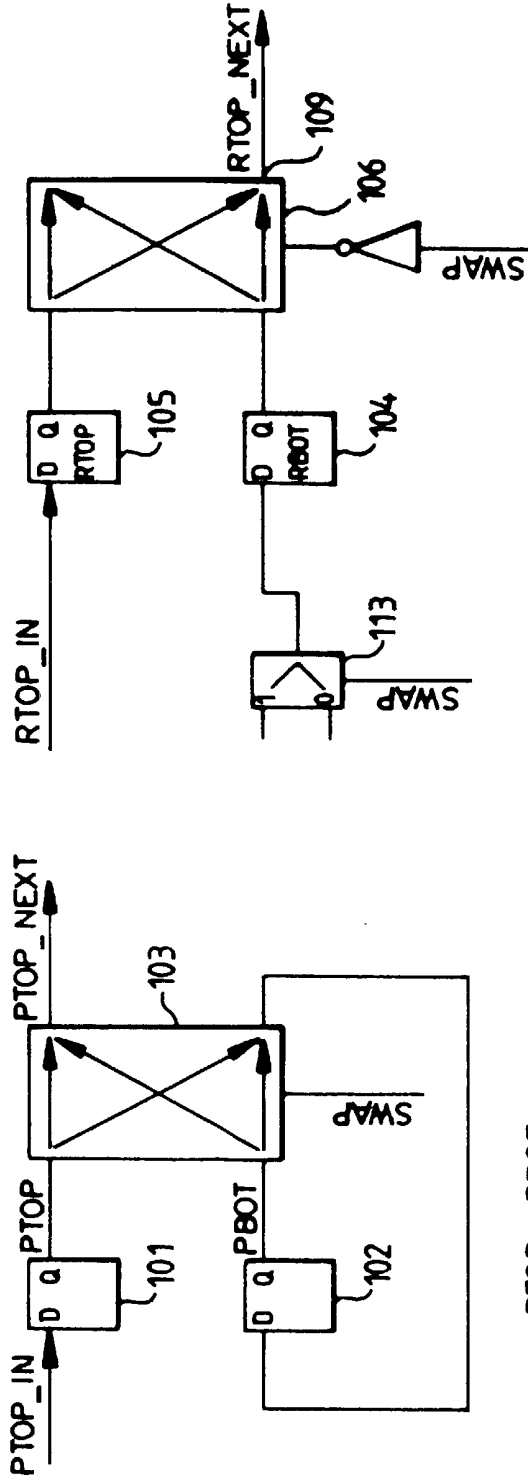


FIG.13

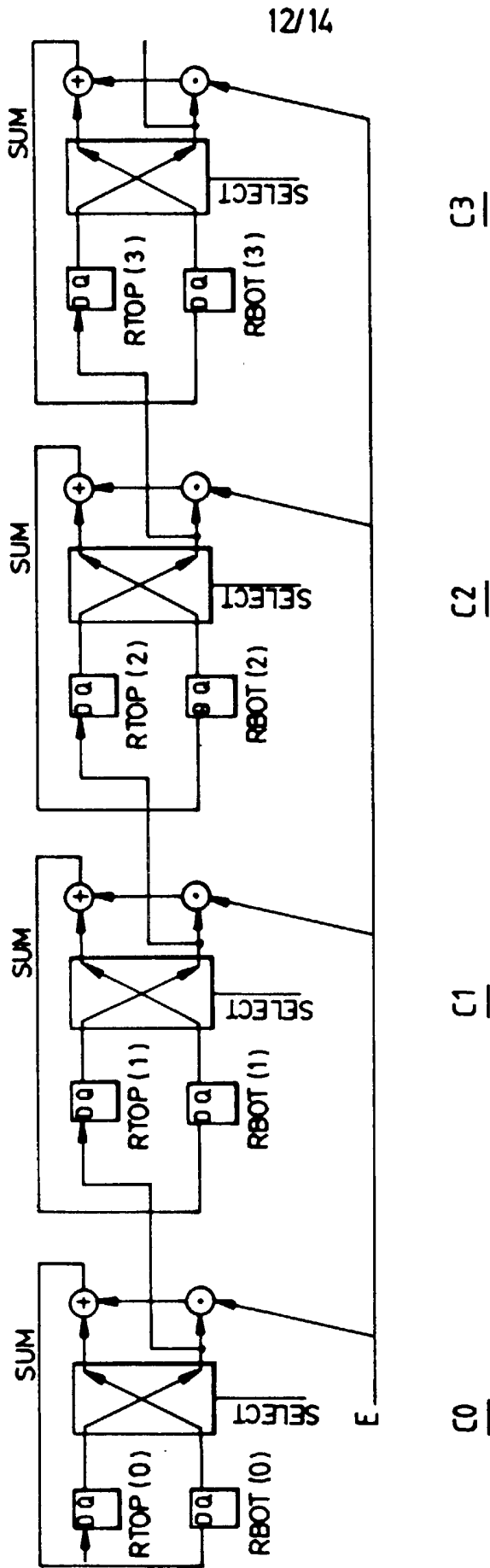


FIG.14

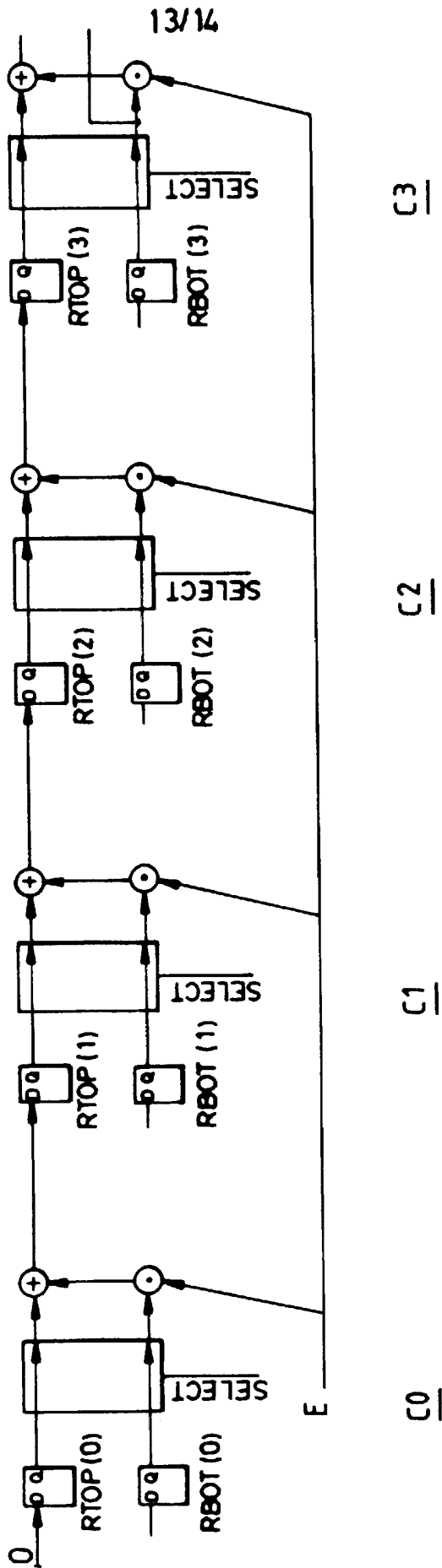


FIG.15

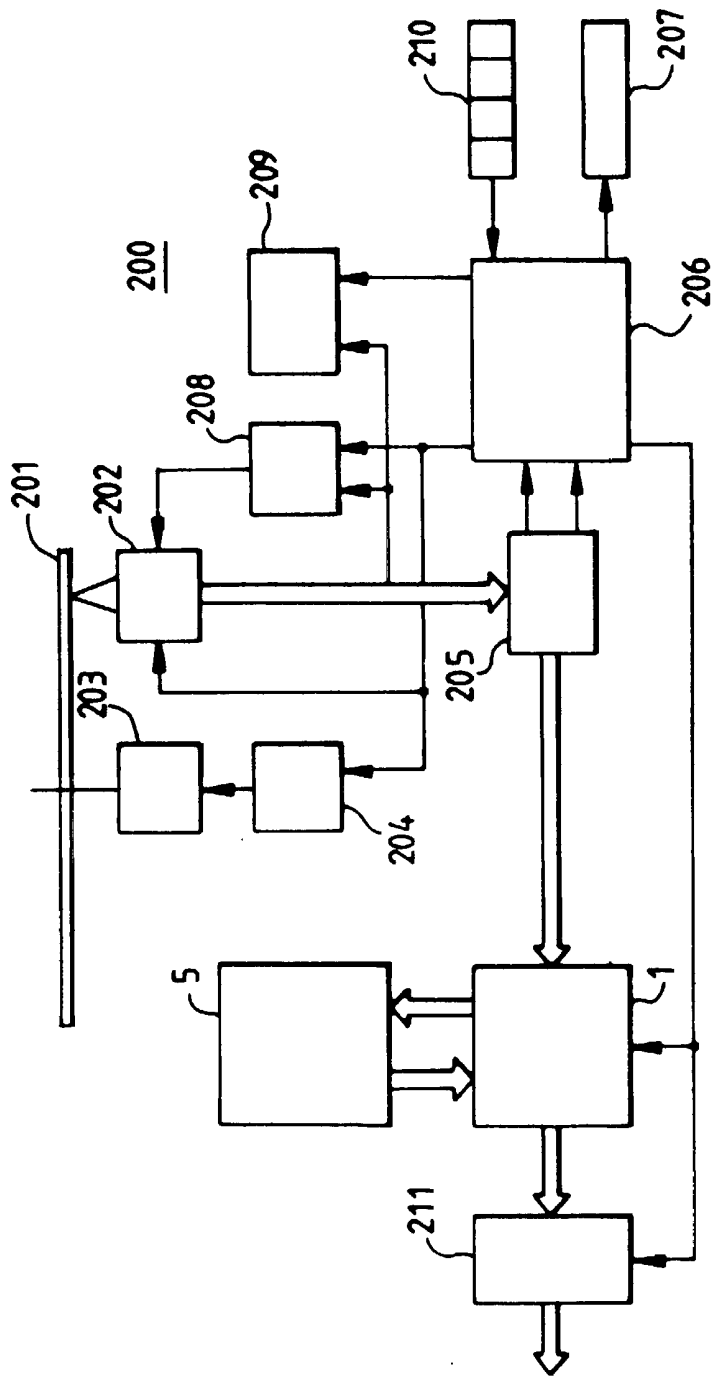


FIG.16

RAPPORT DE RECHERCHE
PRELIMINAIRE

établi sur la base des dernières revendications
déposées avant le commencement de la recherche

N° d'enregistrement
national

FA 558592
FR 9803103

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, des parties pertinentes	
X	EP 0 620 654 A (TOSHIBA) 19 octobre 1994 * figures 1-3,14 * * page 13, ligne 21 - ligne 52 * * page 3, ligne 47 - page 6, ligne 23 *	1-5,9
X	US 5 170 399 A (CAMERON ET AL.) 8 décembre 1992 * colonne 2, ligne 58 - colonne 3, ligne 60 *	1,2,9
X	US 5 715 262 A (GUPTA) 3 février 1998 * colonne 1, ligne 22 - colonne 2, ligne 4 * * colonne 4, ligne 4 - colonne 5, ligne 63 *	1,2,9
X	US 5 642 367 A (KAO) 24 juin 1997 * colonne 12, ligne 27 - colonne 14, ligne 13 *	1,2,9
X	US 4 868 828 A (SHAO ET AL.) 19 septembre 1989 * colonne 5, ligne 9 - colonne 8, ligne 29 *	1,2,9
		DOMAINES TECHNIQUES RECHERCHES (Int.CL.6)
		H03M G11B
Date d'achèvement de la recherche		Examineur
1 octobre 1998		Augarde, E
CATEGORIE DES DOCUMENTS CITES		
<p>X : particulièrement pertinent à lui seul Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général O : divulgation non-écrite P : document intercalaire</p>		
<p>T : théorie ou principe à la base de l'invention E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure. D : cité dans la demande L : cité pour d'autres raisons & : membre de la même famille, document correspondant</p>		