

①9 RÉPUBLIQUE FRANÇAISE  
**INSTITUT NATIONAL  
 DE LA PROPRIÉTÉ INDUSTRIELLE**  
 PARIS

①1 N° de publication :  
 (à n'utiliser que pour les  
 commandes de reproduction)

**2 722 041**

②1 N° d'enregistrement national :

**95 03690**

⑤1 Int Cl<sup>6</sup> : H 03 M 7/42

⑫

**DEMANDE DE BREVET D'INVENTION**

**A1**

②2 Date de dépôt : 29.03.95.

③0 Priorité : 30.06.94 KR 9415677.

④3 Date de la mise à disposition du public de la demande : 05.01.96 Bulletin 96/01.

⑤6 Liste des documents cités dans le rapport de recherche préliminaire : *Ce dernier n'a pas été établi à la date de publication de la demande.*

⑥0 Références à d'autres documents nationaux apparentés :

⑦1 Demandeur(s) : **SAMSUNG ELECTRONICS CO LTD**  
 — KR.

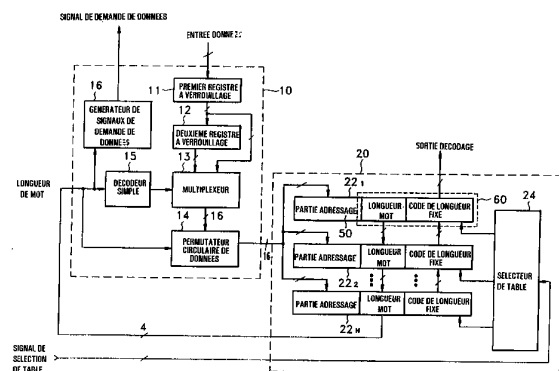
⑦2 Inventeur(s) : **PARK HEON CHUL.**

⑦3 Titulaire(s) :

⑦4 Mandataire : **SOCIETE DE PROTECTION DES INVENTIONS.**

⑤4 **DECODEUR DE HUFFMAN.**

⑤7 Décodeur de Huffman comprenant des registres à verrouillage (11, 12) mémorisant et conservant respectivement les données codées en longueur variable respectivement actuelles et précédentes, un décodeur simple (15) recevant la longueur de mot et produisant un signal de sélection, un générateur de signaux de demande de données (16) produisant un signal lorsque la longueur de mot additionnée dépasse une longueur prédéterminée, un multiplexeur (13) recevant les sorties desdits registres à verrouillage (11, 12) pour sélectionner un bit prédéterminé en fonction du signal de sélection, un permutateur circulaire de données (14) effectuant le décalage circulaire de la sortie du multiplexeur pour former et sortir des données reconstituées, une partie adressage (50) pour comparer ces dernières à une valeur prédéterminée et les decoder, et une mémoire (60) conservant la longueur du mot codé en longueur variable et le code de longueur fixe correspondant, sortis en fonction de la sortie de ladite partie.



FR 2 722 041 - A1



**DECODEUR DE HUFFMAN****Arrière-plan de l'Invention**

5 La présente invention concerne un décodeur de Huffman en tant que décodeur de longueur variable pour comprimer un signal image et, plus particulièrement, un décodeur de Huffman d'un type table à consulter, dans lequel la taille d'une mémoire de recherche est plus petite que jamais.

10 Grâce aux algorithmes de compression, tels que MPEG, JPEG, pX64 (H.261) et analogues, en tant que normes internationales pour multifonctions, un signal image converti DCT est codé en longueur variable pour la transmission. On utilise, par conséquent, un  
15 décodage de Huffman à l'extrémité réceptrice pour décoder le code de longueur variable en code d'origine de longueur fixe. Le décodeur classique de Huffman pour multifonctions fait appel à des procédés tels que la recherche arborescente et la consultation de tables.  
20 Ici, le décodage de Huffman, en tant que sorte de décodage de longueur variable, reçoit un code de longueur variable pour rétablir un code de longueur fixe.

25 Par exemple, le code de longueur variable entré pour un mot de sortie de longueur fixe est, en général, composé d'environ 2 à 24 bits. Théoriquement, la longueur d'une suite d'entrée pour un mot de sortie de longueur fixe de  $n$  bits peut être exprimé par 2 à  $2^n - 1$  bits.

30 Dans le décodage classique de Huffman, lorsque l'on utilise le procédé de recherche arborescente, la taille de la mémoire de recherche requise pour la conversion de code peut être minimisée. Toutefois, le temps de recherche est relativement long à cause des  
35 caractéristiques de la recherche arborescente. Par conséquent, il est difficile d'appliquer le procédé de recherche arborescente à un appareil demandant un

traitement rapide, par exemple la télévision numérique, la télévision à haute définition ou analogues.

Si l'on utilise, par contre, le procédé de consultation de tables, le temps de recherche peut être  
5 réduit. Il se pose toutefois un problème, en ce que la surface de la puce est augmentée lors de la fabrication des circuits VLSI à cause de l'augmentation de la taille de la mémoire pour une consultation de tables.

#### Sommaire de l'Invention

10 Afin de résoudre les problèmes ci-dessus mentionnés, un but de la présente invention consiste à proposer un décodeur de Huffman qui adopte le procédé de consultation de tables tout en réduisant la taille de la mémoire à consulter.

15 Il est proposé selon la présente invention, pour atteindre ce but, un décodeur de Huffman destiné à recevoir des données codées en longueur variable et à produire un code de longueur fixe correspondant aux données reconstituées dans une table de Huffman choisie  
20 en fonction d'un signal de sélection de table, le décodeur de Huffman comprenant : un premier registre à verrouillage pour conserver les données actuelles codées en longueur variable ; un deuxième registre à verrouillage pour conserver les données précédentes  
25 codées en longueur variable ; un décodeur simple pour recevoir la longueur de mot et pour produire un signal de sélection ; un générateur de signaux de demande de données pour produire un signal de demande de données lorsque la longueur de mot additionnée dépasse une  
30 longueur prédéterminée après la sommation des longueurs de mot ; un multiplexeur pour recevoir les sorties des premier et deuxième registres à verrouillage, afin de sélectionner un bit prédéterminé en fonction du signal de sélection ; un permutateur circulaire de données  
35 pour effectuer le décalage circulaire de la sortie du multiplexeur en fonction de la longueur de mot afin de former des données reconstituées et de sortir les

données reconstituées, une partie adressage pour comparer les données reconstituées à une valeur prédéterminée et pour décoder les données ; et une mémoire pour mémoriser la longueur du mot codé en longueur variable et le code de longueur fixe lui correspondant, qui sont produits en fonction de la sortie de la partie adressage.

#### Brève Description des Dessins

Les buts et avantages de la présente invention ressortiront mieux de la description détaillée, qui va suivre, d'un mode de réalisation préféré de celle-ci en se référant aux dessins annexés, sur lesquels :

la figure 1 est un schéma fonctionnel montrant un décodeur de Huffman d'un mode de réalisation préféré selon la présente invention ;

la figure 2 est un exemple de la table de Huffman, montrée à la figure 1 ;

la figure 3 est un exemple détaillé du cinquième comparateur et du cinquième décodeur binaire selon la figure 2 ; et

la figure 4 est un exemple de fonctionnement d'un permutateur circulaire de données selon la figure 1.

#### Description Détaillée de l'Invention

Afin de faciliter la compréhension d'un décodeur de Huffman, on va maintenant décrire quelques généralités du décodage de Huffman.

Dans la compression d'un signal image, un coefficient quantifié de transformée cosinus discrète (DCT) est davantage comprimé (compression de données) par l'intermédiaire d'un codage entropique, un code de longueur variable étant utilisé selon une caractéristique statistique. Il s'agit d'un procédé de codage sans perte et il est différent du procédé de quantification. Les codes entropiques comprennent les codes de Huffman, arithmétiques et universels. Le coefficient de DCT quantifié est codé en longueur variable (VLC) au moyen du codage de Huffman. Ici, la

partie codage en longueur variable classifie un coefficient de DCT en coefficients continus (DC) et en coefficients alternatifs (AC) et code ces deux types de coefficients au moyen de procédés différents. Etant  
5 donné que, en général, la valeur en continu de chaque bloc est en corrélation élevée avec la valeur en continu du bloc périphérique, on obtient une valeur différence par rapport à la valeur en continu du bloc précédent et la valeur différence est codée. Après  
10 l'obtention de la différence entre la valeur en continu du premier bloc et une valeur moyenne 128 de la plage variable des valeurs en continu, la valeur différence est codée par l'intermédiaire d'un code de longueur variable à une dimension. Les coefficients sont  
15 également remis en forme en tant qu'une dimension par l'intermédiaire d'un balayage en zigzag pour effectuer réellement la compression de données, avec la forte probabilité que la valeur du coefficient alternatif ne soit pas "0" à proximité du coefficient continu et  
20 qu'il soit "0" loin du coefficient continu, dans la zone DCT. Ici, le nombre de "0" consécutifs et la valeur du coefficient non "0" sont exprimés en deux dimensions, c'est-à-dire respectivement parcours et niveau. Par exemple, une forme de coefficient de DCT,  
25 telle que 30, 2, 0, 0, -8, 0, 0, 0, 9, ..., obtenus par un balayage en zigzag, est exprimée comme (0, 30), (0, 2), (2, -8) et (3, 9) par le codage parcours-niveau.

Le décodeur de Huffman d'un mode de réalisation préféré selon la présente invention, montré à la figure  
30 1, comprend une partie entrée de données 10 pour recevoir les données codées en longueur variable et leur faire effectuer une permutation circulaire, et une partie tables de Huffman 20 comportant N tables de Huffman, classées selon une caractéristique statistique  
35 des données.

Selon la figure 1, la partie entrée de données 10 comprend des premier et deuxième registres à

verrouillage 11 et 12, un multiplexeur 13, un permutateur circulaire de données 14, un décodeur simple 15 et un générateur de signaux de demande de données 16, pour demander les données à partir de la  
5 partie émission de données selon la valeur additionnée de longueurs de mots (lorsque la valeur additionnée dépasse seize selon un mode de réalisation préféré de la présente invention), afin de recevoir les données codées en longueur variable, pour procéder à une  
10 permutation circulaire des données en fonction de la longueur de mot et pour sortir les données permutées. La partie tables de Huffman 20 comprend un sélecteur de table 24 destiné à sélectionner une table spécifique en fonction d'un signal de sélection de table, et N tables  
15 de Huffman  $22_1$  à  $22_N$  pour produire le code de longueur fixe décodé et la longueur de mot correspondant à la sortie de la partie entrée de données 10. Chacune des tables de Huffman  $22_1$  à  $22_N$  est composée d'une partie adressage 50 et d'une partie mémoire 60 pour conserver  
20 la longueur de mot et le code de longueur fixe et pour sortir la longueur de mot et le code de longueur fixe, décodé en fonction des données entrées. Ici, le terme "longueur de mot" signifie la longueur de données en cours de décodage, et la valeur décodée est un code de  
25 longueur fixe en tant que valeur de sortie du décodeur de Huffman.

C'est-à-dire qu'un code de longueur fixe prédéterminé à produire et la longueur de la suite de données codées en longueur variable, donc la longueur  
30 de mot, sont mémorisés dans une table de Huffman et que l'un des comparateurs de la partie adressage 50 est sélectionné par les bits supérieurs d'un mot de code d'entrée. Les bits inférieurs du mot de code d'entrée sont fournis à un décodeur binaire correspondant au  
35 comparateur sélectionné, afin de choisir un emplacement de la table, et la contenu de cet emplacement est sorti du fait du décodage du code de longueur variable. En

utilisant l'information de longueur (longueur de mot) des données d'entrée mémorisées dans une table de ROM, la partie entrée de données 10 élimine autant de données actuelles que la longueur de mot et, en même temps, augmente les données entrées au premier registre à verrouillage 11 par une quantité équivalente.

Ici, le sélecteur de tables 24, servant à sélectionner une table lorsque l'on utilise une pluralité de tables de Huffman, est alimenté de l'extérieur au moyen d'un processeur central. Dans un mode de réalisation préféré de la présente invention, le processeur central a un bus de données à 16 bits.

Sur la figure 1, le premier registre à verrouillage 11 verrouille et mémorise les données d'entrée à 16 bits et le deuxième registre à verrouillage 12 mémorise les données d'entrée à 16 bits, précédemment mémorisées dans le premier registre à verrouillage 11. Le décodeur simple 15 reçoit la longueur de mot pour que le multiplexeur 13 sélectionne les données à 16 bits requises venant des premier et deuxième registres à verrouillage 11 et 12. C'est-à-dire que le multiplexeur 13 reçoit les sorties des premier et deuxième registres à verrouillage 11 et 12 en tant que groupes de bits supérieurs et inférieurs, respectivement, et qu'il sélectionne les données à 16 bits correspondantes en fonction de la sortie du décodeur simple 15 et qu'il fournit les données sélectionnées au permutateur circulaire de données à 16 bits 14. Ensuite, le permutateur circulaire de données à 16 bits 14 décale la sortie du multiplexeur 13 en fonction de la longueur de mot pour ainsi reconstituer les données de manière à pouvoir accéder à une zone spécifique de la mémoire.

On va maintenant décrire le fonctionnement du permutateur circulaire de données à 16 bits 14, en liaison avec la figure 4.

Le générateur de signaux de demande de données 16 produit le signal de demande de données, le premier

registre à verrouillage 11 devant ainsi recevoir les données chaque fois que la longueur de données à supprimer dépasse seize bits. Le décodeur simple 15 convertit l'entrée binaire en données simples. Lorsque les sorties du décodeur 15 sont "1" et "0", les données à 16 bits du premier registre à verrouillage 11 et les données du deuxième registre à verrouillage 12 sont respectivement sorties et les sorties sont fournies à l'entrée de commande (SEL) d'un multiplexeur. C'est-à-dire que, lorsque l'entrée du décodeur 15 est "0011", la sortie est "1110 0000 0000 000." Le permutateur circulaire de données 14 effectue donc une opération de décalage circulaire vers la gauche.

La figure 2 est un exemple de la table de Huffman montrée à la figure 1. Le Tableau 1 est un exemple de code de longueur variable, c'est-à-dire une table de Huffman pour augmenter une adresse de macrobloc pour l'adressage de macroblocs de MPEG(I) publié par l'ISO/IEC, JTC, 1/SC et 29/WG.

Sur le Tableau 1, on peut voir que de un à cinq bits sont affectés à des valeurs croissantes inférieures à sept, qui sont fréquemment utilisées et que onze bits sont affectés à des valeurs croissantes supérieures à vingt-six, qui ne sont pas fréquemment utilisées. Afin de réduire la taille de la mémoire selon la présente invention, les données de longueur variable sont exprimées sous la forme de huit blocs sur la base d'une position dans laquelle un "1" apparaît dans les données de longueur variable pour la première fois. Ce tableau est uniquement un exemple de N tables de Huffman existant dans la partie tables de Huffman 20, dont la sélection se fait par le signal de sélection de table. Selon la présente invention, la figure 2 montre un exemple dans lequel le Tableau 1 est mis en oeuvre par du matériel.

A la figure 2, la partie adressage 50 est composée des premier au huitième comparateurs 31 à 38 et des



premier au huitième décodeurs binaires 41 à 48. La mémoire 60 conserve les données de la table de Huffman et sort le code de longueur fixe et la longueur de mot correspondants en fonction de la sortie de la partie adressage 50.

Le premier bloc du Tableau 1 est accessible par le premier comparateur 31 et le premier décodeur binaire 41, le cinquième bloc du Tableau 1 est accessible par le cinquième comparateur 35 et le cinquième décodeur binaire 45, et le huitième bloc du Tableau 1 est accessible par le huitième comparateur 38 et le huitième décodeur binaire 48. Les blocs restants sont accessibles de la même manière. A la figure 2, les cinq bits supérieurs des données d'entrée sont fournis à chacun des premier au huitième comparateurs 31 à 38 et un comparateur est choisi parmi ceux-ci. Les premier au huitième décodeurs binaires 41 à 48 reçoivent la sortie du comparateur sélectionné par les bits supérieurs, et les trois bits inférieurs des données d'entrée, et produisent la longueur de mot et la valeur de code fixe décodé, conservée à l'adresse correspondante dans la mémoire 60. Le décodeur de Huffman de la présente invention permet d'accéder simplement à une table de ROM correspondant aux données d'entrée codées en longueur variable sans nécessiter de ROM supplémentaire, et il peut se composer du comparateur et du décodeur binaire, ce qui permet d'éviter une taille de mémoire excessivement grande, non nécessaire. Le temps de recherche en est également réduit.

La figure 3 est un exemple détaillé du cinquième bloc montré sur le Tableau 1.

A la figure 3, le cinquième comparateur 35 met en fonction le cinquième décodeur binaire 45 lorsque les cinq bits supérieurs d'entrée du code de longueur variable (données d'entrée) sont l'ensemble "00001" du comparateur, c'est-à-dire les cinq bits supérieurs de code de longueur variable du cinquième bloc sur le

Tableau 1. C'est-à-dire que les comparateurs 31 à 35 de la figure 2 fixent d'eux-mêmes une adresse de référence (par exemple, le quatrième bloc est "0001" et le sixième bloc est "0000 01") pour classer le bloc et qu'ils mettent en fonction le décodeur binaire correspondant lorsque les cinq bits supérieurs des données d'entrée correspondent à la valeur fixée. Le cinquième décodeur binaire 45 reçoit les trois bits inférieurs des données d'entrée et produit la valeur décodée (code de longueur fixe) et la longueur de mot, conservées dans la ROM correspondante 60. Donc, étant donné que la valeur de données correspondante est "13" ainsi que le montre le Tableau 1, lorsque les données d'entrée sont "000", le "13" mémorisé dans la ROM 60 est sorti en tant que représentation binaire de longueur fixe de cinq bits et le "8" en tant que valeur de longueur de mot (c'est-à-dire le nombre de bits) est sorti en tant que quatre bits. Etant donné que la valeur de données correspondante est "8" et que la longueur du mot est "7" lorsque la donnée d'entrée est "111" comme indiqué sur le Tableau 1, ces valeurs sont sorties sous la forme binaire.

La figure 4 est un exemple de fonctionnement du permutateur circulaire de données.

A la figure 4, les sorties du multiplexeur 13 (à partir du côté droit,  $L_0$  à  $L_{15}$ ) sont restructurées au moyen du décalage circulaire vers la gauche (à partir du côté droit,  $L_1$  à  $L_{15}$  et  $L_0$ ) pour la sortie.

Ainsi qu'il a été décrit ci-dessus, le décodeur de Huffman selon la présente invention, qui est utilisé dans un décodeur de longueur variable, réduit la taille d'une mémoire de recherche, ce qui permet de réduire la taille de la puce lors de la fabrication des circuits VLSI. Le décodage à grande vitesse est également possible grâce à la vitesse de recherche améliorée.

Tableau 1

Code de longueur variable (VLC)	Valeur croissante (contenu du code)
1	1
011 010	2 3
0011 0010	4 5
0001 1 0001 0	6 7
0000 111 0000 110 0000 1011 0000 1010 0000 1001 0000 1000	8 9 10 11 12 13
0000 0111 0000 0110 0000 0101 11 0000 0101 10 0000 0101 01 0000 0101 00 0000 0100 11 0000 0100 10 0000 0100 011 0000 0100 010 0000 0100 001 0000 0100 000	14 15 16 17 18 19 20 21 22 23 24 25
0000 0011 111 0000 0011 110 0000 0011 101 0000 0011 100 0000 0011 011 0000 0011 010 0000 0011 001 0000 0011 000	26 27 28 29 30 31 32 33
0000 0001 111 0000 0001 000	34 (bourrage macrobloc) 35 (changement de code macrobloc)

**REVENDICATIONS**

1. Décodeur de Huffman destiné à recevoir des données codées en longueur variable et à produire un code de longueur fixe correspondant aux données, reconstituées dans une table de Huffman sélectionnée en fonction d'un signal de sélection de table, caractérisé en ce qu'il comprend :

un premier registre à verrouillage (11) pour mémoriser les données actuelles codées en longueur variable ;

un deuxième registre à verrouillage (12) pour conserver des données précédentes codées en longueur variable ;

un décodeur simple (15) pour recevoir la longueur de mot et pour produire un signal de sélection ;

un générateur de signaux de demande de données (16) pour produire un signal de demande de données lorsque la longueur de mot additionnée dépasse une taille prédéterminée après la sommation des longueurs de mots ;

un multiplexeur (13) pour recevoir les sorties desdits premier et deuxième registres à verrouillage (11, 12) et pour sélectionner un bit prédéterminé en fonction dudit signal de sélection ;

un permutateur circulaire de données (14) pour le décalage circulaire de la sortie dudit multiplexeur (13) en fonction de ladite longueur de mot afin de former des données reconstituées, et pour sortir lesdites données reconstituées ;

une partie adressage (50) pour comparer lesdites données reconstituées à une valeur prédéterminée et pour décoder les données ; et

une mémoire (60) pour mémoriser la longueur dudit mot codé en longueur variable et le code de longueur fixe, qui lui correspond, tous deux étant produits en fonction de la sortie de ladite partie adressage (50).

2. Décodeur de Huffman selon la revendication 1, caractérisé en ce que ladite partie adressage (50) comprend :

5 des comparateurs (31 à 38) pour recevoir les bits supérieurs des données permutées sorties par ledit permutateur circulaire de données (14) et pour comparer une adresse fixée dans les comparateurs aux bits supérieurs reçus ; et

10 des décodeurs binaires (41 à 48), mis en fonction par la sortie desdits comparateurs et destinés à recevoir et à décoder les bits inférieurs desdites données permutées.

15 3. Décodeur de Huffman selon la revendication 1, caractérisé en ce qu'il comprend en outre un sélecteur de table (24) afin de sélectionner une table de Huffman spécifique parmi une pluralité de tables de Huffman en fonction dudit signal de sélection de table.

FIG. 1

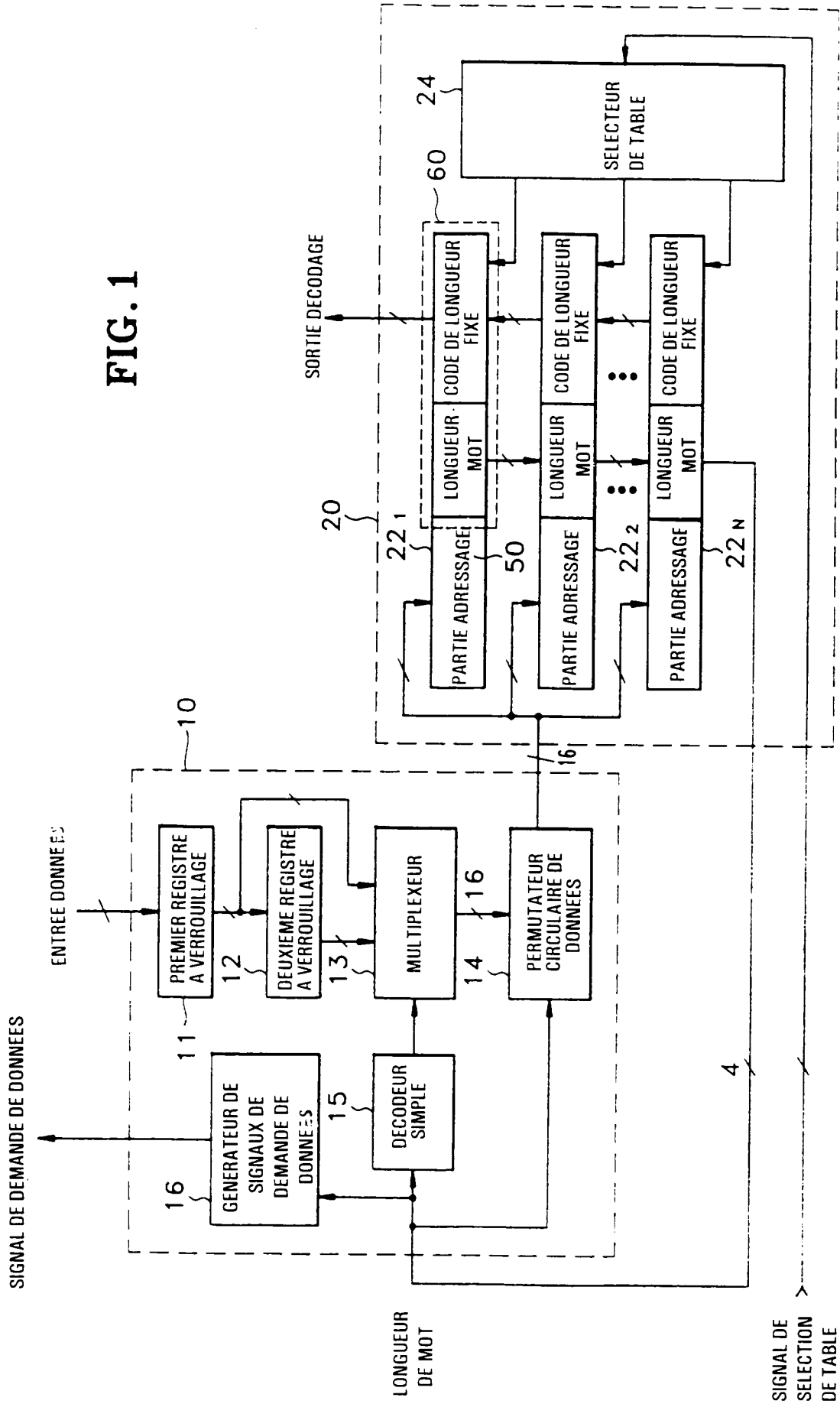


FIG. 2

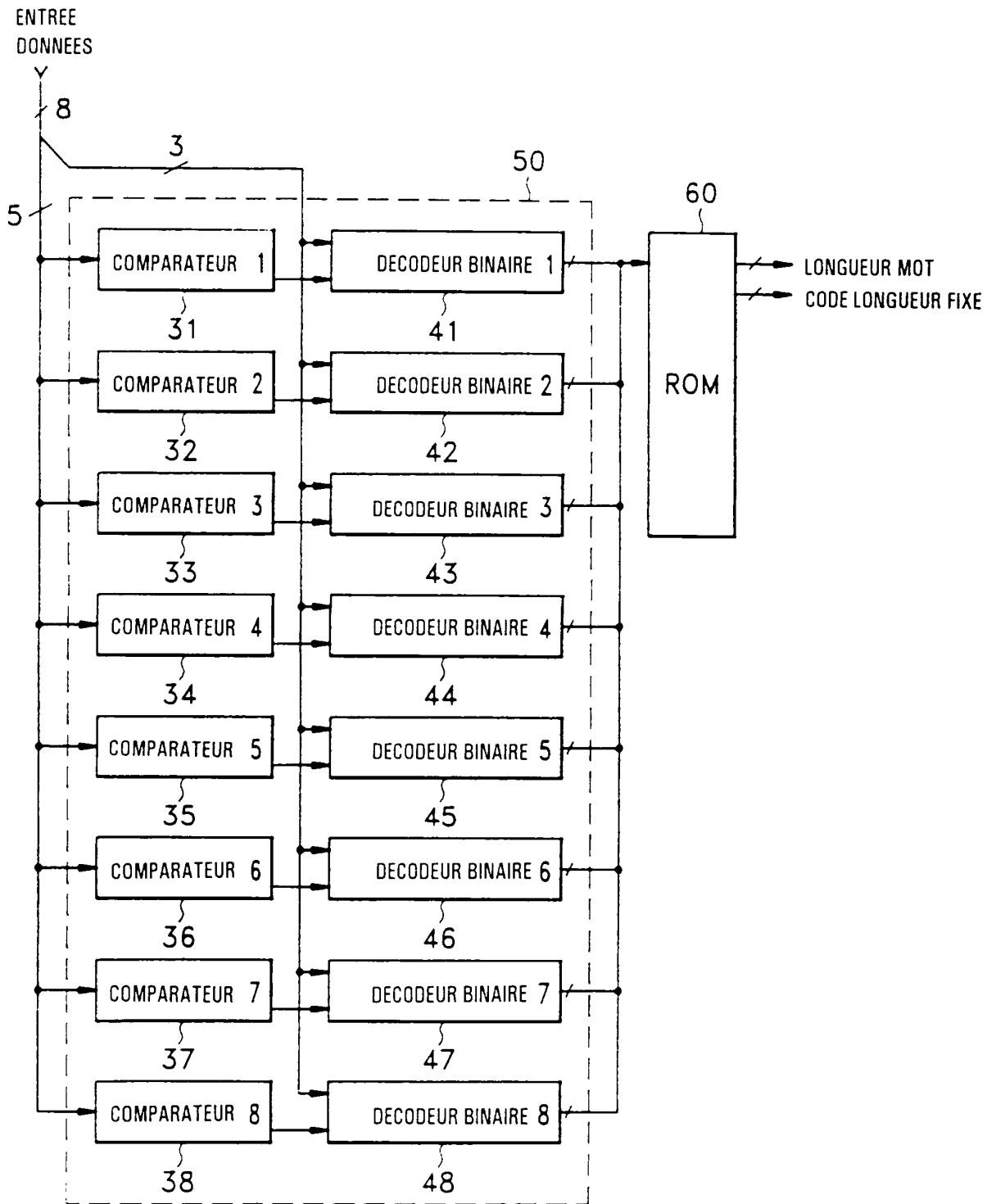


FIG. 3

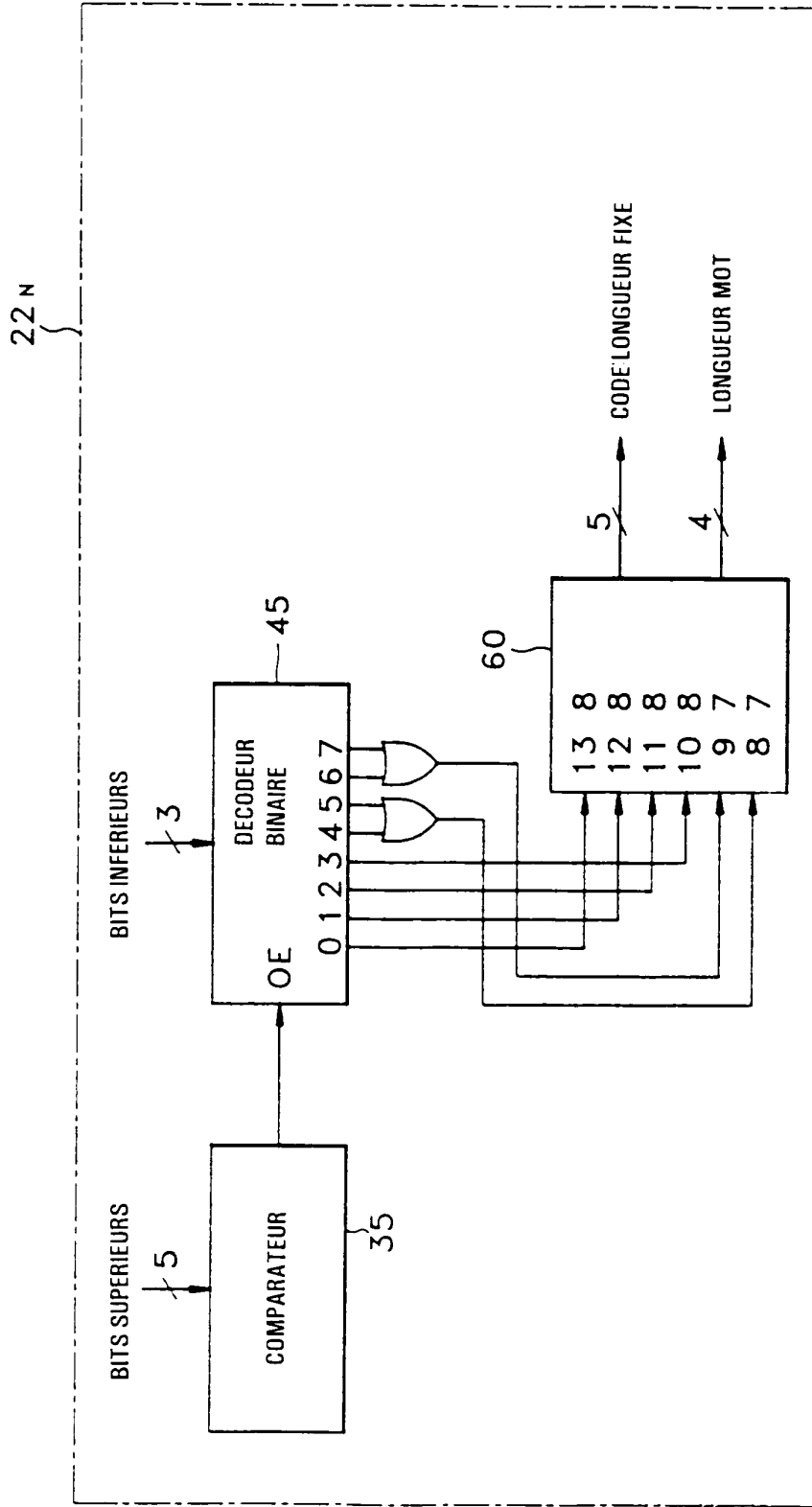
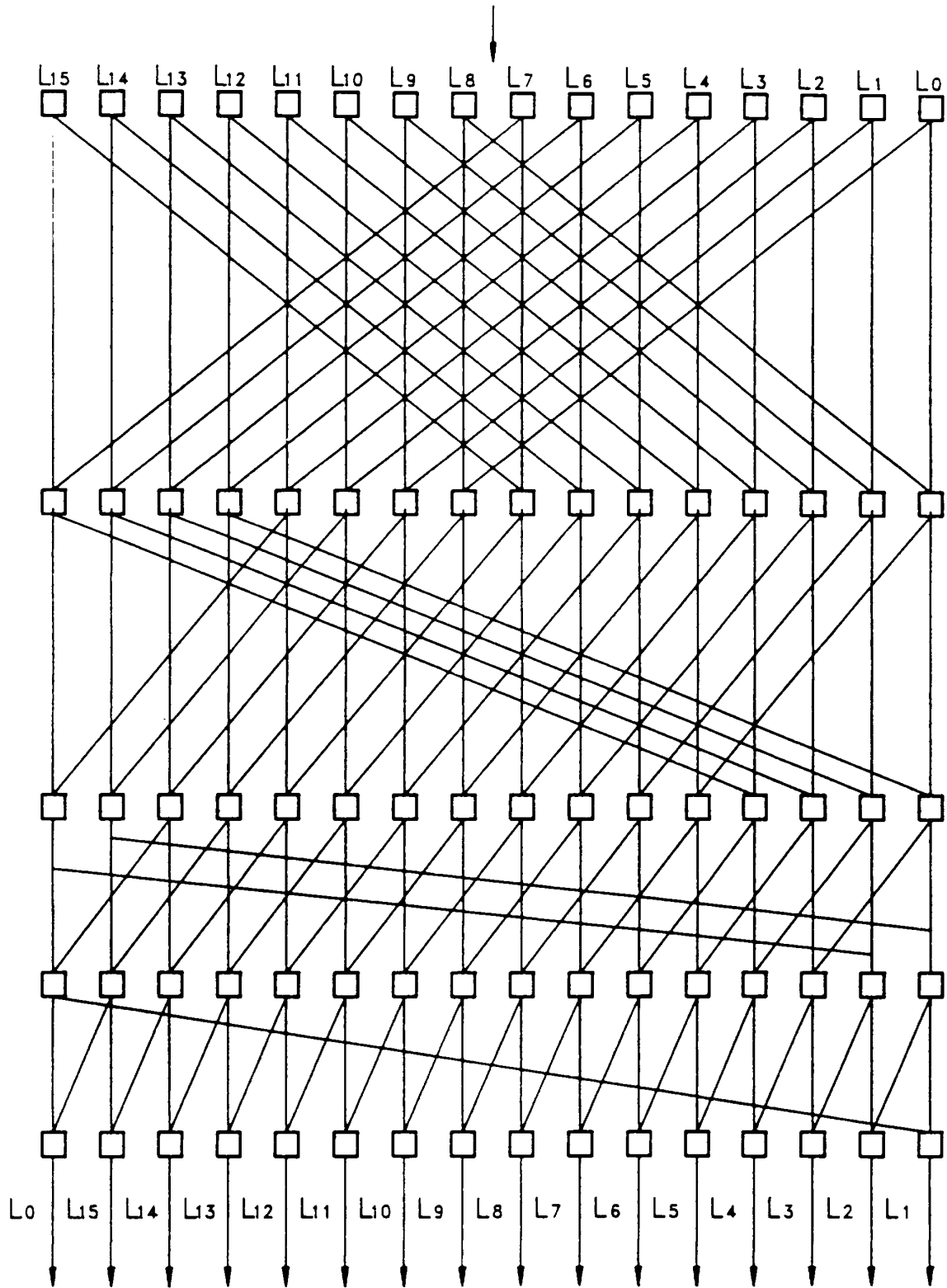




FIG. 4

ENTREE DONNEES 16 BITS



SORTIE DE DONNEES 16 BITS A PERMUTATION CIRCULAIRE