

12 **DEMANDE DE BREVET EUROPEEN**

21 Numéro de dépôt: 84201491.2

51 Int. Cl.⁴: G 05 G 1/16

22 Date de dépôt: 16.10.84

30 Priorité: 19.10.83 FR 8316637

43 Date de publication de la demande:
19.06.85 Bulletin 85/25

84 Etats contractants désignés:
DE FR GB

71 Demandeur: LA RADIOTECHNIQUE, Société Anonyme
dite:
51, rue Carnot BP 301
F-92156 Suresnes Cedex(FR)

84 Etats contractants désignés:
FR

71 Demandeur: N.V. Philips' Gloeilampenfabrieken
Groenewoudseweg 1
NL-5621 BA Eindhoven(NL)

84 Etats contractants désignés:
DE GB

72 Inventeur: Bottiau, Michel Société Civile S.P.I.D.
209, rue de l'Université
F-75007 Paris(FR)

74 Mandataire: Pinchon, Pierre et al,
Société Civile S.P.I.D. 209 rue de L'Université
F-75007 Paris(FR)

54 Procédé pour engendrer des objets dans une image vidéo.

57 Le procédé de l'invention concerne la génération d'une pluralité d'objets mobiles ou fixes sur un écran vidéo, particulièrement dans le cadre de systèmes de jeux.

Il utilise une mémoire topologique 20 représentant l'image point à point et une mémoire de formes 31 dans laquelle les objets sont décrits par segments successifs placés à la suite les uns des autres dans la mémoire. Un générateur d'images 26 exploite le contenu de la mémoire de formes 31 en fonction de celui de la mémoire topologique 20 qui est lue en temps réel par l'interface 21, et fournit à chaque instant l'indication de la couleur à reproduire sur l'écran.

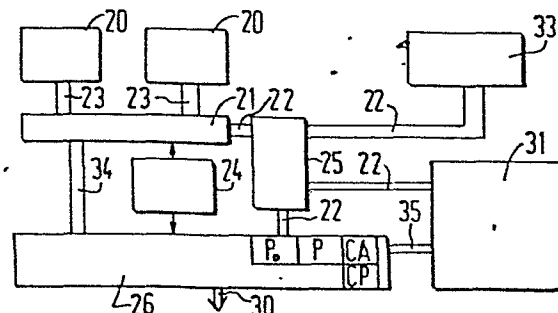


FIG.5

"PROCEDE POUR ENGENDRER DES OBJETS DANS UNE IMAGE VIDEO"

La présente invention concerne un procédé pour engendrer dans une image décrite par une succession de lignes en liaison avec des signaux de synchronisation de ligne et de trame, des objets représentés par une série
5 de segments les uns au-dessus des autres, chacun correspondant à une ligne de l'image.

Elle concerne, de manière générale, l'affichage d'images vidéo et plus particulièrement la génération d'une pluralité d'objets mobiles sur un écran vidéo, dans le cadre de systèmes de jeux vidéo, dans lesquels on utilise un
10 microprocesseur pour gérer l'ensemble d'un système et faire évoluer la position des objets en modifiant le contenu des mémoires.

Pour placer l'image d'un objet mobile dans une trame
15 de télévision, les générateurs vidéo connus comparent les coordonnées X et Y du premier point significatif du mobile au contenu de compteurs qui suivent la position du spot d'un tube à écran cathodique. Un tel procédé est décrit par exemple dans le brevet des Etats-Unis d'Amérique
20 US 4 116 444 dans lequel un élément soustracteur et comparateur contrôle pour chaque ligne d'image si celle-ci comporte un objet, en lisant à chaque fois le contenu entier d'une mémoire contenant les coordonnées des objets.

Un tel procédé est rapidement limité par la vitesse
25 des circuits car un grand nombre de comparaisons de coordonnées et de calculs doit être réalisé au cours de la description de l'image, ce qui limite le nombre d'objets différents pouvant être représentés ensemble. Le procédé décrit dans le document cité apporte une amélioration en
30 ce qu'il prépare à l'avance tout l'affichage d'une ligne

dans une mémoire dite "horizontal memory". Cette mémoire doit cependant être complètement remise à jour pour chaque ligne de l'image, et on dispose seulement pour cela de la durée d'un retour de ligne.

5 Il a également été proposé d'utiliser une mémoire représentant le contenu complet de l'écran.

Le brevet des Etats-Unis d'Amérique US 4 165 072 décrit une telle solution. Ce document indique (colonne 3, dernier paragraphe) qu'une telle architecture est idéale
10 pour deux niveaux de vidéo (par opposition à de la couleur) parce qu'une mémoire du contenu complet de l'écran enregistre "de façon inhérente" deux niveaux de données binaires. Le document mentionne que, cependant, en doublant le volume mémoire, quatre niveaux sont possibles ; en triplant
15 la mémoire, on aurait huit niveaux, etc... Une telle solution conduirait donc dans le cas de la couleur, et avec 4 bits seulement pour la définition de la couleur, à des mémoires de 256 K octets pour décrire une image de 256 x 256 points, ce qui est rédhibitoire. De plus, ces mémoires
20 devraient être très rapides...

Le procédé selon la présente invention évite ces inconvénients et permet de traiter aisément un grand nombre d'objets ensemble. Il permet en outre de traiter un paysage de fond selon la même procédure que les objets mobiles, ce
25 qui simplifie la programmation de l'image complète et offre des possibilités supplémentaires d'animation.

Il est basé sur l'idée d'introduire dans une mémoire de contenu complet de l'écran, à la place de bits décrivant directement chaque point d'image ou pixel, des numéros placés
30 seulement aux endroits où commence un objet et renvoyant à une description des objets dans une autre mémoire.

Le procédé selon la présente invention est ainsi notamment remarquable en ce que :

- on enregistre dans au moins une mémoire dite "topologique"
35 gique" pour chaque ligne de l'image au cours de laquelle on rencontre un objet, un groupe de données indiquant le début du segment de l'objet au niveau de cette ligne et

comportant un numéro d'identification de cet objet, ce groupe de données ayant une position dans la mémoire qui correspond sensiblement à la position du segment dans l'image,

5 - on enregistre dans une mémoire dite "de formes" des chaînes de données constituant la description desdits segments constituant les objets,

 - on enregistre dans un registre d'adresses, associé à la mémoire topologique, à des positions correspondant respectivement à chaque numéro d'objet, des adresses de début et de départ de la description desdits objets dans la
10 mémoire de formes,

 - on lit la mémoire topologique en synchronisme avec la description de l'image,

15 - lorsqu'on y trouve l'indication du début d'un segment et le numéro de l'objet dont il fait partie, on se reporte à la position correspondant audit numéro dans le registre d'adresses pour y trouver l'adresse à laquelle se trouve, dans la mémoire de formes, la description du segment qui
20 doit être représenté, et

 - enfin, on lit la chaîne de données décrivant ce segment, en synchronisme avec la description de l'image.

Avantageusement, ledit registre d'adresses est constitué de deux piles, l'une contenant pour chaque numéro
25 d'objet l'adresse, fixée au moins pour la durée d'une trame, du début de la description dudit objet dans la mémoire de formes, et l'autre contenant une adresse de départ correspondant à la description d'un segment de l'objet, adresse de départ qui est modifiée chaque fois qu'un nouveau segment différent du précédent doit être représenté.
30

Dans une variante particulièrement avantageuse du procédé, au lieu d'enregistrer les groupes de données indiquant le début d'un segment à la place exacte, où commence ce segment, on les enregistre en des positions prédéterminées selon une grille régulière dans la mémoire topologique,
35 et ces groupes de données comportent, outre le numéro d'objet, une série de bits indiquant la valeur du décalage

entre la position exacte du segment et la position du groupe de données.

Pour soulager le microprocesseur d'une partie répétitive des tâches de mise à jour des mémoires, on utilise
5 avantageusement, pour modifier le contenu de la mémoire topologique, un automate qui arrange ledit numéro et ladite valeur du décalage selon un format prédéterminé, et charge ce groupe de données en parallèle dans la mémoire topologique. Ainsi, ces tâches sont considérablement allégées et
10 peuvent être effectuées avantageusement pendant les retours de trame de l'image.

Une amélioration qui permet de gagner de la place en mémoire consiste en ce qu'on enregistre en outre dans chaque groupe de données de la mémoire topologique un bit dit
15 "de répétition" indiquant le cas échéant que, lorsqu'on retrouvera ultérieurement le même numéro d'objet, c'est le même segment qui devra alors être à nouveau représenté, et en ce que, après avoir lu l'adresse de départ dans le registre d'adresses, on y laisse cette même adresse dans le
20 cas d'une répétition, alors qu'on la remplace par celle du segment suivant dans le cas contraire.

Afin d'éviter la perte de l'adresse du segment suivant lorsque la description d'un segment est interrompue par une fin de ligne, il est avantageux d'enregistrer l'adresse de la chaîne de données décrivant le segment suivant
25 en tête de chaque description de segment dans la mémoire de formes.

Lorsqu'un objet qui se déplace sort progressivement de l'image par son bord gauche, l'indication de la présence
30 de cet objet disparaît de la mémoire topologique, puisque c'est le début de chaque segment, c'est-à-dire son extrémité gauche, qui y est indiquée. Pour remédier à cet inconvénient, le procédé est remarquable en ce qu'en outre on subdivise la description de chaque segment dans la mémoire de
35 formes en sous-segments dont la représentation dans l'image a une longueur déterminée constante, sauf pour le dernier sous-segment d'un segment, en ce qu'en tête de chaque chaîne

de données décrivant un sous-segment, on place l'adresse du sous-segment correspondant du segment suivant, et en ce qu'on prévoit dans la mémoire topologique des positions correspondant à des portions de ligne d'image fictives non
5 représentables et situées à gauche et à l'extérieur de l'image.

En outre, pour gagner de la place dans la mémoire de formes, lesdites chaînes de données dans la mémoire de formes sont constituées avantageusement de blocs ayant tous
10 la même longueur et comportant au moins un bloc d'adresse contenant l'adresse de la chaîne de données décrivant le segment suivant, et un bloc dit "bloc de pixels" contenant
- des indications sur sa propre position par rapport aux autres blocs,
15 - et la description d'une succession de points définie par leur couleur et leur nombre.

Pour résoudre les problèmes de collisions entre objets, c'est-à-dire d'objets que leur mouvement entraîne à se recouvrir au moins partiellement l'un l'autre, il est
20 avantageux d'utiliser plusieurs mémoires topologiques correspondant chacune à un plan d'image et qui exploitent ensemble une mémoire de formes commune, et d'établir alors des priorités entre les mémoires topologiques, c'est-à-dire entre les numéros d'objets à représenter issus de la lec-
25 ture simultanée des différentes mémoires topologiques.

La description qui va suivre en regard des figures annexées, données à titre d'exemple non limitatif, fera bien comprendre comment l'invention peut être réalisée.

La figure 1 illustre la manière de représenter un
30 objet.

La figure 2 représente schématiquement un exemple de contenu d'une fraction de la mémoire topologique.

La figure 3 représente schématiquement l'organisation en blocs de la mémoire de formes.

35 La figure 4 représente un objet de grande dimension pouvant sortir de l'image par son bord gauche.

La figure 5 est un schéma synoptique d'une configuration de système permettant de mettre en oeuvre le procédé.

On appelle "forme" l'ensemble des points ou "pixels" constituant l'image d'un objet lorsque celui-ci est analysé suivant la trame télévision. Une telle forme est faite d'une suite de "segments". Un segment est lui-même une suite de points ou de groupes de points qui seront affichés à la suite les uns des autres sur une ligne télévision. Une telle suite de points ou pixels pourrait être décrite en tant que telle en admettant comme cas général que chaque pixel est décrit individuellement. Il est plus avantageux, dans le cas d'une suite de pixels identiques en couleur, qu'un code permette de décrire toute la suite en une seule fois. Les informations décrivant les segments d'une forme sont enregistrées à des adresses consécutives dans une mémoire appelée "mémoire de formes", y constituant ainsi des chaînes de données.

Le fonctionnement du système est basé sur l'utilisation d'une mémoire dite "topologique" associée à ladite mémoire de formes. La mémoire topologique constitue en quelque sorte une carte géographique de l'image, cette dernière étant symboliquement divisée en petits carreaux dont chacun est représenté par une position dans la mémoire topologique. On enregistre dans cette mémoire topologique, pour chaque ligne de l'image au cours de laquelle on rencontre un objet, un groupe de données indiquant le début du segment de l'objet au niveau de cette ligne et comportant un numéro d'identification de cet objet, ce groupe de données ayant une position dans la mémoire qui correspond sensiblement à la position du segment dans l'image. On lit ensuite la mémoire topologique en synchronisme avec la description de l'image et, lorsqu'on y trouve l'indication du début d'un segment et le numéro de l'objet dont il fait partie, on se reporte à la description du segment qui doit être représenté, description qui se trouve dans la mémoire de formes où on lit, en synchronisme avec la description de l'image, la chaîne de données décrivant le segment.

La lecture de la mémoire topologique est, en pratique, décalée très légèrement en avance par rapport à l'exploration de l'image pour tenir compte des temps d'accès mémoire précédant l'obtention des informations nécessaires pour définir chaque point d'image.

Le champ de l'image comporte par exemple 256 lignes de 256 points ou pixels chacune, dont 220 sont affichés.

En principe, et pour faciliter leur manipulation, tous les segments d'une forme sont alignés "à gauche", c'est-à-dire que leurs débuts ont tous la même abscisse sur l'écran. Par contre, ils peuvent être de longueur quelconque.

La figure 1, qui représente un arbre, montre comment la description d'une forme est organisée. Cette silhouette mesure 24 pixels de large et 30 lignes de haut. Soit à placer cet arbre dans la position indiquée sur la figure, sa crête étant située à la ligne 32 et son pied à la ligne 61, son bord le plus à gauche à la colonne 156. Le premier segment rencontré est celui de la ligne 32 : il commence par 9 pixels "transparents" qui sont suivis de 6 pixels colorés, par exemple en vert.

Au bout de ces 6 pixels verts, le segment est terminé. Un segment situé par exemple à la ligne 38 sera constitué de 24 pixels verts. Tous les segments commencent ainsi à la colonne 156. La couleur "transparent" signifie que la couleur d'un autre élément d'objet placé au même endroit a priorité sur elle, et ceci quelque soient d'autres priorités établies par ailleurs. Les segments correspondant aux lignes 38 à 43 sont identiques entre eux : une solution évidente consiste à répéter 6 fois le même segment en mémoire de formes. Il est plus avantageux qu'il n'y figure qu'une fois. Ceci est possible si on prévoit dans la mémoire topologique la possibilité d'indiquer une répétition du même segment. Il en est de même pour les lignes 50 à 58 et pour les deux lignes 60, 61. Ces segments peuvent être décrits une seule fois chacun dans la mémoire de formes, leur répétition étant signalée dans la mémoire topologique.

On enregistre donc dans chaque groupe de données de la mémoire topologique un bit dit "de répétition" indiquant, le cas échéant, par exemple lorsqu'il est à "1", que lorsqu'on retrouvera ultérieurement le même numéro d'objet, c'est le même segment qui devra alors être à nouveau représenté.

Ce bit peut être utilisé, comme ci-dessus, pour éviter la répétition de la description d'un segment dans la mémoire de formes, mais aussi en répétant par exemple deux fois chaque segment d'un objet donné pour dilater ce dernier dans le sens vertical. Il peut aussi être utilisé lorsque plusieurs objets identiques sont situés à une même coordonnée Y. Dans ce cas, chaque segment d'un objet qui n'est pas le premier rencontré est décrit par répétition du segment de l'objet précédent, qui porte le même numéro. Grâce à cet artifice, plusieurs objets peuvent être désignés par le même numéro, ce qui augmente encore le nombre d'objets pouvant être représentés. Inversement, lorsque le même objet doit être représenté, plusieurs fois à des hauteurs différentes dans l'image, il porte un numéro différent à chaque fois, mais ces numéros renvoient toujours à la même description dans la mémoire de formes.

La figure 2 montre les portions de la mémoire topologique dans lesquelles des groupes de données indiquent les débuts de segments représentant l'arbre de la figure 1. Ces groupes de données sont enregistrés en des positions prédéterminées selon une grille régulière dans la mémoire topologique, et ils comportent, outre le numéro d'objet, une série de bits indiquant la valeur du décalage entre la position exacte du segment et la position du groupe de données.

Chaque indication de début de segment occupe un octet de la mémoire, laquelle est organisée en octets. Le groupe de données n'est pas placé en position 156 (correspondant au début du segment dans l'image) car 156 n'est pas divisible par 8. Il est donc placé dans l'octet qui contient la position 156 et commence donc en 152. Les qua-

tre premiers bits "1111" indiquent qu'il s'agit de l'objet
numéro 15. Il ne peut y avoir ici que quinze objets (et
non 16) car le code "0000" signifie "pas d'objet" et non
"objet numéro zéro". Si l'on désire plus de quinze objets,
5 il est facile d'imaginer d'autres codes à plus de quatre
bits. Toutefois, en pratique, on utilise plusieurs mémoi-
res topologiques correspondant chacune à un plan d'image
et qui exploitent ensemble une mémoire de formes commune,
et on établit des priorités entre les plans d'image, c'est-
10 à-dire entre les numéros d'objets à représenter issus de
la lecture simultanée des différentes mémoires topologi-
ques. Ainsi, une image est composée de plusieurs plans,
chacun étant décrit dans une mémoire topologique distincte,
cependant que toutes les descriptions de segments sont ras-
15 semblées dans une mémoire de formes unique, commune à tous
les plans. Ainsi, une grande souplesse de description est
obtenue, en particulier grâce au principe des priorités,
et avec par exemple quatre plans, il est possible de placer
jusqu'à $4 \times 15 = 60$ objets différents.

20 Sur la figure 2, les trois bits suivants "100" indi-
quent que l'objet est décalé de quatre positions par rap-
port à la colonne 152 : il se trouve en position 156. Le
dernier bit 0 indique que ce segment ne sera pas répété.
Aucune autre indication ne figure dans la ligne concernant
25 l'objet 15.

Le numéro d'objet et sa position "156" restent
inchangés pour tout l'objet : le même octet est donc répété
en même position pour les trente lignes de l'objet, à une
différence près : pour les lignes 38 à 42, 50 à 57 et 60,
30 le dernier bit est à 1, indiquant la répétition d'un seg-
ment identique à chaque ligne suivante, soit respectivement
les lignes 39 à 43, 51 à 58 et 61.

A la colonne 180 qui correspond à la fin de l'objet,
rien ne figure dans la mémoire topologique : c'est dans la
35 mémoire de formes que l'on trouvera la mention "fin de seg-
ment" qui, en outre, n'apparaîtra pas forcément à l'abscis-
se 180 mais à une abscisse variable selon la forme du bord
droit de l'objet.

Les octets "10101100" dans les lignes 59 et 60 indiquent le début d'un nouvel objet, représenté sur la figure 1, portant le numéro 10 et débutant à la ligne 59 et en colonne 182.

5 Une autre façon de procéder, selon laquelle on utilise encore un octet par début de segment, consiste à renoncer au bit de répétition et à utiliser quatre bits pour indiquer le décalage. Alors, un octet peut indiquer seize valeurs de décalage en plus du numéro d'objet de 1 à
10 15, c'est-à-dire qu'il permet de placer un objet sur seize positions tout en n'occupant lui-même que huit positions dans la mémoire. De ce fait, une mémoire topologique de 32 K bits suffit à décrire une image de 256 x 256 points.

Le choix entre ces façons de procéder dépend du contenu de l'image : pour la description d'un fond contenant
15 des segments longs et très répétitifs, il est avantageux d'employer un bit de répétition alors que, pour la description d'objets de petite dimension, on préfère indiquer le décalage par 4 bits, sans bit de répétition. Il est également
20 possible d'attribuer un poids égal à "2" au chiffre de décalage, la description par 3 bits indiquant alors un décalage de zéro à seize défini deux unités par deux unités : on perd en précision de positionnement, mais on gagne la moitié de la mémoire topologique. Le procédé préféré
25 consiste, puisque plusieurs mémoires topologiques sont employées correspondant chacune à un plan d'image, à réserver pour les fonds : une mémoire topologique avec bit de répétition et poids double pour la valeur de décalage, et pour les petits objets : des mémoires topologiques avec
30 quatre bits de valeur de décalage et pas de bit de répétition. Bien entendu, toute autre répartition de l'affectation des bits est possible et, également, toute autre dimension totale des groupes de données, en restant dans le cadre de l'invention.

35 On utilise un microprocesseur pour gérer l'ensemble du système et faire évoluer la position des objets en modifiant le contenu des mémoires et du registre d'adresses.

Cette modification est effectuée pendant les retours de trame de l'image.

Pour soulager ce microprocesseur lorsqu'on désire modifier le contenu de la mémoire topologique, un automate
5 arrange ledit numéro et ladite valeur de décalage selon un format prédéterminé et charge ce groupe de données en parallèle dans la mémoire topologique.

Le déplacement d'une forme n'affecte qu'une mémoire topologique. Il comprend deux opérations :

- 10 - l'effacement
- l'inscription.

Pour l'effacement, le microprocesseur doit fournir à l'automate les informations suivantes :

- "Effacement" 1 bit
- 15 - Numéro de plan 1 bit
- Coordonnée X_0 8 bits
- Coordonnée Y_0 8 bits
- Hauteur de l'objet 8 bits

L'automate exécute alors l'effacement de manière
20 automatique. La fin de l'opération est marquée par l'envoi d'une interruption au microprocesseur.

Pour l'écriture, on doit fournir :

- "Ecriture" 1 bit
- Numéro de l'objet 4 bits
- 25 - Bit de répétition 1 bit
- Coordonnées X et Y 2 octets

Si l'on prend le cas d'un objet de hauteur égale à n lignes, il faudra n cycles d'effacement et n cycles d'inscription, soit un total de $2 \times n$ cycles.

30 On peut aussi fournir en une seule fois à l'automate les données suivantes, par exemple dans le cas où il n'y a pas de bit de répétition :

- X_0 , Y_0 , X , Y
- numéro d'objet, numéro de plan
- 35 - hauteur de l'objet.

Le temps nécessaire pour exécuter cette commande, qui dépend du nombre de lignes, est de l'ordre de $0,8 \mu s$

par ligne. Le déplacement d'un arrière-plan complet simple représentant par exemple un terrain de football demande environ 250 μ S. Le déplacement d'un objet moyen nécessite environ 25 μ S (pour une trentaine de lignes). Le
 5 temps de retour de trame d'environ 2 mS permet donc de déplacer un grand nombre d'objets.

Dans la mémoire de formes, les chaînes de données sont constituées de blocs ayant tous la même longueur et elles comportent au moins un bloc d'adresses contenant
 10 l'adresse de la chaîne de données décrivant le segment suivant, et au moins un bloc dit "bloc de pixels" contenant :

- des indications sur sa propre position par rapport aux autres blocs
- 15 - et la description d'une succession de points définie par leur couleur et leur nombre.

La figure 3A donne un exemple de structure de la description d'un bloc de pixels dans la mémoire de formes.

Un tel groupe est décrit par deux octets : l'octet
 20 de couleur 1 qui permet la description de 256 couleurs, et l'octet de service qui donne les informations suivantes :

- le nombre de pixels dans le groupe (référence 2), (5 bits, soit 32 pixels maximum),
- 2 bits de service (référence 3) qui décrivent la
 25 position du groupe dans le segment parmi les différentes possibilités suivantes exclusives les unes des autres : séquence normale (code 00), fin de segment (code 01), fin de l'objet (code 10), "saut ensuite" (code 11) qui signifie que les deux octets suivants doivent être sautés lorsqu'on voudra lire le groupe de pixels suivant,
 30 - un bit de transparence 4 indiquant par un "1" si le groupe de pixels est transparent.

La figure 3A représente les 2 octets définissant les neuf points transparents de la première ligne de l'ar-
 35 bre de la figure 1.

La figure 3B représente les deux octets définissant les six pixels suivants, le code "10110000" représentant

par exemple la couleur verte désirée.

La figure 3C représente les deux octets décrivant, à eux seuls, toute la ligne 38.

Il est bien évident que toute autre organisation de ces chaînes de données est possible, pourvu que l'on y

En outre, les deux premiers octets de chaque description de segment ne décrivent pas un groupe de pixels, mais donnent l'adresse de départ du segment suivant. Ceci est utile dans le cas où un segment en cours serait interrompu avant sa fin par un retour de ligne.

Un exemple de structure de la description d'un segment entier est représentée par la figure 3D. Elle commence par l'adresse 14 du segment suivant, est suivie d'un certain nombre de couples d'octets décrivant des groupes de points (15, 17, 170, 19) dans lesquels sont intercalées d'autres adresses (16, 18). Celles-ci sont utilisées dans un cas particulier qui sera expliqué plus loin.

Les bits de service 3 des couples d'octets situés avant chaque bloc d'adresses intercalé indiquent "saut ensuite" afin que, dans une séquence normale, ce bloc d'adresses soit sauté. Le dernier couple comporte des bits de service 3 indiquant "fin de segment". Sauf dans le cas où une adresse est intercalée, les couples d'octets décrivant les groupes de pixels d'un segment sont mémorisés dans la mémoire de formes à des adresses consécutives. De même, les chaînes de données décrivant les segments sont enregistrées bout à bout dans l'ordre de description de l'objet par lignes.

La mémoire de formes d'objet est par exemple une mémoire statique avec un temps d'accès d'environ 100 nS. Elle est organisée par blocs de 8 bits. Elle peut être adressée en entier par le microprocesseur durant les retours de trame. Pendant le reste du temps, elle est utilisée à 100 % par la circuiterie d'affichage de l'image.

Pour permettre de trouver dans la mémoire de formes les segments rencontrés lors de la lecture de la mémoire

- topologique, identifiés par un numéro d'objet, on enregistre dans un registre d'adresses associé à chaque mémoire topologique, à des positions correspondant respectivement à chaque numéro d'objet, des adresses de début
- 5 et de départ de la description desdits objets dans la mémoire de formes. Ce registre d'adresses est constitué de deux piles, l'une, P_0 , contenant pour chaque numéro d'objet l'adresse, dite "de début", fixée au moins pour
- 10 objet dans la mémoire de formes, et l'autre, P , contenant pour chaque numéro d'objet une adresse dite "de départ" correspondant à la description d'un segment de l'objet, adresse de départ qui est modifiée chaque fois qu'un nouveau segment différent du précédent doit être représenté.
- 15 Avec plusieurs mémoires topologiques, il convient que chacune ait son propre registre d'adresses. Ces registres renvoient tous à la même mémoire de formes.
- Soit, par exemple, à afficher un objet décrit à partir d'une adresse de départ AD dans la mémoire de formes. Soit PL le plan dans lequel on désire placer cet
- 20 objet, et soit N son numéro dans ledit plan. On écrit l'adresse AD à la position N dans la pile d'adresses de début P_0 et dans la pile d'adresses de départ P qui sont associées à la mémoire topologique décrivant le plan PL.
- 25 On introduit, en outre, l'indication de l'objet dans cette mémoire topologique, de la manière décrite plus haut. (s'il s'agit d'un objet nouveau et non du déplacement d'un objet préexistant, on indique conventionnellement comme adresse précédente : 256, 256).
- 30 La pile d'adresses de début P_0 contient donc à tout moment les 15 adresses initiales de 15 chaînes de données décrivant chacune l'ensemble des segments d'un objet. La pile d'adresses de départ P contient l'adresse du segment en cours d'affichage ou du segment suivant.
- 35 Lors des mises à jour, pendant les retours de trame, la pile P_0 est recopiée dans la pile P . Chaque fois qu'on rencontre un nouveau numéro d'objet dans une ligne, on lit

l'adresse du segment à décrire dans la pile P du registre d'adresses. Si le bit de répétition est à "1", on saute les deux octets d'adresse par lesquels commence la chaîne de données. Si ce bit n'est pas à "1", on lit l'adresse
5 contenue dans ces deux octets et on l'enregistre dans le registre P à la place de l'adresse de départ qu'on vient d'utiliser. On lit ensuite le bloc de pixels qui se trouve à la suite du bloc d'adresse. Le nombre de pixels de ce groupe est chargé dans un décompteur CP dont l'horloge
10 fournit une impulsion pour chaque point d'image balayé par le système d'affichage sur écran vidéo. Dans un autre compteur CA, est recopiée l'adresse trouvée dans la pile P. Ce compteur sert à conserver à tout moment l'adresse du groupe de pixels en cours de description. Les piles P₀
15 et P contiennent chacune une adresse par numéro d'objet, soit 15 adresses chacune alors que, par contre, les compteurs CP et CA n'existent qu'à un exemplaire et sont utilisés par tous les objets tour à tour. Lorsque le nombre de points d'image balayés correspond au nombre de pixels
20 du groupe, le décompteur CP arrive à zéro. Le zéro en sortie du compteur CP déclenche l'incréméntation d'une unité de l'adresse contenue dans le compteur d'adresse CA. Alors, quatre cas peuvent se présenter selon le contenu des deux bits de service :

25 1) Séquence normale : le compteur d'adresse CA, qui vient d'être incrémenté, indique donc ainsi l'adresse du groupe de pixels suivant.

2) Fin de segment : puisque les segments sont placés bout à bout dans la mémoire de formes, l'adresse du
30 segment suivant est, comme dans le cas précédent, indiquée par le compteur CA. Mais cette adresse est ici inutile puisqu'elle a déjà été lue en tête de segment et placée dans la pile P.

3) Fin d'objet : on recopie de P₀ dans P l'adresse
35 de début de l'objet.

4) "Saut ensuite" : on incrémente d'une unité supplémentaire le compteur d'adresses CA.

Il est évident qu'on suppose ici la mémoire de formes organisée par couples d'octets de façon que, lorsque le compteur CA est "incrémenté d'une unité", ceci procure l'adresse du couple d'octets suivant. La programmation
5 des opérations décrites ci-dessus : saut d'adresse, copie, opérations sélectionnées selon des bits de service... est un travail bien connu de l'homme du métier qu'il est donc inutile de décrire en détail.

Un cas particulier important est celui où un objet
10 mobile se déplace vers un bord de l'image et franchit ce bord pour sortir de l'image. Si l'objet sort par la droite, il "déborde" de l'image et la description d'au moins certains segments est interrompue par des retours de ligne. L'adresse contenue alors dans le compteur CA n'est plus
15 valide. Mais l'adresse du début du segment suivant est dans la pile P. Tout se passe alors comme dans le cas d'une fin de segment. Si l'objet sort par le bas de l'image, cela ne pose aucun problème puisque les mémoires sont remises à jour pendant les retours de trame. Si l'objet
20 sort par le haut de l'image, son ou ses premier(s) segment(s) commence(nt) par disparaître, puis un ou quelques-uns de plus à chaque image. Lors des mises à jour, le microprocesseur place dans la pile d'adresses de début P_0 l'adresse du $n^{i\text{ème}}$ segment désiré, au lieu de l'adresse de
25 début d'objet. Il trouve cette adresse, pendant les retours de trame, en explorant la mémoire de formes.

Le cas le plus difficile est celui où l'objet disparaît progressivement par la gauche de l'image. Dès que sa limite gauche n'est plus dans l'image, il n'est, en
30 principe, plus possible d'indiquer la présence d'un objet au moyen de la mémoire topologique puisque c'est précisément cette limite gauche qui y est mentionnée et qu'elle a disparu. Pour résoudre ce problème, on prévoit dans la mémoire topologique des positions correspondant à des por-
35 tions de ligne d'image fictives non représentables et situées à gauche et à l'extérieur de l'image. Dans le cas d'objets de grandes dimensions, on est amené à prévoir un

volume de mémoire très important pour représenter les parties fictives, ce qui n'est pas souhaitable. Pour éviter cela, on subdivise la description de chaque segment dans la mémoire de formes en sous-segments dont la représentation dans l'image a une longueur déterminée constante, 5
sauf pour le dernier sous-segment d'un segment et, en tête de chaque chaîne de données décrivant un sous-segment, on place l'adresse du sous-segment correspondant du segment suivant. Par exemple, on prévoit une bande de 32 pixels 10
fictifs à gauche de chaque ligne dans la mémoire topologique, et on divise en sous-segments de 32 pixels la représentation des objets mobiles dans la mémoire de formes. La valeur de 32 pixels est avantageuse car elle correspond au nombre maximum de pixels qu'il est possible de décrire 15
par un seul couple d'octets dans la mémoire de formes.

La description de l'objet commence alors par la première colonne dont une partie est encore visible.

La figure 4 montre l'exemple d'un objet 13 divisé en colonnes 10, 11, 12 dont chacune a une largeur de 32 20
pixels, sauf la dernière. Le bord gauche de l'image est indiqué en 9. La colonne 10 n'est pas utilisée, et l'adresse indiquée dans la pile P_0 est celle de la chaîne de données décrivant le segment qui commence au point AD1. Cette chaîne est représentée sur la figure 3D. En 14, se 25
trouve l'adresse du segment suivant ; en 15, est décrit un bloc de pixels, situés à gauche du point AD1 sur la figure 4, dont l'octet de service est représenté en détail, comportant les bits "11111" : 32 points, "11" : saut ensuite, "1" : transparence ; en 16, se trouve l'adresse du 30
sous-segment correspondant du segment suivant, c'est-à-dire du sous-segment situé en colonne 11 dans la ligne suivante ; en 17, sont décrits les 26 premiers pixels transparents après le point AD1 de la première ligne de la colonne 11, avec un octet de service comportant les bits 35
"11010" : 26 points, "00" : séquence normale, "1" : transparence. En 170, sont décrits les 6 points restants du groupe de 32 points de la première ligne de la

colonne 11, avec un octet de service comportant les bits
"00110" : 6 points, "11" : saut ensuite, "0" : non trans-
parence. En 18, est indiquée l'adresse du sous-segment
situé en colonne 12 dans le segment suivant et, en 19,
5 est décrit le dernier groupe de pixels de la première
ligne de l'objet, avec un octet de service comportant :
"01100" : 12 points, "01" : fin de segment, "0" : non
transparence. La référence AD1 entre les couples d'octets
15 et 16 indique l'adresse qui est enregistrée dans la
10 pile P_0 lors de la mise à jour d'image. Tout se passe
comme si l'objet était un nouvel objet amputé de sa
colonne gauche 10, sa description étant "prise au vol" à
partir de l'adresse AD1. Bien entendu, à chaque amputa-
tion d'une nouvelle colonne, la coordonnée X de l'objet
15 dans la mémoire topologique doit être augmentée de 32,
en plus de la variation correspondant au déplacement sou-
haité, ceci pour compenser la disparition de la colonne.

Le système représenté par la figure 5 est donc
organisé autour d'une mémoire topologique 20, constituée
20 ici de deux blocs de mémoire à accès aléatoire, dite "RAM
dynamique", d'une capacité unitaire de 16 K x 4 bits,
représentant quatre plans d'image en utilisant 32 K bits
par plan, et de la mémoire de formes 31 constituée par
exemple de 8 K octets de mémoire RAM statique rapide.
25 Cette mémoire de formes contient tous les objets utilis-
ables au cours d'une session d'utilisation, et pas seule-
ment ceux qui sont affichés à un instant donné.

Un module d'interface 21, qui contient le susdit
automate, est relié par les liaisons 23, 34 respective-
30 ment à la mémoire topologique 20 et à un module 26 géné-
rateur d'images. Un multiplexeur 25 connecte à la demande
les liaisons 22 entre le microprocesseur 33 et les modules
Un élément de service 24 procure aux différents modules
divers signaux de service : horloge, synchronisation
35 ligne et trame. Le module d'interface 21 assure le charge-
ment de la mémoire 20 pendant les retours de trame et sa
lecture synchrone pendant l'exploration de l'image. Le

générateur d'image 26, qui contient les piles P, P₀ et
les compteurs CA, CP, assure via la liaison 35 la lecture
de la mémoire de formes selon les indications de numéro
d'objet fournies par le module d'interface 21, et fournit
5 en temps réel sur une connexion 30 l'indication de la cou-
leur à reproduire.

- REVENDICATIONS -

1. Procédé pour engendrer, dans une image décrite par une succession de lignes en liaison avec des signaux de synchronisation de ligne et de trame, des objets représentés par une série de segments les uns au-dessus
5 des autres, chacun correspondant à une ligne de l'image, caractérisé en ce que :

- on enregistre dans au moins une mémoire dite "topologique", pour chaque ligne de l'image au cours de laquelle on rencontre un objet, un groupe de données indiquant le
10 début du segment de l'objet au niveau de cette ligne et comportant un numéro d'identification de cet objet, ce groupe de données ayant une position dans la mémoire qui correspond sensiblement à la position du segment dans l'image,

15 - on enregistre dans une mémoire dite "de formes" des chaînes de données constituant la description desdits segments constituant les objets,

- on enregistre dans un registre d'adresses associé à la mémoire topologique, à des positions correspondant respectivement à chaque numéro d'objet, des adresses de début
20 et de départ de la description desdits objets dans la mémoire de formes,

- on lit la mémoire topologique en synchronisme avec la description de l'image,

25 - lorsqu'on y trouve l'indication du début d'un segment et le numéro de l'objet dont il fait partie, on se reporte à la position correspondant audit numéro dans le registre d'adresses pour y trouver l'adresse à laquelle se trouve, dans la mémoire de formes, la description du segment qui
30 doit être représenté, et

- enfin, on lit la chaîne de données décrivant ce segment, en synchronisme avec la description de l'image.

2. Procédé selon la revendication 1, caractérisé en ce que le registre d'adresses est constitué de deux

5 piles, l'une contenant pour chaque numéro d'objet l'adresse, fixée au moins pour la durée d'une trame, du début de la description dudit objet dans la mémoire de formes, et l'autre contenant une adresse de départ correspondant à la description d'un segment de l'objet, adresse de départ qui est modifiée chaque fois qu'un nouveau segment différent du précédent doit être représenté.

10 3. Procédé selon l'une des revendications 1 ou 2, dans lequel on effectue une modification du contenu d'au moins une mémoire afin de faire évoluer la position des objets dans l'image, caractérisé en ce que cette modification est effectuée pendant les temps de retour de trame de l'image.

15 4. Procédé selon l'une quelconque des revendications 1 à 3, caractérisé en ce qu'on enregistre lesdits groupes de données indiquant le début d'un segment en des positions prédéterminées selon une grille régulière dans la mémoire topologique, et en ce que ces groupes de données comportent, outre le numéro d'objet, une série de
20 bits indiquant la valeur du décalage entre la position exacte du segment et la position du groupe de données.

25 5. Procédé selon la revendication 4, caractérisé en ce que le numéro d'identification de l'objet comporte quatre bits et la valeur du décalage est indiquée également par quatre bits.

30 6. Procédé selon la revendication 4, dans lequel on utilise un microprocesseur pour gérer l'ensemble du système et faire évoluer la position des objets en modifiant au moins le contenu de la mémoire topologique, caractérisé en ce qu'on utilise pour modifier ce contenu un automate qui arrange ledit numéro et ladite valeur de décalage selon un format prédéterminé, et charge ce groupe de données en parallèle dans la mémoire topologique.

35 7. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce qu'on enregistre en outre dans chaque groupe de données de la mémoire topologique un bit dit "de répétition" indiquant, le cas échéant,

que lorsqu'on retrouvera ultérieurement le même numéro d'objet, c'est le même segment qui devra alors être à nouveau représenté, et en ce que, après avoir lu l'adresse de départ dans le registre d'adresses, on y laisse cette même
5 adresse dans le cas d'une répétition, alors qu'on la remplace par celle du segment suivant dans le cas contraire.

8. Procédé selon la revendication 7, caractérisé en ce que le numéro d'identification de l'objet comporte quatre bits, la valeur du décalage étant indiquée par
10 trois bits.

9. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que, en tête de chaque description de segment dans la mémoire de formes, on enregistre l'adresse de la chaîne de données décrivant le
15 segment suivant.

10. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce qu'on subdivise la description de chaque segment dans la mémoire de formes en sous-segments dont la représentation dans l'image a une
20 longueur déterminée constante, sauf pour le dernier sous-segment d'un segment, en ce qu'en tête de chaque chaîne de données décrivant un sous-segment, on place l'adresse du sous-segment correspondant du segment suivant, et en ce qu'on prévoit dans la mémoire topologique des positions
25 correspondant à des portions de ligne d'image fictives non représentables et situées à gauche et à l'extérieur de l'image.

11. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce que lesdites chaînes
30 de données sont constituées de blocs ayant tous la même longueur, l'un au moins de ces blocs dit "bloc d'adresses" contenant l'adresse de la chaîne de données décrivant le segment suivant, et au moins un autre bloc dit "bloc de pixels" contenant :

35 - des indications sur sa propre position par rapport aux autres blocs,

- et la description d'une succession de points définie par leur couleur et leur nombre.

12. Procédé selon l'ensemble des revendications 10 et 11, caractérisé en ce que ladite longueur déterminée constante des sous-segments comporte un nombre de points égal au nombre maximum de points que les blocs de pixels
5 sont susceptibles de définir.

13. Procédé selon l'une quelconque des revendications précédentes, caractérisé en ce qu'on utilise plusieurs mémoires topologiques correspondant chacune à un plan d'image et qui exploitent ensemble une mémoire de
10 formes commune, et en ce qu'on établit des priorités entre les différentes mémoires topologiques, c'est-à-dire entre les numéros d'objets à représenter issus de leurs lectures simultanées.

1/2

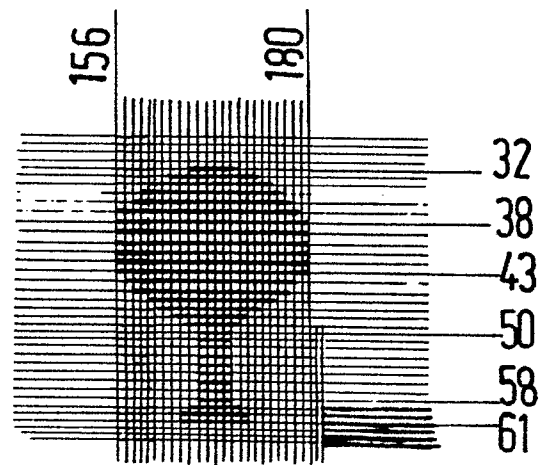


FIG. 1

	152	156	160	164	176	180	184
32	0,0,0	1,1,1,1	1,0,0,0	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0
33	0,0,0	1,1,1,1	1,0,0,0	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0
34	0,0,0	1,1,1,1	1,0,0,0	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0
50	0,0,0	1,1,1,1	1,0,0,1	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0
58	0,0,0	1,1,1,1	1,0,0,0	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0	0,0,0,0,0
59	0,0,0	1,1,1,1	1,0,0,0	0,0,0,0,0	0,1,0,1,0	1,1,0,0	0,0,0,0,0
60	0,0,0	1,1,1,1	1,0,0,1	0,0,0,0,0	0,1,0,1,0	1,1,0,0	0,0,0,0,0
61	0,0,0	1,1,1,1	1,0,0,0	0,0,0,0,0	0,1,0,1,0	1,1,0,0	0,0,0,0,0

FIG. 2

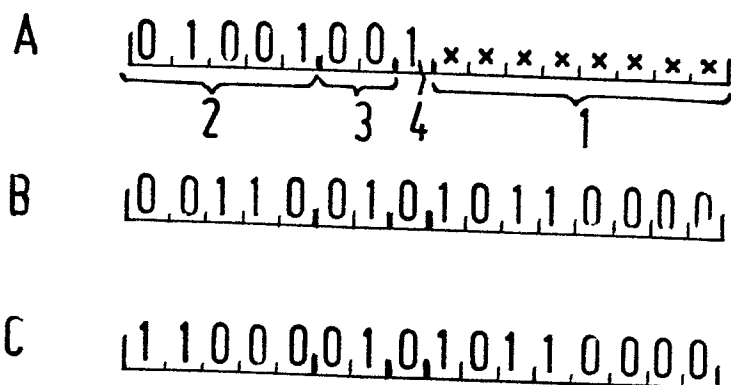


FIG.3

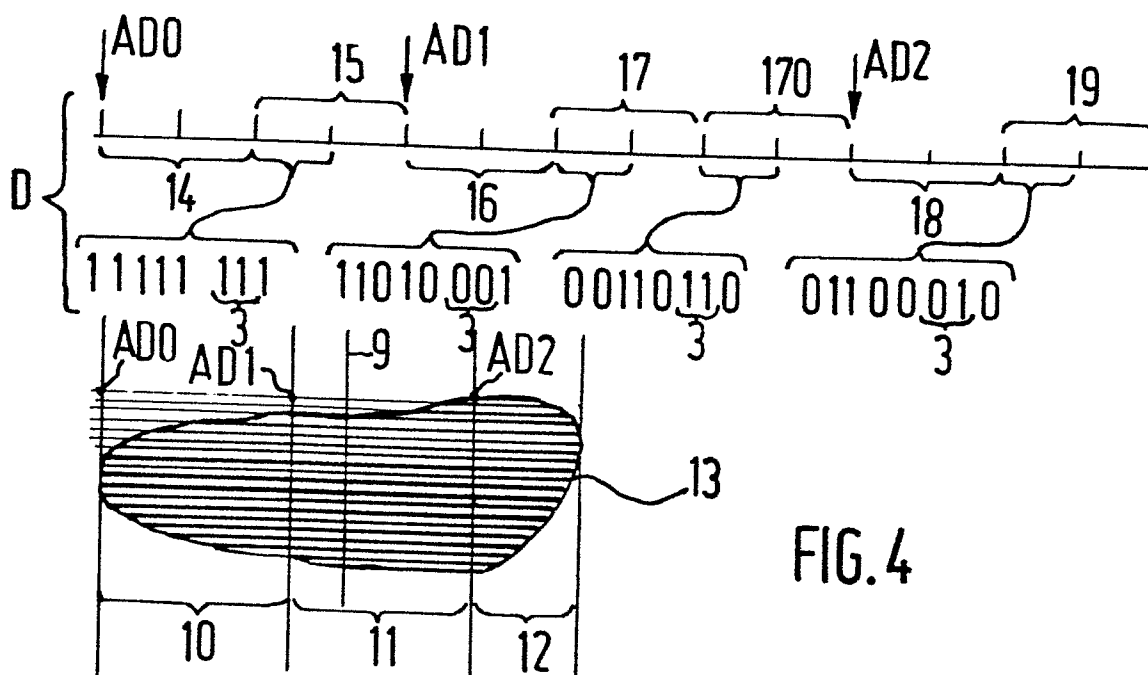


FIG.4

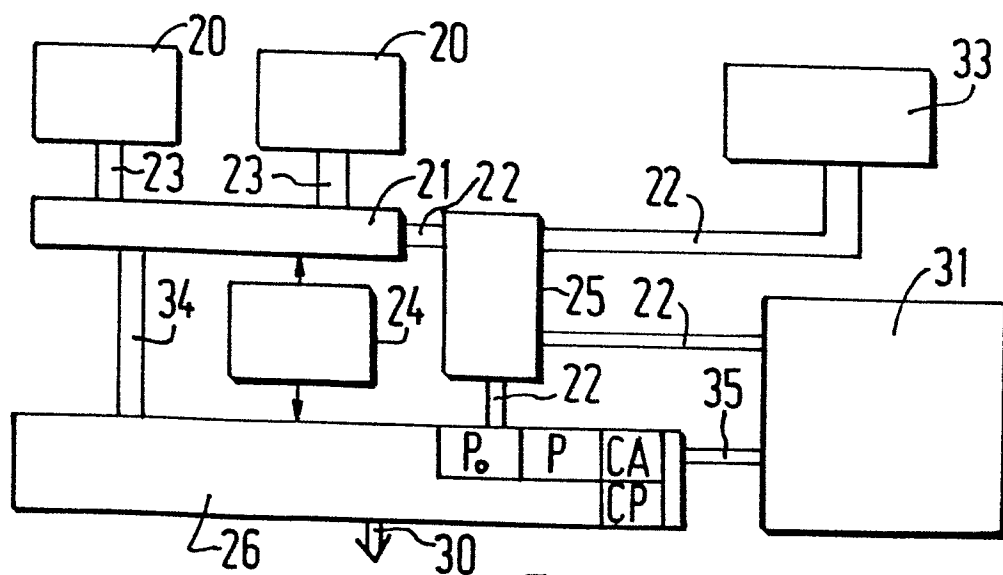


FIG.5