

(19) RÉPUBLIQUE FRANÇAISE
INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE
PARIS

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

2 645 296

(21) N° d'enregistrement national : .

89 04302

(51) Int Cl⁵ : G 06 F 12/16, 15/66; H 05 K 10/00.

(12)

DEMANDE DE BREVET D'INVENTION

A1

(22) Date de dépôt : 31 mars 1989.

(71) Demandeur(s) : Société dite : GENERAL ELECTRIC CGR SA, Société Anonyme. — FR.

(30) Priorité :

(72) Inventeur(s) : Philippe Bodelet, Jean-Luc Renoux et Jean Herzog, Cabinet Ballot-Schmit.

(43) Date de la mise à disposition du public de la demande : BOPi « Brevets » n° 40 du 5 octobre 1990.

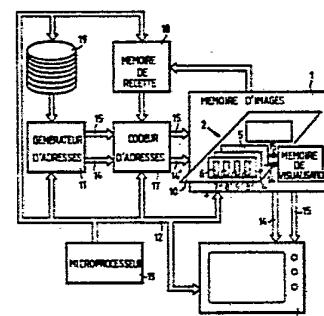
(73) Titulaire(s) :

(60) Références à d'autres documents nationaux appartenus :

(74) Mandataire(s) : Cabinet Ballot-Schmit.

(54) Procédé et dispositif de réparation d'une mémoire d'images.

(57) Pour réparer une mémoire d'images d'un processeur d'images, on utilise une mémoire 1 d'images de plus grande capacité que ce qui est nécessaire, on repère dans cette mémoire d'images les blocs mémoires qui sont défectueux, et on effectue un déroutage 17 d'adresses par interposition d'un codeur d'adresses pour que les informations à stocker soit stockées dans des régions de remplacement de cette mémoire d'images plutôt que dans des régions réputées défectueuses de cette mémoire. D'une manière préférée on code une mémoire de transcodage du codeur d'adresses en fonction d'une liste des mauvais blocs de la mémoire d'images contenue dans une mémoire de recettes 18 non volatile. L'invention trouve plus particulièrement son application dans le domaine médical pour la gestion des processeurs d'images à haute définition.



FR 2 645 296 - A1

D

1

PROCEDE ET DISPOSITIF DE REPARATION
D'UNE MEMOIRE D'IMAGES

La présente invention a pour objet un procédé et un dispositif de réparation d'une mémoire d'images du type de celles utilisées dans le domaine médical, en association avec un processeur d'images, pour visualiser des images hautes définitions (1024 X 1024 points). Ces images sont généralement représentatives d'examens cliniques préalables menés au moyen de tomographes, de tomodensitomètres, ou de machines de résonance magnétique nucléaire (RMN). L'invention peut néanmoins trouver son application dans d'autres domaines, où la visualisation simultanée d'images haute définition est nécessaire.

Une image haute définition avec un Mega éléments d'image (1 Mega = 1024 X 1024) ou 1 Mega pixels, nécessite pour son stockage huit Mega cellules mémoires si chacune de ces cellules mémoires contient un bit, et si des niveaux de luminosité à affecter aux éléments d'images de cette image, sont codés sur huit bits. Le stockage d'une image nécessite donc la présence de circuits intégrés de mémoire en nombre suffisant pour que leurs capacités cumulées soit égale à huit Mega bits. Un processeur d'images de type médical doit être capable de traiter en temps réel les informations contenues classiquement dans environ une dizaine d'images. Cette dizaine d'images représente par exemple des tomographies effectuées dans des tranches adjacentes d'un corps examiné. En pratique, on retient un stockage de seize images. Ceci conduit à devoir disposer en mémoire d'une capacité de $8 \times 16 = 128$ Mega bits. Pour l'accès en temps réel, la mémoire d'images dont il est

question est bien entendu une mémoire très rapide. Dans la technologie actuelle elle est de préférence de type dynamique et donc volatile. Ceci signifie qu'avant d'utiliser le processeur d'images et cette mémoire d'images, il convient de charger cette mémoire, en une opération préalable, avec des informations d'images correspondantes stockées sur un autre support : un disque magnétique par exemple. Compte tenu de la capacité évoquée de 128 Mega bits, et compte tenu de ce que les circuits intégrés élémentaires de mémoire dynamique actuellement disponibles dans le commerce ont une capacité de 1 Mega bits, il est nécessaire pour réaliser un processeur d'images avec la capacité énoncée, d'aligner 128 circuits intégrés élémentaires. L'emprise d'implantation sur une carte mère de ces 128 circuits intégrés électroniques est bien trop importante, et en pratique, ces circuits intégrés ne sont pas intégrés à plat sur une carte mère.

Au contraire, on utilise des plaquettes intermédiaires. Ces plaquettes sont du type circuit imprimé et sont de forme rectangulaire. Sur ces plaquettes on assemble côté à côté habituellement huit circuits intégrés. Les circuits intégrés utilisés sont de forme rectangulaire. Soit ces circuits intégrés sont mis côté à côté adjacents par leurs grands côtés pour former un alignement qui s'étend dans le sens de la longueur de la plaquette rectangulaire. Soit les circuits intégrés sont alignés bout à bout, par leurs petits côtés, et dans ce cas ils sont en plus implantés de part et d'autre de la plaquette. Les plaquettes sont ensuite elles même embrochées sur chant sur la carte mère du processeur. Compte tenu de ce qu'un circuit intégré peut contenir un Mega bits, de ce qu'on en met huit sur une plaquette et qu'ainsi une plaquette

correspond au stockage d'une image, et qu'on veut réaliser des processeurs d'images avec seize images, il convient d'implanter sur la carte mère seize plaquettes mises côte à côte. Pour des raisons de compacité, de compatibilité avec les normes VME, et de fiabilité les plaquettes sont dressées sur chant sensiblement les unes contre les autres avec un espace minimum entre elles. Cet espace minimum est par exemple de l'ordre de 0,5 mm. Il est complètement insuffisant pour permettre le désassemblage d'un circuit intégré de ces plaquettes sans imposer en même temps le débrochage de la plaquette elle même de la carte mère.

Ceci pose des problèmes de réparation et des problèmes de fiabilité. En effet, lorsqu'il s'avère qu'un circuit intégré est défectueux parce qu'au moins une de ses cellules mémoires est incapable de retenir d'une manière fiable l'information qu'on cherche à y enregistrer, il devient nécessaire de le remplacer. On procède alors de la manière suivante. On dessoude d'abord la plaquette de la carte mère, et on dessoude ensuite ce circuit imprimé concerné de la plaquette. Pour remplacer, on effectue normalement l'opération inverse en utilisant un nouveau circuit intégré en bon état. Cependant, les règles de l'art conduisent à ne plus utiliser une soudure directe du circuit intégré de remplacement sur la plaquette, ni même de la plaquette sur la carte mère. Au contraire, on doit plutôt utiliser des connecteurs interposés. Ces connecteurs sont soudés respectivement sur la plaquette d'une part, et sur la carte mère d'autre part. La raison en est la suivante. Autant la technologie de soudure et de dessoudure permet de dessouder une première fois un composant fiché sur un circuit imprimé, autant on admet que cette opération n'est pas envisageable une deuxième fois sans risque

pour le circuit imprimé : la plaquette ou la carte mère. En conséquence, à la première réparation on soude un connecteur intermédiaire de sorte que si le circuit intégré tombe à nouveau en panne, la deuxième réparation est aussi plus simple. Elle consiste à arracher mécaniquement le composant du connecteur pour en enficher un autre en bon état à la place.

Avec un connecteur intermédiaire la réparation présente un gros inconvénient : l'interposition des connecteurs modifie l'encombrement de la plaquette au-dessus de la carte mère. La plaquette ainsi réparée ne peut plus être alors glissée entre les autres plaquettes de la mémoire d'images du processeur d'images puisqu'elle est maintenant agrémentée de composants hors gabarit. De même, la plaquette dressée sur chant dépasse au-dessus du niveau des autres du fait de la présence de son connecteur. Compte tenu de la compacité, les plaquettes ont une hauteur telle qu'elles souscrivent juste à une norme d'espacement entre cartes principales d'une armoire électronique où la carte mère du processeur d'images est engagée.

Le résultat de cette réparation est que la nouvelle carte mère ou la nouvelle plaquette n'est plus engageable dans l'espace trop justement calculé. La solution qui consiste à réaliser des espacements hors normes n'est industriellement pas acceptable car les problèmes de compatibilité entre les équipements qu'elle pose disqualifie son utilisation dans des grands ensembles électroniques. On rappelle à ce sujet que les normes de fabrication s'imposent à tous les concepteurs. Leurs non respect entraîne des retards considérables dans la fabrication définitive des matériels vendables sur le marché. On note par ailleurs, que la réparation des mémoires d'images ainsi préconisée n'est pas sans

effet sur la fiabilité de la mémoire finale. En effet, en ayant subi des opérations de soudure et dessoudure on altère d'une manière ou d'une autre la structure de cette mémoire d'images et donc sa fiabilité.

Dans l'invention, on propose de remédier à cet inconvénient de réparation en proposant une réparation complètement différente. Le principe de l'invention consiste à réaliser des processeurs d'images avec une mémoire d'images comportant une capacité de mémoire d'images en excédent par rapport aux besoins nominaux d'utilisation. Cet excédent de capacité est utilisé pour remplacer les parties utiles mais défectueuses de la mémoire d'images.

En pratique, on utilise un transcodeur d'adresses qui permet de stocker dans des zones de remplacement de la mémoire, des informations qui devraient normalement être stockées dans d'autres zones de celle-ci, dont on sait par ailleurs qu'elles présentent des dysfonctionnements. Le transcodeur d'adresses est placé normalement entre un générateur d'adresses, piloté par un microprocesseur et la mémoire d'images. Ceci signifie que la réparation de la mémoire d'images devient transparente pour l'utilisateur qui continue à gérer par l'intermédiaire du microprocesseur, cette mémoire d'images.

Cette manière de faire a pour résultat immédiat que la fiabilité du processeur d'images, ou de la mémoire d'images ainsi réalisés n'est pas affectée par la réparation. Une étude des pannes a permis de montrer que les mémoires d'images tombent en panne dans 90 % des cas du fait d'un mauvais fonctionnement d'une cellule de circuit mémoire. En conséquence, en agissant ainsi on divise par dix le taux de panne du processeur d'images complet.

La détermination du bon état de fonctionnement des circuits intégrés électroniques élémentaires de mémoire est effectuée par un programme dit de recette. Ce programme de recette est mis en oeuvre par le microprocesseur qui gère la mémoire. Ce programme de recette délivre une liste des blocs ou des cellules de cette mémoire qui sont défectueux. C'est avec cette liste que normalement on effectue les réparations. Dans l'invention, on code le transcodeur interposé entre le générateur d'adresses et la mémoire d'images en fonction du résultat de cette recette : en fonction du contenu de cette liste. En conséquence, une opération de réparation de la mémoire consiste à mettre en service régulièrement, par exemple tous les trois mois, le programme de recette. L'immobilisation du processeur d'images et de la mémoire d'images pendant le déroulement de ce programme de recette est de l'ordre de 1 heure. Elle ne peut pas être considérée comme une panne au même titre que celle que l'on connaissait dans l'état de la technique où, pour les réparations avec débrochages et rembrochages la durée d'immobilisation était estimée en jours voire même en semaines. Dans le cas présent "la réparation" peut en plus être effectuée directement par l'utilisateur qui n'a qu'une procédure de type logiciel à mettre en oeuvre. Cet utilisateur n'a besoin d'avoir aucune compétence en technologie de réparation. Le processeur d'images monté avec le procédé de l'invention possède alors un MTBF (Moyen Temps de Bon Fonctionnement) de l'ordre de 7500 heures alors qu'il n'avait un MTBF que de 1500 heures dans l'état de la technique.

En conséquence, l'invention a pour objet un procédé de réparation d'une mémoire d'images d'un processeur d'images, cette mémoire comportant

- un jeu de modules mémoires plaqués les uns contre les autres ,
- chaque module comportant une plaquette rectangulaire de connexion, du type circuit imprimé,
- chaque plaquette étant munie d'un certain nombre de circuits intégrés électroniques élémentaires de mémoire,
- ces modules et ces circuits intégrés étant en quantité tels qu'ensemble ils puissent convenir, par partition, à la mise en mémoire d'informations correspondant à plus d'une image, ce processeur comportant
- un générateur d'adresses pour accéder à des cellules mémoires d'une première partition de cette mémoire, ces cellules mémoires de cette première partition étant chargées d'informations relatives à des éléments d'images d'une première image,
- un circuit de lecture pour lire des contenus d'information stockés dans lesdites cellules,
- une mémoire dite de recette pour mémoriser une liste de cellules mémoires du jeu des modules mémoires de la mémoire d'images dont le fonctionnement a été préalablement jugé défectueux, et
- interposé entre le générateur d'adresses et les modules mémoires, un codeur d'adresses, caractérisé en ce qu'on effectue
 - un transcodage des adresses des cellules défectueuses de la première partition délivrées par le générateur d'adresses en des adresses de cellules de remplacement contenues dans une deuxième partition de la mémoire,
 - en fonction du contenu d'information de la mémoire de recette.

L'invention a également pour objet un processeur d'images apte à mettre en oeuvre le procédé écrit ci-dessus. Notamment, dans une réalisation particulière, la liste des cellules ou des groupes de cellules jugés défectueux est chargée dans une mémoire non volatile associée au microprocesseur de manière à ce qu'à chaque allumage de ce microprocesseur celui-ci puisse programmer le transcodage d'adresses en fonction du contenu de cette mémoire volatile.

L'invention sera mieux comprise à la lecture de la description qui suit et à l'examen des figures qui l'accompagnent. Celles-ci ne sont données qu'à titre indicatif et nullement limitatif de l'invention.

Les figurent montrent :

- figure 1 : un schéma synoptique d'un processeur d'images selon l'invention apte à mettre en oeuvre le procédé inventé ;

- figure 2 : une représentation schématique de l'affectation de parties d'une zone en excéder de la mémoire à des parties correspondantes de cette mémoire ayant été jugées défectueuses.

La figure 1 montre un processeur d'images utilisable pour mettre en oeuvre le procédé de l'invention. Une mémoire d'images 1 à réparer comporte, dans une banque 2 des modules mémoires tels que 3 à 5. Chaque module comporte une plaquette de connexion du type circuit imprimé, muni d'un certain nombre de circuits intégrés tel que 6 à 9 électroniques élémentaires de mémoire. Les modules sont plaqués les uns contre les autres sur chant au-dessus d'une carte dite carte mère 10 de telle façon que l'enlèvement des circuits intégrés 6 à 9 en vue de leurs remplacements ne soit normalement pas possible, au moins pour les plaquettes qui ne sont pas aux extrémités. Dans un

exemple la banque 2 comporte 16 plaquettes de manière à contenir 16 images.

Le processeur d'images comporte un générateur d'adresses 11 pour accéder à des cellules mémoires d'une première partition de cette mémoire, cette première partition correspondant à l'une des images stockées. Dans l'exemple décrit il y a 16 partitions. Le nombre de partition peut bien entendu être différent et les partitions de la mémoire peuvent être de tailles différentes : notamment la partition de réparation peut avoir une capacité plus petite ou plus grande que celle nécessaire au stockage d'une image. Le processeur d'images comporte également un circuit de lecture écriture pour lire des informations stockées dans les cellules mémoires. Ce circuit de lecture comporte essentiellement un bus de commande 12 par lequel le microprocesseur 13 envoie des instructions de lecture, écriture pour que, par l'intermédiaire d'un bus d'adresses 14 et d'un bus de données 15, les images à représenter soient visualisées sur un monitor 16.

La particularité de l'invention se situe dans la présence d'un codeur d'adresses 17 intermédiaire entre le générateur d'adresses 11 et la mémoire d'images 1. Ce codeur d'adresses 17 est en fait un transcodeur. Il reçoit des adresses de cellules mémoires de la mémoire d'images, à lire ou à écrire, provenant du générateur d'adresses. Si les adresses des cellules concernées sont des adresses qui correspondent à des cellules en bon état, le codeur d'adresses les laisse passer telles quelles. En conséquence, les cellules mémoires de la mémoire d'images qui sont ainsi adressées par le générateur d'adresses sont effectivement celles où le générateur d'adresses avait destiné l'information. Par contre le codeur d'adresses 17, lorsqu'il reconnaît une

adresse défectueuse, émet sur le bus 14 en aval une adresse différente de celle qu'il a reçu, et correspondant à une cellule mémoire de remplacement dont l'adresse est associée. En pratique, ce codeur d'adresses 17 comporte en entrée un décodeur d'adresses et en sortie une mémoire programmable à accès aléatoire. Le contenu d'information chargé à chaque adresse de cette mémoire programmable à accès aléatoire du codeur d'adresse est, d'une part l'adresse elle même si cette adresse correspond à une cellule mémoire réputée en bon état de la mémoire d'images ou, d'autre part, une adresse d'une cellule mémoire de remplacement dans le cas contraire.

Pour simplifier l'explication on peut évoquer ce qui ce passe dans un exemple théorique chiffré, cependant non limitatif de l'invention. Dans le cas où une image possède $1024 \times 1024 = 1$ Mega pixels, et que les niveaux de luminosité à affecter à chacun de ces pixels sont codés sur huit bits (un octet) la capacité mémoire nécessaire pour contenir cette image est de 1 Mega octets. On peut admettre en conséquence que la mémoire d'images comporte huit Mega cellules, ou 1 Mega octets. On va admettre, dans un premier temps, que le remplacement des cellules défectueuses est fait octet par octet. En pratique, on utilisera des partitions bien plus grande de la mémoire pour ne pas perdre trop de temps ni trop de place dans le décodeur d'adresses du codeur d'adresses 17. Pour accéder à chacun des 1 Mega octets de l'image, il convient que le générateur d'adresses 15 émette un mot d'adresse comportant 20 bits. Dans ces conditions, le nombre d'emplacements mémoires de la mémoire programmable du codeur d'adresses 17 doit être de 1 Mega, et la capacité de chacun de ces emplacements doit être de 20 bits.

Avant de charger une image dans la mémoire d'images, on extrait d'une mémoire de recette 18 des adresses, dites de défaut, d'octets mémoires de la mémoire d'images qui sont défectueux. On fait émettre par le générateur d'adresses 15 ces adresses de défauts. On a alors accès aux cases mémoires de la mémoire programmable du codeur d'adresse 17 qui correspondent aux adresses de cellules mémoires défectueuses de la mémoire d'images. On charge alors ces cases mémoires de la mémoire programmable du codeur d'adresse 17 avec, comme contenu d'information, des adresses d'octets de la mémoire d'images en bon état, et, situés dans la partition en excédent de la mémoire d'images. Ce faisant, on a programmé le codeur d'adresses 17 en fonction du contenu de la mémoire de recette 18. Pour les autres adresses de la mémoire d'images, on charge dans les cases de la mémoire programmable situées à ces adresses un contenu d'information identique justement à ces adresses. Le codeur d'adresses 17 est alors apte à jouer son rôle de déroutage, en lecture ou en écriture, des cellules mémoires de la mémoire d'images à l'initiative du générateur d'adresses 15 piloté par le microprocesseur 13.

Mais la réalisation, dans le codeur d'adresses 15, d'une mémoire de 1 Mega mots de 20 bits (tout ça pour accéder en définitive à des octets mémoires d'une mémoire d'images de 1 Mega mots de huit bits) n'est pas une bonne solution, surtout compte tenu de ce qu'un processeur d'images comporte en général une mémoire d'images susceptible de stocker 16 images. Pour faire un remplacement octet par octet il serait ainsi nécessaire d'avoir dans le codeur d'adresses une mémoire de 16 Mega mots et chaque mot aurait une capacité de 24 bits !

Aussi de préférence à un remplacement octet par octet, et a fortiori cellule par cellule, on préfère un remplacement groupe d'octets par groupe d'octets. En pratique, dans une application où on peut stocker en mémoire 16 images de 1 Mega mots de huit bits, on se contente d'un stockage de 15 images et on réserve la capacité de la seizième image à la contenance des groupes d'octets de remplacement. D'une manière préférée, au lieu d'accéder en réparation à chaque octet indépendamment, on accédera à tous les octets d'un seizième de l'image.

En conséquence, le décodeur d'adresses du codeur d'adresses 17 ne devra plus recevoir en adresse que des mots de huit bits et ne plus contenir dans chacune des 256 cases de sa mémoire programmable correspondante que des mots de huit bits. Les huit bits d'adresse correspondent, pour quatre premiers bits, au choix d'une image parmi les 16 possibles, et pour les quatre bits suivants, à quatre bits de poids fort des adresses des éléments d'images de l'image choisie. La programmation du codeur d'adresses 17 est alors la suivante. On repère dans une partition mémoire affectée à une image (on connaît ainsi les quatre premiers bits) un octet correspondant à une cellule mémoire défectueuse. De l'adresse codée sur 20 bits dans cette image de cet octet défectueux on ne retient que les quatre bits de poids fort. Ces quatre bits de poids fort constituent, avec les quatre bits de l'image choisie, les huit bits de l'adresse globalisée du défaut.

Ces huit bits sont introduits en adresse dans le décodeur d'adresse du codeur d'adresses 17. Ils donnent accès à une case mémoire de la mémoire programmable de ce codeur d'adresses. Cette case mémoire est alors chargé d'un mot de huit bits. Les quatre premiers bits

de ce mot sont immuables, par exemple 1111, pour désigner la seizième partition de la mémoire d'images. Les quatre suivants sont, dans cette partition supplémentaire, l'adresse d'un seizième de bloc de cette partition supplémentaire. On effectue le même travail pour toutes les cellules jugées défectueuses de la mémoire d'images. Une fois que ceci est terminé, on charge les cases mémoires non encore chargées de la mémoire programmable du codeur d'adresses 17 avec des contenus identiques aux adresses d'accès à ces cases. Dans ces conditions le codeur d'adresses est complètement programmé pour jouer le rôle qu'on attend de lui : celui de transcodage selon l'invention.

Il est connu des programmes de scrutation des mémoires d'images pour la recette qui permettent de déterminer l'état de fonctionnement des cellules, des octets, ou des groupes d'octets des modules mémoires de ces mémoires d'images. Ces programmes produisent normalement une liste qui permet aux techniciens du service après vente de déterminer quels sont les circuits intégrés de mémoire qui sont défectueux et qu'il faut changer. Dans l'invention, et selon ce qui a été dit ci-dessus, on garde dans une mémoire non volatile les huit bits de poids fort qui renseignent sur la position des blocs mémoires en mauvais état. À chaque mise sous tension il devient alors possible de programmer la mémoire programmable du codeur d'adresse 17. Compte tenu de cette opération, la mise en service, par exemple chaque matin, du processeur d'images de l'invention dure environ 1 seconde de plus que d'habitude. Le programme de recette qui évalue l'état de fonctionnement des cellules mémoires de la mémoire d'images dure environ 1 heure. Ce programme peut être lancé régulièrement, tous les 3 mois par exemple, ou dès

que l'on a détecté par d'autres moyens un défaut de la mémoire d'images. Ces autres moyens sont par exemple la détection dans les images d'un défaut local. D'une manière préférée la mémoire de recette 18 est donc une mémoire non volatile qui permet, au prix de cette durée supplémentaire de 1 seconde de disposer d'une mémoire d'images 1 constamment réparée.

Les images sont normalement stockées en longue durée dans une unité de disque ou de tambour magnétique 19. Elles sont chargées quand le besoin s'en fait sentir à partir de cette unité 19 dans la mémoire d'images 1 en utilisant le générateur d'adresse 15 couplé au codeur d'adresses 17. Donc, pour cet organe aussi le transcodage d'adresses est transparent.

La mémoire d'images n'est pas normalement la mémoire qui est lue pour que ces images soient visualisées sur le moniteur 16. Dans la pratique on utilise une mémoire de visualisation 30, qui fait partie intégrante de la mémoire d'images 1 mais dans laquelle sont chargées en fonction des besoins des parties de modules mémoires de la mémoire d'images que l'on veut voir apparaître. La mémoire 30 a bien entendu une capacité au moins égale à la résolution maximale de l'image la plus grande qu'on veut voir. En pratique, elle a au moins 1 Mega octets de capacité. On s'est alors rendu compte que cette mémoire de visualisation était aussi sujette à des pannes et qu'il était possible d'appliquer le même de raisonnement pour réparer les éventuelles déficiences de cette mémoire de visualisation 30. On a ainsi été conduit à réaliser une mémoire de visualisation de 2 Mega octets de capacité alors qu'on en utilise en fait que 1 Mega octets.

Dans le Mega octets supplémentaires on a réalisé des partitions de telle façon qu'on puisse puiser dans

ces partitions des régions de remplacement susceptibles de se substituer à des régions utiles de cette mémoire de visualisation mais qui se sont par ailleurs révélées comme étant défectueuses. Normalement les adresses de stockage, dans la mémoire de visualisation, des éléments d'images qu'on veut voir sont les adresses produites par le générateur d'adresses 15 avant leurs transcodages. Si on a détecté des défauts et si on veut procéder à un remplacement dans la mémoire de visualisation 30, le codeur d'adresses 17 comporte, en supplément, une partie destinée à effectuer la même fonction que précédemment, mais en ce qui concerne la mémoire de visualisation cette fois. Les bus d'adresses 14 et de données 15 sont gérés par le microprocesseur 13 de la même façon que pour les modules mémoires de la mémoire d'images.

La figure 2 représente schématiquement la capacité mémoire 20 de la mémoire d'images 1. Cette mémoire comme indiqué jusqu'ici est divisée en seize partitions, la seizième partition 21 étant destinée à servir de partition de réparation. La mise en oeuvre du programme de recette à permis de stocker dans la mémoire de recette 18 des adresses de blocs mémoires 22 ou 23 de la mémoire 2 qui se sont avérés défectueux. Le codeur d'adresses 17 comporte dans sa mémoire programmable un codage qui permet d'affecter à chacun de ces blocs, par exemple le bloc 24, une région correspondante 25 (entourée par deux traits) de la seizième partition 21. Bien entendu, le programme de recette qui scrute aussi la seizième partition 21 peut déterminer la présence dans cette seizième partition 21 d'un mauvais bloc 26. Dans ces conditions il interdit l'utilisation de ce bloc 26 à titre de remplacement.

REVENDICATIONS

1. Procédé de réparation d'une mémoire (1) d'images d'un processeur d'images, ce processeur comportant
 - un générateur d'adresses (15) pour accéder à des cellules mémoires d'une première partition de cette mémoire, ces cellules étant chargées d'informations relatives à des éléments d'image d'une première image,
 - un circuit de lecture écriture (12-14) pour lire des contenus d'information stockés dans ces cellules,
 - une mémoire dite de recette (18) pour mémoriser une liste de cellules mémoires du jeu des modules mémoires de la mémoire dont le fonctionnement a été préalablement jugé défectueux, et
 - interposé entre le générateur d'adresses et les modules mémoires, un codeur d'adresses (17), caractérisé en ce qu'on effectue
 - un transcodage des adresses des cellules défectueuses (24) de la première partition en des adresses (25) de cellules de remplacement contenues dans une deuxième partition de la mémoire
 - en fonction du contenu d'information de la mémoire de recette.
2. Procédé selon la revendication 1 caractérisé en ce qu'on répare une mémoire d'images comportant
 - un jeu de modules (3-5) mémoires plaqués les uns contre les autres,
 - chaque module comportant une plaquette rectangulaire de connexion, du type circuit imprimé,
 - chaque plaquette étant munie d'un certain nombre de circuits intégrés électroniques élémentaires de mémoire (6-9),

- ces modules et ces circuits intégrés étant en quantité telle qu'ensemble ils puissent convenir, par partition, à la mise en mémoire de plus d'une image.

3. Procédé selon la revendication 1 ou la revendication 2, caractérisé en ce qu'on réalise la mémoire de recette avec une mémoire non volatile.

4. Procédé selon l'une quelconque des revendications 1 à 3, caractérisé en ce qu'on réalise des sous partition de la première partition et en ce qu'on transcode toutes les adresses des cellules mémoire d'une sous partition dès qu'une des cellules mémoires de cette sous partition a été jugée défectueuse.

5. Processeur d'image comportant une mémoire (1) d'images, un générateur d'adresses (15) pour accéder à des cellules mémoires d'une première partition de cette mémoire, ces cellules étant chargées d'informations relatives à des éléments d'image d'une première image, un circuit (12-14) de lecture pour lire des contenus d'information stockés dans ces cellules, et une mémoire dite de recette (18) pour mémoriser la liste des cellules mémoires du jeu des modules mémoires de la mémoire dont le fonctionnement a été préalablement jugé défectueux, caractérisé en ce qu'il comporte, interposé entre le générateur d'adresses et les modules mémoires, un codeur d'adresses (17) pour effectuer un transcodage des adresses des cellules défectueuses (24) de la première partition en des adresses (25) de cellules de remplacement contenues dans une deuxième partition de la mémoire, et en ce que le transcodage est effectué en fonction du contenu d'information de la mémoire de recette.

6. Processeur selon la revendication 5, caractérisé en ce que la mémoire d'images comporte un jeu de modules (3-5) mémoires plaqués les uns contre les autres, chaque

module comportant une plaquette rectangulaire de connexion, du type circuit imprimé, chaque plaquette étant munie d'un certain nombre de circuits intégrés (6-2) électroniques élémentaires de mémoire, ces modules et ces circuits intégrés étant en quantité telle qu'ensemble ils puissent convenir, par partition, à la mise en mémoire d'au moins deux images.

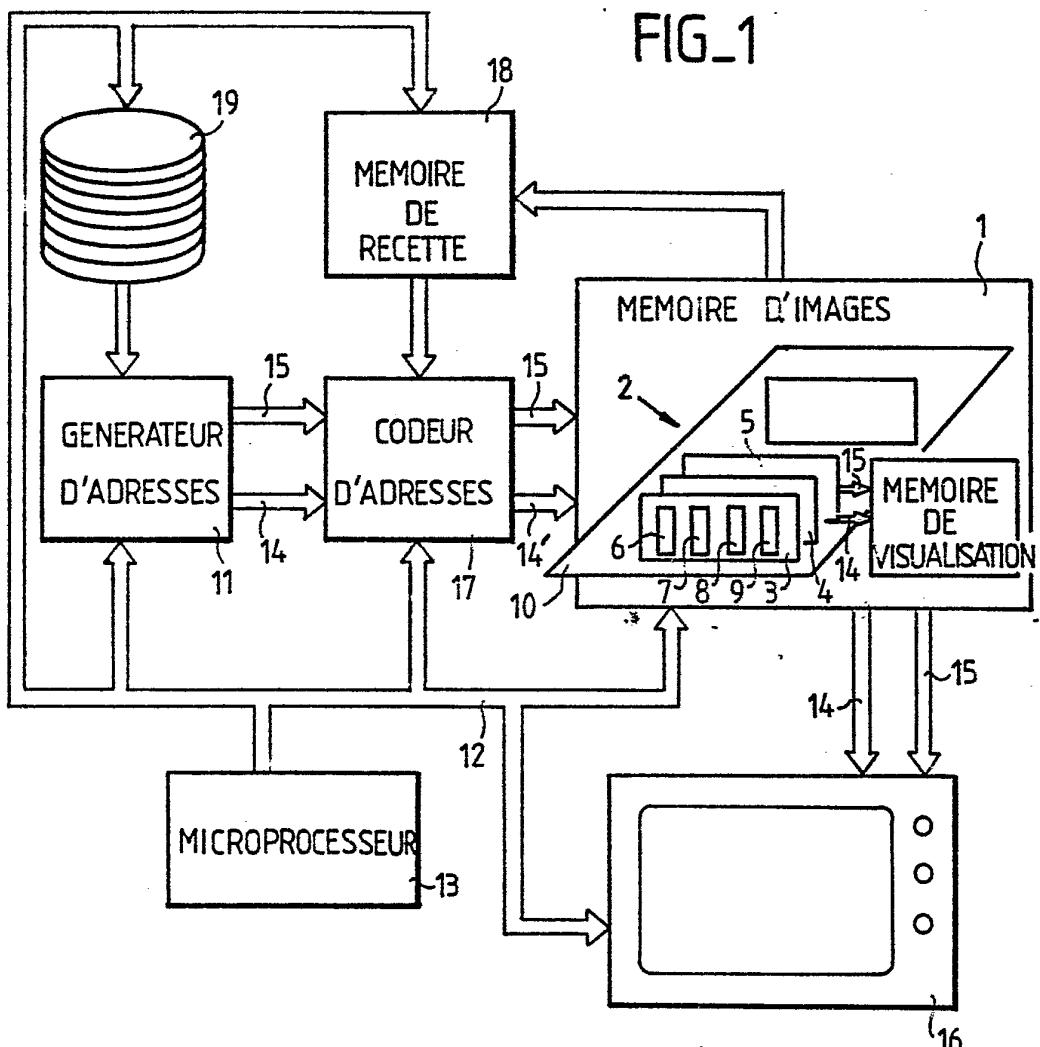
7. Processeur selon la revendication 5 ou la revendication 6, caractérisé en ce que chaque plaquette est dressée et est connectée directement et sensiblement sur chant sur la carte mère (10) du processeur.

8. Processeur selon l'une quelconque des revendications 5 à 7, caractérisé en ce que la mémoire est une mémoire d'images (30) en relation fonctionnelle directe avec un moniteur (16) de visualisation

9. Processeur selon l'une quelconque des revendications 5 à 8, caractérisé en ce que la deuxième partition correspond à une deuxième image.

10. Processeur selon l'une quelconque des revendications 5 à 9, caractérisé en ce que la mémoire de recette est une mémoire non volatile.

1/1

**FIG_2**