

①②

DEMANDE DE BREVET D'INVENTION

A1

②② Date de dépôt : 20.12.02.

③⑦ Priorité :

④③ Date de mise à la disposition du public de la
demande : 25.06.04 Bulletin 04/26.

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

⑥⑦ Références à d'autres documents nationaux
apparentés :

⑦① Demandeur(s) : STMICROELECTRONICS SA
Société anonyme — FR.

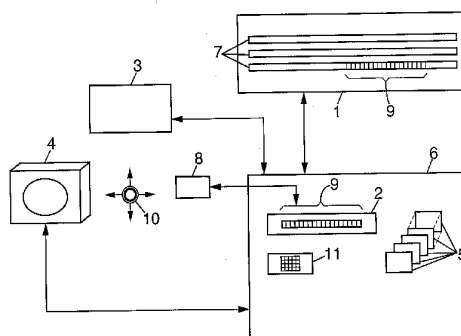
⑦② Inventeur(s) : ROELENs FREDERIC.

⑦③ Titulaire(s) :

⑦④ Mandataire(s) : CABINET PLASSERAUD.

⑤④ PROCÉDE ET DISPOSITIF ET DECODAGE ET D'AFFICHAGE EN MARCHE ARRIERE D'IMAGES MPEG,
CIRCUIT PILOTE VIDEO ET BOITIER DECODEUR INCORPORANT UN TEL DISPOSITIF.

⑤⑦ Un procédé de décodage et d'affichage en marche ar-
rière d'images d'un flux de données d'images compressées
selon la norme MPEG utilisant un nombre déterminé N de
mémoires de trame adaptées chacune pour stocker une
image décodée, où N est un nombre entier supérieur ou
égal à 4, applique des règles de sélection d'une mémoire de
trame (5) lorsqu'une mémoire de trame doit être écrasée
pour permettre le stockage d'une nouvelle image à décoder.



PROCEDE ET DISPOSITIF DE DECODAGE ET D’AFFICHAGE EN MARCHÉ
ARRIERE D’IMAGES MPEG, CIRCUIT PILOTE VIDEO ET BOITIER
DECODEUR INCORPORANT UN TEL DISPOSITIF

La présente invention se rapporte au décodage et à l’affichage de programmes audiovisuels contenus dans un flux de données numériques compressées.

Elle trouve des applications dans les décodeurs vidéo comportant une
5 fonctionnalité d’enregistrement numérique personnel (PVR, de l’anglais
"Personal Video Recorder"). Dans de tels décodeurs, un enregistrement de
programmes est effectué sur un support numérique de stockage de masse (ou
DSM, de l’anglais "Digital Storage Media") tel qu’un disque dur, une cassette
vidéo numérique, un DVD ("Digital Versatile Disk"), etc...

10 Les décodeurs concernés sont notamment les boîtiers décodeurs ("Set-
Top Box" en anglais) assurant l’interface entre une arrivée de signaux
numériques diffusés en temps réel par satellite, par câble ou par diffusion
numérique terrestre (ou DVB, de l’anglais "Digital Video Broadcasting") d’une
part, et un téléviseur analogique d’autre part. Un tel boîtier décodeur est un
15 appareil autonome ("Stand-alone"). Mais l’invention s’applique aussi aux
décodeurs intégrés à un téléviseur numérique, ou à un appareil numérique de
lecture/enregistrement tel qu’un lecteur de DVD, un lecteur de cassettes vidéo
numériques ("Digital Video Cassette Recorder"), ou similaire.

Un boîtier décodeur reçoit un ou plusieurs programmes sous la forme
20 d’un flux de données numériques compressées, c’est-à-dire dont les trames
vidéo (i.e., les images) et les trames audio sont codées de manière à réduire le
volume des données diffusées. Par exemple, ce codage respecte les
spécifications de la norme MPEG-2 (ISO/IEC 13818-2) ci-après norme MPEG
("Motion Pictures Expert Group"). Cette norme définit une technique de
25 compression de données pour le codage des images animées et du son
(notamment pour la télévision numérique).

Dans la présente description, le terme "présentation", lorsqu’il est utilisé
en relation avec un programme, désigne la restitution des données audio et
vidéo décodées, sous forme respectivement audible et visible par un utilisateur.
30 Le terme "lecture" ("playback" en anglais), utilisé en relation avec un
programme, désigne plus spécifiquement le décodage et l’affichage du

programme à partir d'un enregistrement du programme sur un disque dur. Le terme affichage ("display" en anglais) est plus particulièrement utilisé pour désigner la présentation des trames vidéo.

Le disque dur permet d'offrir un certain nombre de fonctionnalités, notamment la mise en œuvre de modes de lecture évolués ("trick modes", en anglais), c'est-à-dire une lecture à une vitesse différente d'une vitesse nominale (x1) en marche avant. Il s'agit notamment de l'avance rapide ("fast forward") et de la lecture en marche arrière ("rewind") accélérée ou non.

Les difficultés rencontrées aujourd'hui pour la mise en œuvre de cette dernière fonctionnalité proviennent du fait que le flux MPEG est, par nature, adapté à une lecture en marche avant. En effet, les images sont transmises dans un ordre destiné à faciliter le décodage en marche avant. Pour la lecture en marche arrière, cet ordre de transmission impose de disposer de ressources de calcul ayant une rapidité de décodage suffisante pour réaliser toutes les opérations de décodage nécessaires au décodage de chaque image. A défaut, il convient de disposer d'une grande capacité de stockage pour stocker toutes les images décodées (i.e., les données d'images après décompression) à partir du début d'un groupe d'images ("Group Of Pictures" ou GOP) au sens de la norme MPEG, dans la mesure où les images du début du GOP peuvent à nouveau servir au décodage d'une image suivante dans le GOP. Cette capacité de stockage est fournie par des mémoires de trame, dont chacune est adaptée pour stocker les données d'une image décodée.

Dans la demande de brevet européen EP-A-0 735 780, il est proposé une technique de lecture en marche arrière d'un flux vidéo MPEG utilisant le même nombre de mémoires de trames que pour la lecture en marche avant. Ce nombre est égal à trois. Néanmoins la méthode consiste à sauter les images B, et à ne décoder que trois images I ou P au maximum, puis à passer au GOP précédent (dans l'ordre d'affichage en marche avant) dès que ce nombre est atteint.

La demande internationale de brevet WO 01/37572 propose, pour la lecture en marche arrière d'un flux vidéo MPEG, une technique permettant d'afficher toutes les images d'un GOP. La technique consiste à stocker temporairement, dans au minimum quatre mémoires de trame, les données

décompressées de certaines images utiles au décodage ultérieur de plusieurs images à afficher, plutôt que de recommencer les opérations de décodage successives depuis le début du GOP. Ce document décrit un exemple de GOP qui peut ainsi être décodé dans son entier en utilisant quatre mémoires de trame seulement. Cet exemple correspond au cas de la lecture d'un programme stocké sur un DVD, avec des GOPs contenant un nombre limité d'images. Cependant il n'est fourni aucune garantie sur le résultat de la technique.

Un but de l'invention est, parmi d'autres, de proposer des règles de gestion pour optimiser l'usage des mémoires de trame disponibles pour l'affichage en marche arrière d'images d'un flux vidéo MPEG, en particulier un flux vidéo MPEG, stocké sur le disque dur d'un boîtier décodeur.

Un premier aspect de l'invention propose ainsi un procédé de décodage et d'affichage en marche arrière d'images d'un flux de données d'images compressées selon la norme MPEG utilisant un nombre déterminé N de mémoires de trame adaptées chacune pour stocker une image décodée, où N est un nombre entier supérieur ou égal à 4. Les règles suivantes sont appliquées lorsqu'une mémoire de trame doit être écrasée pour permettre le stockage d'une nouvelle image à décoder :

- on écrase en priorité une mémoire de trame dans laquelle est stockée une image obsolète c'est-à-dire une image qui a déjà été affichée et qui n'est plus nécessaire pour effectuer un prochain décodage d'image ou, à défaut ;

- on écrase en priorité une mémoire de trame dans laquelle est stockée une image qui présente un coût de décodage minimum ;

- une mémoire de trame dans laquelle est stockée une image qui est en cours d'affichage ne peut pas être écrasée ;

- une mémoire de trame dans laquelle est stockée une image qui est prête pour l'affichage mais n'a pas encore été affichée ne peut pas être écrasée ;

- une mémoire de trame dans laquelle est stockée une image de référence nécessaire pour le décodage de ladite nouvelle image à décoder ne peut pas non plus être écrasée ;

- si aucune mémoire de trame ne peut être écrasée en application des

règles précédentes, on suspend le décodage jusqu'à l'affichage d'une image prête pour l'affichage et on évalue à nouveau ces règles après cet affichage.

Le fait qu'une image satisfasse la condition selon laquelle une image a déjà été affichée et n'est plus nécessaire pour le décodage d'une autre image, 5 (en particulier une autre image à décoder pour obtenir la prochaine image à afficher) peut être déterminée à partir d'une table d'indexation dans sa version enrichie (voir plus loin).

Ces règles permettent d'atteindre un résultat optimal en terme de continuité de l'affichage, en fonction notamment de la rapidité du décodage 10 pour une vitesse d'affichage donnée. Quand aucune mémoire de trame ne peut être écrasée pour le stockage de la nouvelle image à décoder en application des cinq premières règles, la dernière règle conduit à attendre qu'il y en ait une qui puisse être utilisée, ceci se produit quand est affichée une image qui est prête pour l'affichage en étant stockée dans l'une des mémoires de trame. Le 15 procédé peut se réaliser en utilisant un nombre N de mémoires de trames qui est idéalement aussi grand que possible. Le procédé produit ses effets dès lors que N est inférieur au nombre d'images contenues dans un GOP, groupe d'images ("Group Of Pictures") au sens de la norme MPEG. Au minimum, N doit être égal à quatre. En utilisant ces mémoires selon le procédé, on limite le 20 nombre total d'opérations de décodage à réaliser pour afficher les images en marche arrière. Lorsque N est égal à quatre, l'une des mémoires de trame est utilisée pour stocker l'image en cours d'affichage, et, parmi les trois autres, l'une peut être utilisée pour recevoir les données décompressées de l'image à 25 décodage et les deux autres pour stocker les au plus deux images de référence nécessaires à cet effet (si l'image à décoder est une image B). Bien entendu, l'affectation des mémoires de trames évolue dynamiquement de façon non déterministe.

Avantageusement, les images de type I présentent un premier coût de décodage déterminé ; les images de type P prédites à partir d'une image de 30 type I présentent un deuxième coût de décodage, supérieur audit premier coût de décodage ; et les images de type P prédites à partir d'une autre image de type P présentent un coût de décodage supérieur au coût de décodage de ladite autre image de type P.

Par exemple, si le coût de décodage d'une image I est égal à l'unité, le coût de décodage d'une image P prédite à partir de n images de référence (image I et/ou image(s) P précédentes) est égal à $n + 1$.

De préférence, on considère que les images de type B présentent le même coût de décodage que les images I, car elles ne sont décodées qu'au moment de leur affichage et peuvent être écrasées immédiatement après (puisque'une image B n'est pas une image de référence servant au décodage d'une autre image).

Dans un mode de mise en œuvre avantageux, on vérifie avant de décoder une prochaine image à afficher, si elle n'est pas déjà stockée dans l'une quelconque des mémoires de trame. Dans ce cas en effet, il est inutile de la décoder à nouveau.

Pour l'affichage, le procédé peut comprendre des étapes consistant à déterminer à chaque événement de synchronisation d'affichage d'image si les mémoires de trames contiennent une image prête pour l'affichage. Un tel événement correspond par exemple à une impulsion du signal de synchronisation verticale (Signal Vsync) de l'écran d'un téléviseur analogique. On entend par image prête pour l'affichage, une image décodée venant à la suite, dans l'ordre d'affichage des images en marche arrière, des images qui ont déjà été affichées. Si une telle image est présente alors on l'affiche, et on la marque comme obsolète après l'affichage si elle n'est plus nécessaire pour effectuer un décodage d'image, c'est-à-dire si elle n'est pas une image de référence pour une autre image à décoder/afficher. Pour la vérification de cette dernière condition, on peut avantageusement se baser sur le contenu d'une table d'indexation enrichie (voir plus loin) associée à la portion de flux comprise dans la mémoire tampon. Si au contraire aucune telle image n'est présente, on répète l'affichage de l'image affichée à l'événement de synchronisation d'affichage d'image précédent. Ceci revient à effectuer un gel d'image, qui est préférable à un manque d'image apparaissant comme une image noire à l'écran.

Dans le type d'applications envisagé, le flux de données compressées est lu par portions à partir d'un support numérique de stockage de masse, par exemple un disque dur, en réponse à une commande de lecture en marche

arrière.

Un mode de mise en œuvre de l'invention comprend les étapes consistant à :

- a) charger une portion du flux dans une mémoire tampon ;
- 5 b) analyser ladite portion du flux afin d'identifier le point d'accès ("Access Point") d'un groupe d'images décodable et de déterminer au moins pour chacune des images dudit groupe, des informations incluant l'adresse de l'image dans la mémoire tampon, une estampille temporelle de présentation associée à un ordre d'affichage de l'image en marche avant, et le type I, P ou B
- 10 c) obtenir une table d'indexation enrichie en complétant la table d'indexation obtenue à l'étape b en précisant pour chaque image de type P ou B sa ou ses images de référence ;
- 15 d) à partir de la table d'indexation, déterminer pour une prochaine image à afficher, une liste ("Trick List") de reconstruction qui est une liste ordonnée des images de référence à décoder séquentiellement à partir dudit point d'accès pour obtenir ladite prochaine image à afficher ;
- 20 e) à partir de ladite liste de reconstruction, obtenir ladite prochaine image à afficher en décodant tout ou partie des images de ladite liste de reconstruction.

Un groupe d'images décodable est un groupe d'images consécutives du flux qui peuvent être décodées sans avoir besoin de connaître d'autres images que celles du groupe (hormis pour les images constituant ce qui est appelé un

25 lien cassé ("Broken Link") dans la norme MPEG. Typiquement un tel groupe correspond à un GOP, bien qu'un GOP puisse parfois contenir plusieurs tels groupes d'images décodables. Un point d'accès, au sens de la norme MPEG, est une image I à partir de laquelle il est possible de décoder un groupe d'images venant à la suite dans le flux.

30 De préférence, le procédé comprend, entre l'étape b) et l'étape c), l'attribution à chacune des images codées dans ladite première portion du flux d'un numéro d'identification unique permettant de l'identifier de manière univoque dans le flux.

Dans un exemple, les numéros d'identification sont attribués aux images codées dans le flux de façon séquentielle, suivant l'ordre d'affichage des images. A cet effet, on utilise les PTS précitées.

5 Ainsi, à l'étape d), la prochaine image à afficher en marche arrière non encore décodée peut être déterminée à partir de son numéro d'identification, en décomptant les numéros d'identification.

10 Le numéro d'identification des images est par exemple codé sur 32 bits, et est initialisé à la valeur h0x800000 (c'est-à-dire 0x800000 en notation hexadécimale) et est respectivement incrémenté ou décrétementé d'une unité selon qu'on parcourt le flux en marche avant ou en marche arrière. Ainsi, cela permet la lecture en marche avant ou en marche arrière, de plusieurs heures de vidéo.

15 Avantageusement, pour obtenir une prochaine image, on vérifie d'abord si l'une au moins des images de sa liste de reconstruction n'est pas déjà stockée dans l'une quelconque des mémoires de trame. Dans l'affirmative, on commence à décoder séquentiellement les images de ladite liste non pas à partir du point d'accès du groupe d'images décodable mais à partir de ladite image déjà stockée, ou, le cas échéant, à partir de celles desdites images déjà stockées qui présentent le coût de décodage le plus élevé. On évite ainsi
20 d'effectuer à nouveau des opérations de décodage qui ont déjà été faites et dont le résultat est toujours disponible dans les mémoires de trames.

Dans un mode de mise en œuvre avantageux on calcule, à chaque fois qu'une image est affichée, un retard par rapport à une vitesse d'affichage théorique (par exemple la vitesse nominale x1 si on affiche en marche arrière à
25 une vitesse non accélérée). De plus, à l'étape d), on détermine la prochaine image à afficher de manière à annuler le retard en sautant des images (c'est-à-dire en ne les décodant pas et en ne les affichant pas), selon les règles de priorité suivantes :

- on saute en priorité une ou plusieurs images B ;
- 30 - si un ou des sauts d'images B ne suffisent pas pour annuler le retard, on saute une ou plusieurs images P ;
- si un ou des sauts d'images P ne suffisent pas pour annuler le retard, on saute une ou plusieurs portions du flux de données lors du

chargement, à l'étape a), d'une prochaine portion du flux de données dans la mémoire tampon.

Ces règles permettent notamment de déterminer s'il convient de réaliser des sauts dans les images à afficher, en fonction du retard pris dans l'affichage par rapport à la vitesse nominale, voire si une portion de flux ne peut pas être
5 décodée. Ainsi, on peut compenser des retards du processus de décodage par rapport au processus d'affichage.

De préférence, la mémoire tampon comprend un premier et un second banc de mémoire. Les étapes d) et e) sont réalisées dans ledit premier banc de
10 mémoire pour une première portion du flux de la mémoire tampon, pendant que les étapes a) à c) sont réalisées dans ledit second banc de mémoire pour une seconde portion du flux. Le processus de décodage est ainsi plus fluide.

Avantageusement, on charge la mémoire tampon de manière que l'un ou l'autre des premier et second bancs de mémoire n'est pas écrasé à l'étape
15 a) tant que les données qu'il contient sont encore nécessaires au traitement des données contenues dans l'autre desdits premier et second bancs de mémoire.

De préférence, l'un ou l'autre des premier et second bancs de mémoire est écrasé avec une nouvelle portion du flux, à l'étape a), dès que les données
20 qu'il contient ne sont plus nécessaires au traitement des données contenues dans l'autre desdits premier et second bancs de mémoire. On perd ainsi un minimum de temps avant de pouvoir entamer le processus de décodage pour une portion suivante du flux.

Avantageusement, les portions de flux chargées dans le premier banc
25 de mémoire et dans le second banc de mémoire se chevauchent, une jonction entre ces deux portions pouvant être réalisée entre les deux parties de flux au niveau d'une image commune. Ceci est particulièrement avantageux lorsque le flux est embrouillé ("scrambled"). En effet, il est alors impossible de courber le flux de données exactement entre deux images.

30 On notera que la table d'indexation peut alors avantageusement être commune pour les deux portions de flux.

A noter qu'on se place ici au niveau de l'image dans un but de généralisation de l'exposé, mais que le procédé s'applique de la même façon à

chaque trame d'une image dans le cas d'un système d'affichage entrelacé ("Interleaved Display").

Un deuxième aspect de l'invention se rapporte à un dispositif comprenant des moyens pour la mise en œuvre du procédé ci-dessus.

5 Un troisième aspect de l'invention se rapporte à un circuit pilote vidéo ("Video Driver" en anglais) comprenant un dispositif selon le deuxième aspect.

Un autre aspect encore de l'invention concerne un boîtier décodeur ("Set Top Box" en anglais) comprenant un circuit pilote vidéo selon le troisième aspect.

10 D'autres particularités et avantages de la présente invention apparaîtront dans la description ci-après d'exemples de réalisation non limitatifs, en référence aux dessins annexés, dans lesquels :

- la figure 1 présente un dispositif permettant de mettre en œuvre l'invention ;

15 - la figure 2 présente un synoptique présentant les étapes du procédé selon une réalisation de notre invention ;

- la figure 3 présente une table d'indexation telle que constituée dans l'étape b) d'une réalisation de la présente invention ;

20 - la figure 4 présente la table d'indexation de la figure 3 dans sa version enrichie après l'étape c) ;

- la figure 5 présente des images contenues dans un GOP et les liens de prédiction entre les différentes images ;

- la figure 6 présente le contenu d'une table d'indexation relative à une première portion de flux dans un premier banc de mémoire ;

25 - la figure 7 présente le contenu d'une seconde portion de flux dans un second banc de mémoire avant l'étape d'indexation ;

- la figure 8 présente la table d'indexation commune aux deux portions de flux des deux bancs de mémoires selon les figures 6 et 7, après la phase de jonction entre la première et la seconde portion de flux.

30 Rappelons tout d'abord quelques notions utilisées dans la norme MPEG. Tout d'abord, le type I, P ou B des images codées est défini de la manière suivante :

- une image de type I (intra) ou image I est une image codée sans

référence à une autre image et pour laquelle l'ensemble des informations nécessaires au décodage est contenu dans l'image elle-même ;

- une image de type P (prédite) ou image P est une image qui doit être décodée en utilisant les informations relatives à une image de référence I ou P
5 antérieure suivant l'ordre d'affichage en marche avant (par la suite, on nommera une image P prédite d'une image I « image P prédite une fois », une image P prédite elle-même d'une image P prédite une fois « image P prédite deux fois » et plus généralement une image P « image P prédite n fois » lorsqu'elle est prédite depuis une image I via n-1 images P ;

10 - une image de type B (bidirectionnelle) ou image B est une image qui doit être décodée en utilisant des informations relatives à une première et une seconde images de référence (I et/ou P) antérieures ou postérieures suivant l'ordre d'affichage en marche avant.

Par conséquent, les images forment des groupes élémentaires qui sont
15 décodables indépendamment des autres groupes élémentaires. Un groupe d'images c'est-à-dire un GOP, au sens de la norme MPEG peut contenir plusieurs de ces groupes élémentaires. Dans les faits, il s'avère toutefois qu'un GOP comprend au plus un seul groupe élémentaire. Des images B bidirectionnelles peuvent être interpolées entre des images de ces groupes
20 élémentaires, constituant alors un lien cassé ("broken-link") au sens de la norme.

Des points d'accès directs associés à des codes de démarrage (codes de séquence et codes d'extension de séquence) caractérisent le début de ces groupes élémentaires.

25 La figure 1 montre schématiquement un dispositif pour la mise en œuvre du procédé selon l'invention.

Le dispositif comprend :

- un disque dur 1 sur lequel est stocké un flux 7 de données contenant des images codées selon la norme MPEG ;

30 - une mémoire 6, qui comprend :

une mémoire tampon 2 destinée à stocker une portion 9 de flux compressé et,

une table d'indexation 11 sur lesquelles on reviendra plus loin et,

des mémoires de trames 5 destinées à stocker les données d'images décompressées.

- un module analyseur 8, adapté pour détecter les codes de démarrage, extraire du flux des informations permettant de localiser les images dans la mémoire, en donnant un point de départ à un comptage d'octets utilisé pour la localisation,

- un décodeur vidéo 3, en particulier un décodeur MPEG, qui est capable de décoder une image MPEG définie notamment par sa position dans la mémoire tampon et dont les paramètres définissant les zones mémoires où il faut stocker l'image à décoder et celles où il faut lire les données des images de référence (décodées) nécessaires à ce décodage, sont configurables,

- un module d'affichage 4, adapté pour lire les images décodées dans la mémoire 6 et les traduire dans le format utilisé pour l'affichage, par exemple le format PAL, SECAM ou NTSC. Il permet en outre la sélection des trames dans le cas d'images entrelacées,

- une unité de contrôle 10 qui supervise le fonctionnement du dispositif.

Les étapes du procédé selon l'invention vont maintenant être exposées en référence au diagramme de la figure 2.

Le procédé démarre à la réception d'une commande de lecture en marche arrière des images contenues dans le flux 7 stocké sur le disque dur 1. Dans une première étape 21, la mémoire tampon 2 est chargée avec une première portion 31 (figure 1) du flux 7. Ces images sont disponibles dans leur ordre d'enregistrement sur le disque dur et aucune information sur les images notamment sur leur localisation dans cette portion 31 n'est disponible à ce stade (on suppose que le démultiplexage/désembrouillage a été réalisé en amont, sinon une étape intermédiaire de démultiplexage/désembrouillage doit être ajoutée).

Puis, dans une étape 22, la portion 31 de flux 7 est analysée à l'aide du module analyseur 8, afin de notamment déterminer des informations incluant la position des codes de démarrage des groupes élémentaires, l'adresse de départ des images dans la mémoire tampon 2, pour chaque image une estampille temporelle associée à un ordre d'affichage en marche avant, et le type I, P ou B de chacune des images codées dans ladite première portion de

flux. Ces informations sont stockées dans une table d'indexation telle que 11 (figure 1).

Un exemple de table d'indexation résultant de la mise en œuvre de l'étape 22 est donné par le tableau de la figure 3. Elle correspond à une portion de flux 31 provenant d'une diffusion satellite ; l'astérisque sur les images n°2, 14 et 26 signale la présence d'un code de démarrage au niveau de ces images correspondant au début d'un groupe élémentaire. La mémoire tampon 2 contient ici 28 images (numérotées ici de 0 à 27 pour faciliter le repérage). L'image n°27 n'est pas complète : seul le début des données qui la codent a été extrait du flux 7. Pour chaque image, la table contient l'adresse dans la mémoire tampon du début de l'image, le type I, P ou B et l'estampille temporelle de l'image dans le groupe. Sur la figure 3, les images sont identifiées par leur type I, B ou P et par leur numéros d'ordre d'affichage en marche avant dans le groupe élémentaire (soit pour l'exemple considéré de 0 à 11) ; ainsi I1, qui a été diffusé dans le flux avant B0, est une image de type I qui sera affichée (selon un ordre d'affichage en marche avant) avant l'image de type B identifiée par B2 et avant l'image de type B identifiée par B0, pour le groupe élémentaire considéré.

Ensuite, dans une étape 23, on attribue à chaque image du flux un identifiant unique. Il s'agit par exemple d'un nombre codé sur 32 bits (la valeur d'initialisation étant par exemple h0x80000000) croissant selon l'ordre d'affichage en marche avant. De plus on détermine pour chacune des images identifiées de façon unique dans la portion 31 du flux 7, quelles sont ses images de référence c'est-à-dire les images décodées à obtenir pour pouvoir décoder l'image considérée.

Un exemple du résultat de l'étape 23 mise en œuvre sur le contenu de la table d'indexation 11 considérée précédemment et illustrée sur la figure 3, est illustré par le tableau de la figure 4, dont la colonne position donne l'adresse du début de chaque image dans la mémoire tampon, la seconde colonne indique le type d'image et le numéro d'ordre d'affichage en marche avant dans le groupe élémentaire correspondant, la troisième colonne indique l'identifiant unique attribué représenté dans sa forme décimale pour plus de lisibilité, les quatrième et cinquième colonnes indiquent les images de références

respectivement antérieures et postérieures des images de type P et B. Ainsi, par exemple, pour obtenir l'image B4 d'identifiant 65523, le tableau indique que les images de référence P3 d'identifiant 65522 et P5 d'identifiant 65524 sont nécessaires. Les points d'interrogation indiquent que les informations ne sont pas disponibles ; le sigle N indique qu'aucune donnée n'est nécessaire.

Dans une étape 24, une image donnée étant décodée et étant en cours d'affichage, on détermine via les informations de la table d'indexation produite à l'étape 22, la prochaine image à afficher et son adresse, et on décode cette image. On considère au regard du tableau de la figure 4, l'exemple suivant :

10 l'image P5 d'identifiant 65524 est en cours d'affichage ; la prochaine image à décoder et afficher est donc l'image B4 d'identifiant 65523. Pour réaliser le décodage de l'image B4, on réalise les étapes consistant à :

- dans une étape 241, à partir des images de référence de l'image à décoder B4 d'identifiant 65523 et des résultats de l'étape 23, on établit la liste (ordonnée) des images de référence successives qu'il est nécessaire de décoder depuis les images de type I correspondant à un code de démarrage de groupe élémentaire d'images. Cette liste est appelée "liste de reconstruction". Ainsi pour obtenir B4 d'identifiant 65523, il faut avoir P5 d'identifiant 65524 et P3 d'identifiant 65522. Pour avoir P3 d'identifiant 65522, il faut avoir I1 d'identifiant 65520. Pour avoir P5 d'identifiant 65524, il faut avoir P3 d'identifiant 65522, donc avoir I1 d'identifiant 65520. I1 d'identifiant 65520 est une image de type I, correspondant à un code de démarrage. La liste de reconstruction de B4 d'identifiant 65523 est donc {B4, P5, P3, I1}, dont les identifiants ont été indiqués ci-dessus.

25 - la prochaine étape 242 consiste à réaliser ensuite, à partir de la liste de reconstruction constituée à l'étape 241, le décodage MPEG dans l'ordre d'affichage en marche avant des images contenues dans la liste de reconstruction pour réaliser in fine le décodage MPEG de l'image B4 d'identifiant 65523. Ainsi on reconstruit cette image B4 en décodant les images

30 I1 d'identifiant 65520, puis P3 d'identifiant 65522, puis P5 d'identifiant 65524, puis B4 d'identifiant 65523.

La dernière étape (étape 25) consiste, une fois les données décompressées de B4 d'identifiant 65523 ainsi obtenues, à afficher cette

image. Les étapes d'affichage et de décodage se font en parallèle. En pratique, l'affichage ne peut toutefois avoir lieu, à une vitesse d'affichage nominale commandée par l'utilisateur, que lorsque l'image devant être affichée a été décodée. En cas de retard du décodage, le dispositif d'affichage continue
5 donc à afficher l'image en cours d'affichage tant que l'image à afficher suivante n'est pas disponible. Par ailleurs, le décodage peut se réaliser en avance de phase sur les prochaines images à afficher, dans la mesure où les ressources en mémoire et la vitesse du décodeur le permettent. Par exemple, pendant que l'image B10 d'identifiant 65529 est affichée, les étapes 241 et 242
10 peuvent, si les ressources en mémoire et la vitesse du décodeur le permettent, être mises en œuvre pour décoder l'image P9 d'identifiant 65528, puis B8 d'identifiant 65527, puis P7 d'identifiant 65526 et les tenir à disposition du dispositif d'affichage.

Ces dernières étapes 24 et 25 sont reproduites tant qu'il reste des
15 images à afficher non décodées parmi toutes les images contenues dans la première portion du flux 31 et pour lesquelles les informations nécessaires au décodage sont présentes.

Pour stocker les données décompressées des images décodées utilisées pendant les phases de décodage et d'affichage, on utilise un certain
20 nombre de mémoires de trames 5. L'une de ces mémoires de trames est utilisée pour recueillir les données de l'image en cours d'affichage, les autres sont utilisées pour le décodage de la prochaine image à afficher. Dans un cas idéal, l'élément décodable individuellement étant un groupe élémentaire, il faudrait autant de mémoires de trames que d'images dans un groupe
25 élémentaire, généralement entre 12 et 24.

Cependant il n'est pas toujours possible de disposer d'autant de ressource mémoire et il faut en pratique se limiter à un nombre inférieur de mémoires de trame.

On sait qu'une image I se suffit à elle-même donc une mémoire de
30 trame suffit pour décoder une image I. On sait aussi qu'une image P nécessite de disposer d'une autre image pour pouvoir être décodée, c'est pourquoi deux mémoires de trame sont nécessaires pour reconstruire une image P depuis sa référence. Enfin on sait qu'une image B a besoin de deux autres images pour

pouvoir être décodée, c'est pourquoi trois mémoires de trame sont nécessaires. Ainsi au minimum, quatre mémoires de trames sont nécessaires et suffisantes (une pour le stockage de l'image en cours d'affichage et trois pour le décodage d'une prochaine image à afficher).

- 5 L'utilisation d'un nombre de mémoires de trames inférieur au nombre total de décodages successifs nécessaires pour décoder une prochaine image à afficher signifie que des images différentes doivent être successivement stockées dans une même mémoire trame, et qu'il faut écraser au fur et à mesure des données correspondant à une image décodée ; cet écrasement
- 10 signifie la perte des données d'une image décompressée de l'image décodée qu'il faudra peut-être à nouveau décoder ultérieurement.

Ci-après, l'invention propose d'appliquer pour le décodage d'une prochaine image à afficher (que l'on appelle ci-dessous "l'image cible"), les critères suivants pour choisir la mémoire de trame à utiliser pour stocker les

15 données d'une nouvelle image dans le cas où aucune des mémoires de trame n'est vide :

- on écrase en priorité les données d'une image qui a déjà été affichée et qui ne sera pas réutilisée pour le décodage d'une prochaine image à afficher, d'après la table d'indexation 11,
- 20 - à défaut, on écrase les données d'une image I ou P non nécessaire au décodage en cours, qui ne doit pas être affichée avant l'image cible et qui présente un coût de décodage minimum, le coût de décodage étant défini ci-dessous.

Dans un exemple de mise en œuvre, le coût de décodage d'une image

25 est de :

- 1 pour une image I,
- $n+1$ pour une image P prédite à partir de n images de référence,
- 1 pour les images B.

La figure 5 représente une liste d'images référencées dans la table

30 d'indexation représentée sur le tableau de la figure 4. Un certain nombre de ces images constituent un GOP. Les flèches représentent comment sont prédites les images de type P et B, les flèches pointant d'une image de référence vers l'image prédite correspondante. Ainsi les images de référence

de B2 sont P3 et I1, de B4 sont P3 et P5, de B6 sont P5 et P7 ; l'image de référence de P3 est I1, celle de P5 est P3 etc. Les coûts en décodage pour les images I1, P3, P5, P7, P9 , P11 sont respectivement 1, 2, 3, 4, 5 et 6. le coût en décodage d'une image de type B est de 1.

5 De préférence, on applique également des règles pour optimiser le nombre d'opérations de décodage à réaliser en fonction de celles qui ont déjà été effectuées et dont le résultat est encore disponible, en comparant le contenu des mémoires de trames à l'image cible à décoder :

10 - si l'image cible est présente dans une au moins des mémoires de trame car elle a servi à un décodage précédent, l'image cible est disponible et n'a pas besoin d'être décodée ;

- sinon on compare le contenu des mémoires de trames à la liste de reconstruction associée à la prochaine image cible et ;

15 - si aucune image de la liste de reconstruction associée n'est présente dans l'une au moins des mémoires de trame, on réalise l'étape 242 à partir de l'image I point d'accès direct de la liste de reconstruction associée ;

20 - sinon on réalise l'étape 242 mais en débutant le décodage MPEG en marche avant à partir de l'image (ou des images dans le cas où l'image cible est une image B) de plus fort coût de décodage de la liste de reconstruction contenue dans une des mémoires de trame de décodage et non plus à partir de l'image I. La liste de reconstruction générée à l'étape 241 ne contient que les autres images à décoder. Par exemple, P11 étant prédite de P9, elle-même de P7, elle-même de P3, elle-même de I1, si P3 est disponible dans une mémoire de trame, alors la liste de reconstruction générée par le décodage de
25 B4 ne comportera que trois images : P7, P9 et P11.

30 Le fait de ne disposer que d'un nombre limité de mémoires de trame nécessite donc une certaine synchronisation entre le processus de décodage et le processus d'affichage. Le nombre d'images à afficher qui sont décodées en avance par rapport à celle en cours d'affichage n'est pas déterministe. En effet, on rappelle que les processus d'affichage et de décodage se font en parallèle mais de façon non indépendante : le module d'affichage affiche la prochaine image à afficher si elle est disponible, sinon il attend qu'elle ait été décodée par le décodeur. Le décodeur, quant à lui, décode les prochaines

images à afficher en avance par rapport à l'affichage, tant que l'ensemble des images identifiées à l'étape 22 ne sont pas prêtes à être affichées, et tant que des mémoires de trames sont disponibles pour ce faire, sinon il attend qu'une mémoire de trame puisse être utilisée.

5 On va maintenant illustrer ces derniers aspects, en considérant à nouveau la portion de flux à décoder et afficher illustrée par le tableau de la figure 4 : le but est de décoder et d'afficher l'image I1 d'identifiant 65532 à l'image B4 d'identifiant 65511 ; la vitesse d'affichage est la vitesse x1 et le nombre de mémoires de trames est égal à 4.

10 Supposons que l'image I1 d'identifiant 65532 est en cours d'affichage, que les mémoires de trame nommées respectivement T1, T2, T3, T4 contiennent respectivement les images I1 d'identifiant 65520 (T1), P5 d'identifiant 65524 (T2), P3 d'identifiant 65522 (T3) et I1 d'identifiant 65532 (T4), et que la prochaine image à afficher est l'image P11 d'identifiant 65530 (l'image d'identifiant 65531 n'étant pas complète). Le but du décodage est
15 d'obtenir l'image P11 d'identifiant 65530 qui n'a pas encore été décodée, puis les images à afficher par la suite tant que des mémoires de trame sont disponibles.

Pour décoder P11 d'identifiant 65530 qui a pour liste de reconstruction
20 {I1, P3, P5, P7, P9, P11}, on démarre le décodage à partir de P5 d'identifiant 65524 qui est disponible dans la mémoire de trame T2. Dans T1, on écrase I1 d'identifiant 65520 qui est l'image non en cours d'affichage ayant le coût de décodage le plus faible pour décoder P7 d'identifiant 65526. Puis on décode P9 d'identifiant 65528 qui est de la même façon est stockée dans T3 en
25 écrasant P3 d'identifiant 65522, qui est alors l'image non encore affichée ayant le coût de décodage le plus faible. Puis on décode P11 d'identifiant 65530 qui est stockée dans T2, en écrasant P5 d'identifiant 65524 pour les mêmes raisons.

Si la synchronisation verticale survient à ce stade, l'image P11
30 d'identifiant 65530 peut être affichée. L'image I1 d'identifiant 65532 n'est plus affichée et n'est plus une image de référence pour une prochaine image à afficher ; elle peut donc être écrasée. La prochaine image à afficher est B10 d'identifiant 65529, décodée à partir de P9 et P11 qui sont disponibles

respectivement dans les mémoires de trames T2 et T3. la liste de reconstruction de B10 ne contient qu'une image, à savoir elle-même. Donc B10 est stockée dans T4 à la place de I1 en attendant la prochaine synchronisation du signal de synchronisation verticale.

- 5 L'image à décoder suivante est P9 d'identifiant 65528, disponible dans la mémoire de trame T2. La prochaine image à décoder est ensuite B8, décodée à partir de P9 et P7. La liste de reconstruction ne contient qu'une seule image : elle-même. Or, à ce stade les mémoires T1, T2, T3, T4 contiennent respectivement les images P7, P9, P11 et B10 et c'est P11 qui est
- 10 en cours d'affichage. Aucune mémoire de trame n'est donc disponible pour stocker B10. Il faut donc attendre la prochaine synchronisation verticale pour décoder cette image, qui sera stockée dans la mémoire de trame T3.

- Avantageusement, on peut partager la mémoire tampon 2 en deux sous-mémoires ou bancs de mémoire et exécuter les étapes 24 et 25 sur une
- 15 première portion du flux de données contenue dans une première sous-mémoire, en même temps que sont exécutées les étapes 22 et 23 sur une seconde portion (venant après dans l'ordre d'affichage en marche arrière) dudit flux de données, cette portion suivante étant contenue dans une seconde sous-mémoire. Lors de cette étape 22, on compare le contenu des deux sous-
- 20 mémoires pour s'assurer que lesdites première et seconde portions du flux sont bien consécutives, et la table d'indexation pour la première portion du flux dans la première sous-mémoire est complétée et mise à jour avec les nouvelles informations relatives à la deuxième portion du flux, tandis que les informations relatives aux images déjà affichées et qui ne sont plus utilisées
- 25 par la suite sont effacées.

Une seule table d'indexation est ainsi considérée et contient les informations utiles concernant les deux portions du flux considérées.

- Avantageusement, les deux portions du flux peuvent se chevaucher partiellement, afin de trouver une image commune sur laquelle la jonction
- 30 puisse être faite, ce qui permet de contourner la difficulté à réaliser des découpages précis du flux. Ceci est avantageux, en particulier pour un flux de données embrouillées.

Chacune des portions du flux de données n'est à nouveau remplacée qu'une fois seulement que l'ensemble des images contenues dans ladite portion a pu être affiché et n'est plus nécessaire pour le décodage des images contenues dans l'autre portion.

5 La figure 6 illustre le contenu d'une table d'indexation d'une première sous-mémoire sur laquelle le processus de décodage a lieu tandis qu'une nouvelle portion de flux illustrée par la figure 7 est chargée dans l'autre sous-mémoire, cette nouvelle portion du flux correspondant aux images à afficher
10 suivantes dans l'ordre d'affichage en marche arrière. La dernière image de la nouvelle portion n'est pas complète mais ce n'est pas problématique car cette image est toujours disponible en tant que première image de l'autre portion. La jonction entre les deux portions de flux est faite au niveau de cette image. Puis l'étape 22 est effectuée en mettant à jour la liste d'indexation de laquelle ont
15 été en outre retirées les images de la première portion de flux qui ont déjà été décodées et affichées : la liste résultante est illustrée par le tableau de la figure 8.

Une fois que les images d'identifiant 66506 à 65510 sont affichées, une nouvelle portion de flux est chargée dans la première sous-mémoire.

20 La taille de chaque sous-mémoire peut être comprise par exemple entre 512 kilooctets et 1 Mégaoctets, ce qui leur permet de stocker au moins un GOP.

25 Le procédé exposé ci-dessus fournit des méthodes pour décoder de façon optimisée toutes les images à afficher en marche arrière. Certaines portions de flux, par exemple celles ayant beaucoup d'images prédites et peu de points d'accès direct nécessiteraient toutefois pour être décodées en
30 totalité, à la fois une capacité mémoire allouée très importante et un décodeur très rapide. L'invention propose par conséquent des critères de décision ayant pour conséquence le non-décodage et donc le non-affichage de certaines portions de flux.

30 Tout d'abord, lorsque deux portions consécutives de flux ne contiennent pas de point d'accès direct, les portions sont considérées comme non décodables et les mémoires tampons sont alors chargées avec de nouvelles portions de flux. En outre, si une erreur dans le flux est détectée par exemple

en causant une violation de syntaxe dans le processus de décodage, l'image concernée est écartée du décodage et également les images à partir de cette image (identifiées dans les étapes 22 et 23) s'il s'agit d'une image I ou P.

Ensuite, certaines images présentant des coûts de décodage élevé (7 ou 8), il est possible qu'un décodeur standard disposant d'une capacité mémoire de quatre mémoires de trame ne puisse fournir au dispositif d'affichage l'image décodée, dans le temps théorique associé à la vitesse d'affichage choisie par l'utilisateur (par exemple 25 images par seconde pour une vitesse d'affichage $x1$). Dans ce cas le dispositif d'affichage gèle l'image en cours d'affichage et continue l'affichage du programme dès que l'image suivante est disponible.

En outre, un mode de mise en œuvre avantageux du procédé de l'invention propose le calcul du retard éventuellement pris par rapport à une vitesse d'affichage commandé et la détermination, à l'étape 241, de la prochaine image à afficher de manière à annuler le retard. L'annulation du retard est obtenue en sautant des images (c'est-à-dire en ne les décodant pas et en ne les affichant pas), par exemple selon les règles de priorité suivantes :

- on saute en priorité une ou plusieurs images B ;
- si le saut d'images B ne suffit pas pour annuler le retard, on saute une ou plusieurs images P ;
- si le saut d'images P ne suffit toujours pas pour annuler le retard, on saute une portion du flux de données compressées lors de la lecture, à l'étape 21, d'une prochaine partie de ce flux de données sur le support numérique de stockage de masse.

Ces mécanismes de calcul de retard et de saut permettent de garder le même algorithme de décodage en marche arrière quelle que soit la vitesse de lecture en marche arrière choisie par l'utilisateur ; en cas de lecture à vitesse rapide, les sauts peuvent être fréquents.

Par exemple, en considérant à nouveau la liste d'images à afficher illustrée par le tableau de la figure 4, on suppose que l'image I1 d'identifiant 65532 est en cours d'affichage, que la vitesse d'affichage en marche arrière est $x1$ et que le décodage de l'image à afficher ensuite, c'est-à-dire P11 d'identifiant 65530, qui a un coût de décodage important. Supposons en outre

que le décodeur a une vitesse relativement lente et fournit l'image P11 décodée avec un certain retard par rapport à la vitesse d'affichage, pendant lequel le dispositif gèle l'image I1. Alors au lieu de décoder l'image B10 d'identifiant 65529, il va directement passer au décodage de l'image P9 d'identifiant 65528.

5 Avec des vitesses d'affichage en marche arrière différentes de la vitesse $x1$, mais avec des rapports entre la vitesse $x1$ et la vitesse sélectionnée de type M/N , où M et N sont entiers, en considérant cette fois-ci l'image au niveau de ses deux trames, on peut déterminer quelle trame est la plus proche de
10 celle à afficher à la vitesse nominale et ainsi apprécier le retard.

Par exemple, avec une vitesse $x2$, on n'affiche qu'une image sur deux, avec une vitesse $x3$, on affiche une image, on saute les deux suivantes. Avec une vitesse $x0,5$, on affiche la même image deux fois.

REVENDEICATIONS

1. Procédé de décodage et d'affichage en marche arrière d'images d'un flux de données d'images compressées selon la norme MPEG utilisant un
5 nombre déterminé N de mémoires de trame adaptées chacune pour stocker une image décodée, où N est un nombre entier supérieur ou égal à 4, caractérisé en ce que les règles suivantes sont appliquées lorsqu'une mémoire de trame doit être écrasée pour permettre le stockage d'une nouvelle image à décoder :
 - 10 - on écrase en priorité une mémoire de trame dans laquelle est stockée une image obsolète c'est-à-dire une image qui a déjà été affichée et qui n'est plus nécessaire pour effectuer un prochain décodage d'image ou, à défaut,
 - on écrase en priorité une mémoire de trame dans laquelle est
15 stockée une image qui présente un coût de décodage minimum,
 - une mémoire de trame dans laquelle est stockée une image qui est en cours d'affichage ne peut pas être écrasée,
 - une mémoire de trame dans laquelle est stockée une image qui est prête pour l'affichage mais n'a pas encore été affichée ne peut pas être
20 écrasée,
 - une mémoire de trame dans laquelle est stockée une image de référence nécessaire pour le décodage de ladite nouvelle image à décoder ne peut pas non plus être écrasée,
 - si aucune mémoire de trame ne peut être écrasée en application
25 des règles précédentes, on suspend le décodage jusqu'à l'affichage d'une image prête pour l'affichage et on évalue à nouveau ces règles après cet affichage.
2. Procédé selon la revendication 1, caractérisé en ce que les images
30 de type I présentent un premier coût de décodage déterminé, en ce que les images de type P prédites à partir d'une image de type I présentent un deuxième coût de décodage, supérieur audit premier coût de décodage, et en ce que les images de type P prédites à partir d'une autre image de type P

présentent un coût de décodage supérieur au coût de décodage de ladite autre image de type P.

3. Procédé selon la revendication 2, caractérisé en ce que les images
5 de type B présentent ledit premier coût de décodage.

4. Procédé selon l'une quelconque des revendications précédentes,
caractérisé en ce qu'on vérifie si une nouvelle image à décoder n'est pas déjà
stockée dans l'une quelconque des mémoires de trame avant de décoder ladite
10 nouvelle image.

5. Procédé selon l'une quelconque des revendications précédentes,
caractérisé en ce qu'il comprend des étapes consistant à :
- déterminer à chaque événement de synchronisation d'affichage
15 d'image si une image prête pour l'affichage est présente dans les mémoires de
trames,
- si oui, afficher ladite image et la marquer comme obsolète si elle
n'est plus nécessaire pour effectuer un décodage d'image,
- et si non, répéter l'affichage de l'image affichée à l'événement de
20 synchronisation d'affichage d'image précédent.

6. Procédé selon l'une quelconque des revendications précédentes,
caractérisé en ce que le flux de données compressées est lu par portions à
partir d'un support numérique de stockage de masse, en réponse à une
25 commande de lecture en marche arrière.

7. Procédé selon l'une quelconque des revendications précédentes,
caractérisé en ce qu'il comprend les étapes consistant à :
a) charger (21) une portion du flux dans une mémoire tampon ;
30 b) analyser (22) ladite portion du flux afin d'identifier un point d'accès
d'un groupe d'images décodable et de déterminer au moins pour chacune des
images dudit groupe, des informations incluant l'adresse de l'image dans la
mémoire tampon, une estampille temporelle de présentation associée à un

ordre d'affichage de l'image en marche avant, et le type I, P ou B de l'image, et stocker lesdites informations dans une table d'indexation ;

c) obtenir une table d'indexation enrichie en complétant la table d'indexation obtenue à l'étape b en précisant pour chaque image de type P ou B sa ou ses images de référence ;

d) à partir de la table d'indexation, déterminer pour une prochaine image à afficher, une liste de reconstruction qui est une liste ordonnée des images de référence à décoder séquentiellement à partir dudit point d'accès pour obtenir ladite prochaine image à afficher ;

e) à partir de ladite liste de reconstruction, obtenir ladite prochaine image à afficher en décodant tout ou partie des images de ladite liste de reconstruction.

8. Procédé selon la revendication 7, caractérisé en ce qu'il comprend, entre l'étape b) et l'étape c), l'attribution (23) à chacune des images codées dans ladite première portion du flux d'un numéro d'identification unique permettant de l'identifier de manière univoque dans le flux.

9. Procédé selon la revendication 8, suivant lequel, à l'étape d), la prochaine image à afficher est déterminée (25) à partir de son numéro d'identification.

10. Procédé selon l'une quelconque des revendications 7 à 9, caractérisé en ce que, pour obtenir la prochaine image à afficher, on vérifie si l'une au moins des images de la liste de reconstruction associée à la prochaine image à afficher n'est pas déjà stockée dans l'une quelconque des mémoires de trame, et en ce que, dans l'affirmative, on commence à décoder séquentiellement les images de ladite liste non pas à partir du point d'accès du groupe d'images décodable mais à partir de ladite image déjà stockée, ou, le cas échéant, à partir de celle desdites images déjà stockées qui présente le coût de décodage le plus élevé.

11. Procédé selon l'une quelconque des revendications 7 à 10, caractérisé en ce que, à chaque fois qu'une image est affichée, on calcule un retard par rapport à une vitesse d'affichage théorique, et en ce que, à l'étape d), on détermine la prochaine image à afficher de manière à annuler le retard
- 5 en sautant des images, selon les règles de priorité suivantes :
- on saute en priorité une ou plusieurs images B ;
 - si un ou des sauts d'images B ne suffit pas pour annuler le retard, on saute une ou plusieurs images P ;
 - si un ou des sauts d'images P ne suffit pas pour annuler le retard,
- 10 on saute une ou plusieurs portions du flux de données lors du chargement, à l'étape a), d'une prochaine portion du flux de données dans la mémoire tampon.
12. Procédé selon l'une quelconque des revendications 7 à 11, caractérisé en ce que la mémoire tampon comprend un premier et un second bancs de mémoire, et en ce que les étapes d) et e) sont réalisées dans ledit premier banc de mémoire de la mémoire tampon pour une première portion du flux, pendant que les étapes a) à c) sont réalisées dans ledit second banc de mémoire de la mémoire tampon pour une seconde portion du flux.
- 20 13. Procédé selon la revendication 12, caractérisé en ce que, l'un ou l'autre des premier et second bancs de mémoire n'est pas écrasé à l'étape a) tant que les données qu'il contient sont encore nécessaires au traitement des données contenues dans l'autre desdits premier et second bancs de mémoire.
- 25 14. Procédé selon la revendication 12 ou la revendication 13, caractérisé en ce que l'un ou l'autre des premier et second bancs de mémoire est écrasé avec une nouvelle portion du flux à l'étape a), dès que les données qu'il contient ne sont plus nécessaires au traitement des données contenues
- 30 dans l'autre desdits premier et second bancs de mémoire.
15. Procédé selon l'une quelconque des revendications 12 à 14, caractérisé en ce que les portions de flux chargées dans le premier banc de

mémoire et dans le second banc de mémoire se chevauchent, et en ce qu'une jonction entre ces deux portions peut être réalisée entre les deux parties de flux au niveau d'une image commune.

- 5 16. Procédé selon les revendications 12 à 15, caractérisé en ce que les tables d'indexation relatives à chacune des portions de flux chargées dans le premier banc de mémoire et dans le second banc de mémoire constituent une seule table d'indexation
- 10 17. Dispositif de décodage et d'affichage en marche arrière d'images d'un flux de données d'images compressées selon la norme MPEG comprenant un nombre déterminé N de mémoires de trame adaptée chacune pour stocker une image décodée, où N est un nombre entier supérieur ou égal à 4, caractérisé en ce qu'il comprend des moyens pour appliquer les règles
- 15 suivantes lorsqu'une mémoire de trame doit être écrasée pour permettre le stockage d'une nouvelle image à décoder :
 - on écrase en priorité une mémoire de trame dans laquelle est stockée une image qui a déjà été affichée et qui n'est plus nécessaire pour effectuer un décodage d'image ou, à défaut,
 - 20 - on écrase en priorité une mémoire de trame dans laquelle est stockée une image qui présente un coût de décodage minimum,
 - une mémoire de trame dans laquelle est stockée une image qui est en cours d'affichage ne peut pas être écrasée,
 - une mémoire de trame dans laquelle est stockée une image qui est
 - 25 prête pour l'affichage mais n'a pas encore été affichée ne peut pas être écrasée,
 - une mémoire de trame dans laquelle est stockée une image de référence nécessaire pour le décodage de ladite nouvelle image à décoder ne peut pas non plus être écrasée,
 - 30 - si aucune mémoire de trame ne peut être écrasée en application des règles précédentes, on suspend le décodage jusqu'à l'affichage d'une image prête pour l'affichage et on évalue à nouveau ces règles après cet affichage.

18. Dispositif selon la revendication 17, caractérisé en ce qu'il comprend des moyens de calcul du coût en décodage d'une image donnée, les images de type I présentant un premier coût de décodage déterminé, les images de type P prédites à partir d'une image de type I présentant un deuxième coût de décodage, supérieur audit premier coût de décodage, et les images de type P prédites à partir d'une autre image de type P présentant un coût de décodage supérieur au coût de décodage de ladite autre image de type P.
- 10 19. Dispositif selon la revendication 18, caractérisé en ce que les moyens de calcul du coût en décodage fournissent ledit premier coût de décodage pour les images de type B.
- 15 20. Dispositif selon l'une quelconque des revendications 17 à 19, caractérisé en ce qu'il comprend des moyens pour vérifier qu'une nouvelle image à décoder n'est pas déjà stockée dans l'une quelconque des mémoires de trame avant de décoder ladite nouvelle image.
- 20 21. Dispositif selon l'une quelconque des revendications 17 à 20, caractérisé en ce qu'il comprend des moyens pour :
- déterminer à chaque événement de synchronisation d'affichage d'image si une image prête pour l'affichage est présente dans les mémoires de trames,
 - si oui, afficher ladite image et la marquer comme obsolète si elle
 - 25 n'est plus nécessaire pour effectuer un décodage d'image,
 - et si non, répéter l'affichage de l'image affichée à l'événement de synchronisation d'affichage d'image précédent.
- 30 22. Dispositif selon l'une quelconque des revendications 17 à 21, caractérisé en ce qu'il comprend des moyens pour lire le flux de données compressées par portions à partir d'un support numérique de stockage de masse, en réponse à une commande de lecture en marche arrière.

23. Dispositif selon l'une quelconque des revendications 17 à 22, caractérisé en ce qu'il comprend :

a) des moyens pour charger une portion du flux dans une mémoire tampon ;

5 b) des moyens pour analyser ladite portion du flux afin d'identifier un point d'accès d'un groupe d'images décodable et de déterminer au moins pour chacune des images dudit groupe, des informations incluant l'adresse de l'image dans la mémoire tampon, une estampille temporelle de présentation associée à un ordre d'affichage de l'image en marche avant, et le type I, P ou B de l'image, et stocker lesdites informations dans une table d'indexation ;

10 c) des moyens pour obtenir une table d'indexation enrichie en complétant la table d'indexation obtenue à l'étape b en précisant pour chaque image de type P ou B sa ou ses images de référence ;

 d) des moyens pour déterminer à partir de la table d'indexation, pour
15 une prochaine image à afficher, une liste de reconstruction qui est une liste ordonnée des images de référence à décoder séquentiellement à partir dudit point d'accès pour obtenir ladite prochaine image à afficher ;

 e) des moyens pour obtenir à partir de ladite liste de reconstruction, ladite prochaine image à afficher en décodant tout ou partie des images de
20 ladite liste de reconstruction.

24. Dispositif selon la revendication 23 caractérisé en ce que lesdits moyens pour attribuer les numéros d'identification aux images codées dans le flux attribuent lesdits numéros de façon séquentielle, suivant l'ordre d'affichage
25 des images.

25. Dispositif selon la revendication 24, suivant lequel lesdits moyens pour déterminer déterminent la prochaine image à afficher à partir de son numéro d'identification.

30

26. Dispositif selon l'une quelconque des revendications 23 à 25, caractérisé en ce que lesdits moyens pour obtenir la prochaine image à afficher comprennent des moyens pour vérifier si l'une au moins des images de la liste

de reconstruction associée à la prochaine image à afficher n'est pas déjà stockée dans l'une quelconque des mémoires de trame, et, dans l'affirmative, des moyens pour commencer à décoder séquentiellement les images de ladite liste non pas à partir du point d'accès du groupe d'images décodable mais à
5 partir de ladite image déjà stockée, ou, le cas échéant, à partir de celle desdites images déjà stockées qui présente le coût de décodage le plus élevé.

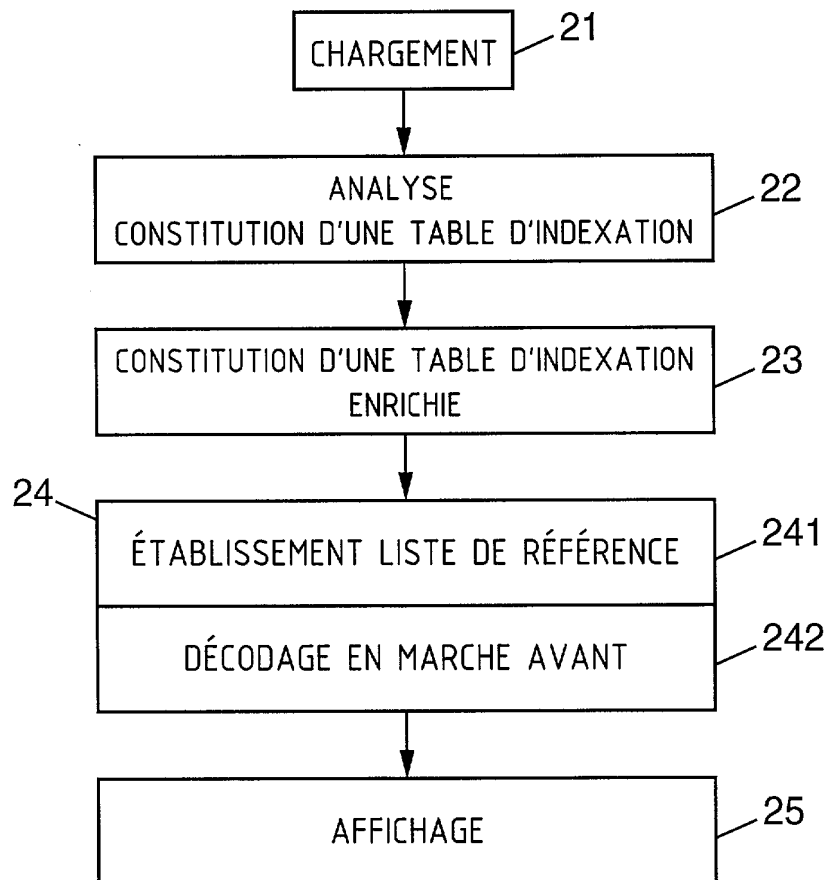
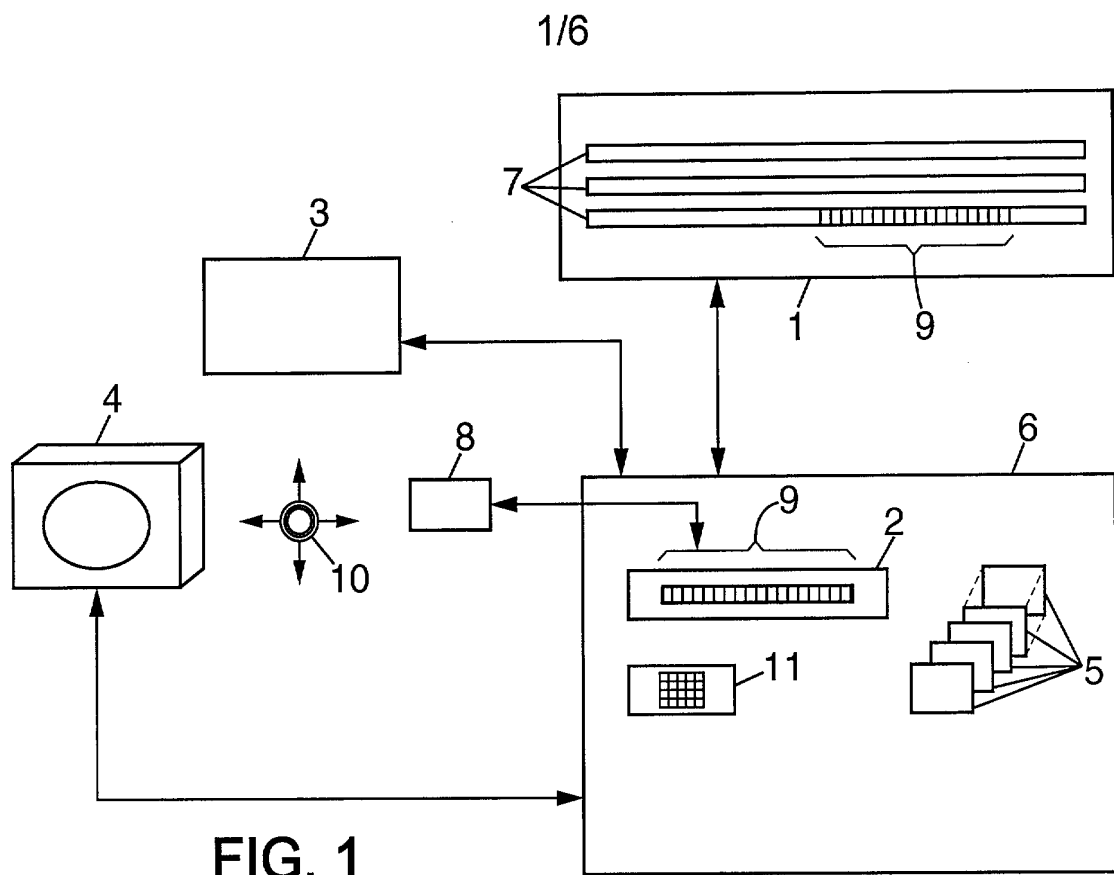
27. Dispositif selon l'une quelconque des revendications 23 à 26, caractérisé en ce qu'il comprend des moyens pour, à chaque fois qu'une image
10 est affichée, calculer un retard par rapport à une vitesse d'affichage théorique, et des moyens pour déterminer la prochaine image à afficher de manière à annuler le retard en sautant des images selon les règles de priorité suivantes :

- on saute en priorité une ou plusieurs images B ;
- si un ou des sauts d'images B ne suffit pas pour annuler le retard,
15 on saute une ou plusieurs images P ;
- si un ou des sauts d'images P ne suffit pas pour annuler le retard, on saute une ou plusieurs portions du flux de données lors du chargement d'une prochaine portion du flux de données dans la mémoire tampon.

20 28. Dispositif selon l'une quelconque des revendications 23 à 27, caractérisé en ce que la mémoire tampon comprend un premier et un second bancs de mémoire, et en ce que lesdits moyens pour changer et lesdits moyens pour analyser opèrent dans ledit premier banc de mémoire de la mémoire tampon pour une première portion du flux, pendant que lesdits
25 moyens pour obtenir la table d'indexation, et lesdits moyens pour obtenir la prochaine image à afficher opèrent dans ledit second banc de mémoire de la mémoire tampon pour une seconde portion du flux.

29. Dispositif selon la revendication 28, caractérisé en ce qu'il comprend
30 des moyens pour empêcher d'écraser l'un ou l'autre des premier et second bancs de mémoire tant que les données qu'il contient sont encore nécessaires au traitement des données contenues dans l'autre desdits premier et second bancs de mémoire.

30. Dispositif selon la revendication 28 ou la revendication 29, caractérisé en ce qu'il comprend des moyens pour écraser l'un ou l'autre des premier et second bancs de mémoire avec une nouvelle portion du flux à l'étape a), dès que les données qu'il contient ne sont plus nécessaires au traitement des données contenues dans l'autre desdits premier et second bancs de mémoire.
31. Dispositif selon l'une quelconque des revendications 28 à 30, caractérisé en ce qu'il comprend des moyens pour que les portions de flux chargées dans le premier banc de mémoire et dans le second banc de mémoire se chevauchent, et pour pouvoir réaliser une jonction entre ces deux portions au niveau d'une image commune les deux parties de flux.
32. Dispositif selon les revendications 28 à 31, caractérisé en ce qu'il comprend des moyens pour réunir en une seule table d'indexation les tables d'indexation relatives à chacune des portions de flux chargées dans le premier banc de mémoire et dans le second banc de mémoire.
33. Circuit pilote vidéo caractérisé en ce qu'il comprend un dispositif selon l'une des revendications 17 à 32.
34. Boîtier décodeur caractérisé en ce qu'il comprend un circuit pilote vidéo selon la revendication 33.



N°	adresse	type/estampille temporelle/point d'accès direct
0	: 0000096a	(P11)
1	: 000057e6	(B10)
2	: 00007ba9	(I 1) *
3	: 000155b6	(B 0)
4	: 00017932	(P 3)
5	: 0001c60e	(B 2)
6	: 0001ee1e	(P 5)
7	: 00023526	(B 4)
8	: 00025dde	(P 7)
9	: 0002a776	(B 6)
10	: 0002cea2	(P 9)
11	: 00031816	(B 8)
12	: 00033cf6	(P11)
13	: 000386aa	(B10)
14	: 0003af9d	(I 1) *
15	: 00049876	(B 0)
16	: 0004bf5e	(P 3)
17	: 000509b2	(B 2)
18	: 0005300e	(P 5)
19	: 000578de	(B 4)
20	: 00059e66	(P 7)
21	: 0005e7e2	(B 6)
22	: 00060b0e	(P 9)
23	: 0006559e	(B 8)
24	: 00067bce	(P11)
25	: 0006c0ce	(B10)
26	: 0006e9e1	(I 1) *
27	: 0007d626	(B 0)

FIG. 3

3/6

N	adresse	type/estampille	identifiant Unique	référence antérieure	référence postérieure
0 :	0000096a	(P11)	(65506)	???? N	????
1 :	000057e6	(B10)	(65505)	???? 0(P11)	????
2 :	00007ba9	(I 1) *	(65508)	N N	
3 :	000155b6	(B 0)	(65507)	0(P11) 2(I 1)	????
4 :	00017932	(P 3)	(65510)	2(I 1) N	
5 :	0001c60e	(B 2)	(65509)	2(I 1) 4 (P 3)	
6 :	0001ee1e	(P 5)	(65512)	4(P 3) N	
7 :	00023526	(B 4)	(65511)	4(P 3) 6 (P 5)	
8 :	00025dde	(P 7)	(65514)	6(P 5) N	
9 :	0002a776	(B 6)	(65513)	6(P 5) 8 (P 7)	
10 :	0002cea2	(P 9)	(65516)	8(P 7) N	
11 :	00031816	(B 8)	(65515)	8(P 7) 10(P 9)	
12 :	00033cf6	(P11)	(65518)	10(P 9) N	
13 :	000386aa	(B10)	(65517)	10(P 9) 12(P11)	
14 :	0003af9d	(I 1) *	(65520)	N N	
15 :	00049876	(B 0)	(65519)	12(P11) 14(I 1)	
16 :	0004bf5e	(P 3)	(65522)	14(I 1) N	
17 :	000509b2	(B 2)	(65521)	14(I 1) 16(P 3)	
18 :	0005300e	(P 5)	(65524)	16(P 3) N	
19 :	000578de	(B 4)	(65523)	16(P 3) 18(P 5)	
20 :	00059e66	(P 7)	(65526)	18(P 5) N	
21 :	0005e7e2	(B 6)	(65525)	18(P 5) 20(P 7)	
22 :	00060b0e	(P 9)	(65528)	20(P 7) N	
23 :	0006559e	(B 8)	(65527)	20(P 7) 22(P 9)	
24 :	00067bce	(P11)	(65530)	22(P 9) N	
25 :	0006c0ce	(B10)	(65529)	22(P 9) 24(P11)	
26 :	0006e9e1	(I 1) *	(65532)	N N	
27 :	0007d626	(B 0)	(65531)	24(P11) 26(I 1)	????

FIG. 4

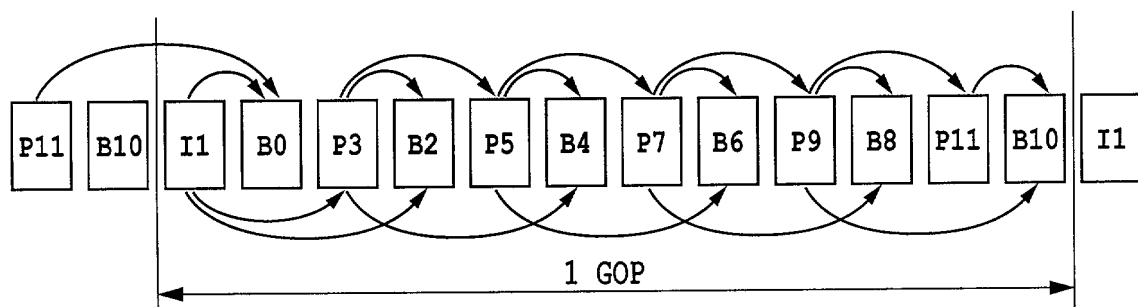


FIG. 5

N	adresse	type/estampille	PTS identifiant unique
0 :	0000096a	(P11)	0x0ef86cb (65506)
1 :	000057e6	(B10)	0x0ef78bb (65505)
2 :	00007ba9	(I 1)	0x0efa2eb (65508)
3 :	000155b6	(B 0)	0x0ef94db (65507)
4 :	00017932	(P 3)	0x0efbf0b (65510)
5 :	0001c60e	(B 2)	0x0efb0fb (65509)
6 :	0001ee1e	(P 5)	0x0efdb2b (65512)
7 :	00023526	(B 4)	0x0efcd1b (65511)
8 :	00025dde	(P 7)	0x0eff74b (65514)
9 :	0002a776	(B 6)	0x0efe93b (65513)
10 :	0002cea2	(P 9)	0x0f0136b (65516)
11 :	00031816	(B 8)	0x0f0055b (65515)
12 :	00033cf6	(P11)	0x0f02f8b (65518)
13 :	000386aa	(B10)	0x0f0217b (65517)
14 :	0003af9d	(I 1)	0x0f04bab (65520)
15 :	00049876	(B 0)	0x0f03d9b (65519)
16 :	0004bf5e	(P 3)	0x0f067cb (65522)
17 :	000509b2	(B 2)	0x0f059bb (65521)
18 :	0005300e	(P 5)	0x0f083eb (65524)
19 :	000578de	(B 4)	0x0f075db (65523)
20 :	00059e66	(P 7)	0x0f0a00b (65526)
21 :	0005e7e2	(B 6)	0x0f091fb (65525)
22 :	00060b0e	(P 9)	0x0f0bc2b (65528)
23 :	0006559e	(B 8)	0x0f0ae1b (65527)
24 :	00067bce	(P11)	0x0f0d84b (65530)
25 :	0006c0ce	(B10)	0x0f0ca3b (65529)
26 :	0006e9e1	(I 1)	0x0f0f46b (65532)
27 :	0007d626	(B 0)	0x0f0e65b (65531)

FIG. 6

N	adresse	type/estampille	PTS	identifiant unique
0	: 00080000	(P 3)	0x0edc4cb	(???)
1	: 00083dc2	(B 2)	0x0edb6bb	(???)
2	: 00086436	(P 5)	0x0ede0eb	(???)
3	: 0008acf2	(B 4)	0x0edd2db	(???)
4	: 0008d122	(P 7)	0x0edfd0b	(???)
5	: 00091bea	(B 6)	0x0edeefb	(???)
6	: 00093fe2	(P 9)	0x0ee192b	(???)
7	: 00098b8a	(B 8)	0x0ee0b1b	(???)
8	: 0009b0fa	(P11)	0x0ee354b	(???)
9	: 0009f92a	(B10)	0x0ee273b	(???)
10	: 000a21c9	(I 1)	0x0ee516b	(???)
11	: 000affd6	(B 0)	0x0ee435b	(???)
12	: 000b241e	(P 3)	0x0ee6d8b	(???)
13	: 000b6f76	(B 2)	0x0ee5f7b	(???)
14	: 000b93ba	(P 5)	0x0ee89ab	(???)
15	: 000bdbf6	(B 4)	0x0ee7b9b	(???)
16	: 000c038e	(P 7)	0x0eea5cb	(???)
17	: 000c4866	(B 6)	0x0ee97bb	(???)
18	: 000c71de	(P 9)	0x0eec1eb	(???)
19	: 000cb90e	(B 8)	0x0eeb3db	(???)
20	: 000ce0c2	(P11)	0x0eede0b	(???)
21	: 000d2b7a	(B10)	0x0eecffb	(???)
22	: 000d51b1	(I 1)	0x0eefa2b	(???)
23	: 000e2202	(B 0)	0x0eeec1b	(???)
24	: 000e45c6	(P 3)	0x0ef164b	(???)
25	: 000e922e	(B 2)	0x0ef083b	(???)
26	: 000eb922	(P 5)	0x0ef326b	(???)
27	: 000eff92	(B 4)	0x0ef245b	(???)
28	: 000f2a46	(P 7)	0x0ef4e8b	(???)
29	: 000f6fda	(B 6)	0x0ef407b	(???)
30	: 000f99b2	(P 9)	0x0ef6aab	(???)
31	: 000fe402	(B 8)	0x0ef5c9b	(???)
32	: 000ff800	(P11)	0x0ef86cb	(???)

FIG. 7

N	adresse	type/estampille	PTS identifiant unique
0 :	00080000	(P 3)	0x0edc4cb (65474)
1 :	00083dc2	(B 2)	0x0edb6bb (65473)
2 :	00086436	(P 5)	0x0ede0eb (65476)
3 :	0008acf2	(B 4)	0x0edd2db (65475)
4 :	0008d122	(P 7)	0x0edfd0b (65478)
5 :	00091bea	(B 6)	0x0edeefb (65477)
6 :	00093fe2	(P 9)	0x0ee192b (65480)
7 :	00098b8a	(B 8)	0x0ee0b1b (65479)
8 :	0009b0fa	(P11)	0x0ee354b (65482)
9 :	0009f92a	(B10)	0x0ee273b (65481)
10 :	000a21c9	(I 1)	0x0ee516b (65484)
11 :	000affd6	(B 0)	0x0ee435b (65483)
12 :	000b241e	(P 3)	0x0ee6d8b (65486)
13 :	000b6f76	(B 2)	0x0ee5f7b (65485)
14 :	000b93ba	(P 5)	0x0ee89ab (65488)
15 :	000bdbf6	(B 4)	0x0ee7b9b (65487)
16 :	000c038e	(P 7)	0x0eea5cb (65490)
17 :	000c4866	(B 6)	0x0ee97bb (65489)
18 :	000c71de	(P 9)	0x0eec1eb (65492)
19 :	000cb90e	(B 8)	0x0eeb3db (65491)
20 :	000ce0c2	(P11)	0x0eede0b (65494)
21 :	000d2b7a	(B10)	0x0eecffb (65493)
22 :	000d51b1	(I 1)	0x0eefa2b (65496)
23 :	000e2202	(B 0)	0x0eeec1b (65495)
24 :	000e45c6	(P 3)	0x0ef164b (65498)
25 :	000e922e	(B 2)	0x0ef083b (65497)
26 :	000eb922	(P 5)	0x0ef326b (65500)
27 :	000eff92	(B 4)	0x0ef245b (65499)
28 :	000f2a46	(P 7)	0x0ef4e8b (65502)
29 :	000f6fda	(B 6)	0x0ef407b (65501)
30 :	000f99b2	(P 9)	0x0ef6aab (65504)
31 :	000fe402	(B 8)	0x0ef5c9b (65503)
+-----+			
32 :	0000096a	(P11)	0x0ef86cb (65506)
33 :	000057e6	(B10)	0x0ef78bb (65505)
34 :	00007ba9	(I 1)	0x0efa2eb (65508)
35 :	000155b6	(B 0)	0x0ef94db (65507)
36 :	00017932	(P 3)	0x0efbf0b (65510)

FIG. 8



RAPPORT DE RECHERCHE PRÉLIMINAIRE

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

**N° d'enregistrement
national**

FA 627528
FR 0216328

DOCUMENTS CONSIDÉRÉS COMME PERTINENTS		Revendication(s) concernée(s)	Classement attribué à l'invention par l'INPI
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes		
A	EP 1 005 226 A (SONY CORPORATION) 31 mai 2000 (2000-05-31) * le document en entier *	1,17	H04N7/32 H04N7/167
D,A	WO 01 37572 A (KONINKLIJKE PHILIPS ELECTRONICS N.V.) 25 mai 2001 (2001-05-25) * le document en entier *	1,17	
A	EP 1 003 339 A (SONY CORPORATION) 24 mai 2000 (2000-05-24) * le document en entier *	1,17	
			DOMAINES TECHNIQUES RECHERCHÉS (Int.CL.7)
			H04N
Date d'achèvement de la recherche		Examineur	
24 septembre 2003		Verleye, J.	
CATÉGORIE DES DOCUMENTS CITÉS		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	
X : particulièrement pertinent à lui seul Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : arrière-plan technologique O : divulgation non-écrite P : document intercalaire			

ANNEXE AU RAPPORT DE RECHERCHE PRÉLIMINAIRE**RELATIF A LA DEMANDE DE BREVET FRANÇAIS NO. FR 0216328 FA 627528**

La présente annexe indique les membres de la famille de brevets relatifs aux documents brevets cités dans le rapport de recherche préliminaire visé ci-dessus.

Les dits membres sont contenus au fichier informatique de l'Office européen des brevets à la date du **24-09-2003**

Les renseignements fournis sont donnés à titre indicatif et n'engagent pas la responsabilité de l'Office européen des brevets, ni de l'Administration française

Document brevet cité au rapport de recherche		Date de publication	Membre(s) de la famille de brevet(s)	Date de publication
EP 1005226	A	31-05-2000	JP 2000165812 A	16-06-2000
			JP 2000175151 A	23-06-2000
			CN 1254917 A	31-05-2000
			EP 1005226 A2	31-05-2000
			KR 2000035681 A	26-06-2000
WO 0137572	A	25-05-2001	WO 0137572 A1	25-05-2001
			EP 1153511 A1	14-11-2001
			JP 2003515290 T	22-04-2003
EP 1003339	A	24-05-2000	JP 8079702 A	22-03-1996
			JP 3248366 B2	21-01-2002
			JP 8079688 A	22-03-1996
			EP 1003339 A1	24-05-2000
			AT 204111 T	15-08-2001
			AU 699861 B2	17-12-1998
			AU 3043095 A	21-03-1996
			AU 711638 B2	21-10-1999
			AU 8958898 A	07-01-1999
			BR 9503861 A	17-09-1996
			CN 1139796 A , B	08-01-1997
			DE 69522059 D1	13-09-2001
			DE 69522059 T2	21-03-2002
			EP 0700220 A2	06-03-1996
			TW 400514 B	01-08-2000
			US 5771331 A	23-06-1998
			US 6009231 A	28-12-1999