

A1

**DEMANDE
DE BREVET D'INVENTION**

(21)

N° 81 13675

(54) Procédé de diffusion de données sur canal de télévision.

(51) Classification internationale (Int. Cl. ³). H 04 L 5/22; H 03 K 13/32; H 04 N 7/00.

(22) Date de dépôt..... 10 juillet 1981.

(33) (32) (31) Priorité revendiquée :

(41) Date de la mise à la disposition du
public de la demande..... B.O.P.I. — « Listes » n° 2 du 14-1-1983.

(71) Déposant : SOCIETE POUR L'ETUDE ET LA FABRICATION DES CIRCUITS INTEGRES SPECIAUX (EFCIS), société anonyme. — FR.

(72) Invention de : Michel Henry.

(73) Titulaire : *Idem* (71)

(74) Mandataire : Thomson-Csf, Philippe Guilguet, Scpi,
173, bd Haussmann, 75360 Paris cedex 08.

PROCÉDE DE DIFFUSION DE DONNEES SUR CANAL DE TELEVISION.

La présente invention concerne la diffusion de données numériques à sens unique par canal de télévision selon des normes compatibles avec la diffusion actuelle des émissions publiques de télévision.

5 On essaye actuellement de créer des réseaux de transport d'informations par voie hertzienne à sens unique, de manière que des usagers possédant un récepteur de télévision puissent recevoir ces informations. On ne s'intéresse ici qu'à la transmission des informations, même si certaines applications prévoient une possi-
10 bilité d'interaction entre la source d'information et l'utilisateur : en effet, dans ce cas, le retour de l'utilisateur vers la source s'effectuera en principe autrement que par canal de télévision.

En pratique, il est prévu que l'on transmette les données numériques, sous forme d'impulsions binaires modulant en fréquence
15 la porteuse du canal de télévision, soit sur un canal réservé à cet effet, soit pendant les quelques lignes correspondant au retour de trame dans un canal réservé à une émission normale de télévision. Les données numériques transmises sont décodées au niveau d'un récepteur spécialisé et peuvent être affichées sur
20 l'écran du récepteur de télévision, par exemple sous forme d'un texte qui s'inscrit sur l'écran.

Surtout dans le cas où on transmet des informations pendant quelques lignes du retour de trame, les données numériques doivent être transmises par paquets correspondant chacun à une li-
25 gne vidéo. Compte tenu de la fréquence porteuse des canaux utilisés, de la modulation possible de cette fréquence, et de la durée imposée des lignes vidéo, on conçoit que l'on peut transmettre seulement un petit nombre d'informations pendant la durée de chaque ligne. En pratique, les paquets ne peuvent guère comporter
30 plus d'une quarantaine d'octets.

De plus, il faut réserver dans chaque paquet un certain nombre d'octets qui sont destinés à la synchronisation des don-

nées, à l'identification des paquets, à la détection des pertes de paquets, etc...

On a donc déjà proposé des procédés de transmission dans lesquels les paquets sont divisés en un certain nombre d'octets d'en-tête et un certain nombre d'octets de données.

Plusieurs systèmes ont été proposés, et notamment un système dans lequel l'en-tête est décomposé en une première partie et une seconde partie de la manière suivante.

La première partie comprend :

10 - des bits de synchronisation destinés à la définition de la fréquence de transmission des données qui suivent ; le récepteur détecte ces bits et se synchronise sur leur fréquence ;

- un octet de départ ayant une valeur particulière que reconnaît le récepteur pour définir ensuite les limites d'octets 15 qui suivent par comptage de groupes de huit bits à partir de l'octet de départ ;

- plusieurs octets d'adresse définissant le paquet transmis par rapport aux autres paquets : ces octets peuvent concerner par exemple un numéro de voie s'il y a transmission multiplexée 20 des paquets en plusieurs voies, ou encore un numéro de magazine d'information, ou de page ou de paragraphe, etc... auxquels se réfère le paquet ;

- éventuellement, un indice de continuité qui est un octet définissant le numéro du paquet dans un groupe de paquets 25 transmis sur une même voie : si deux paquets de la même voie se suivent en présentant deux indices de continuité non consécutifs, il y aura là une indication de perte d'un ou plusieurs paquets.

La deuxième partie de l'en-tête du système le plus performant proposé actuellement comporte un octet de format qui indique 30 que combien d'octets suivent dans le paquet de données. Cet octet de format est nécessaire car on ne connaît pas a priori la longueur des paquets de données, notamment du fait que cette longueur dépend de la capacité de transmission du canal, donc de la fréquence de celui-ci qui peut varier dans de larges proportions.

35 Il faut indiquer ici que les canaux de transmission hertzienne de télévision sont relativement bruyants et qu'il faut pro-

téger efficacement les données transmises contre les erreurs. En particulier, les octets de l'en-tête doivent être tout spécialement protégés, et c'est pourquoi on prévoyait dans le système décrit ci-dessus que chaque octet était, en réalité un octet
5 redondant comportant quatre bits utiles et quatre bits de redondance, les bits de redondance étant codés par exemple selon le code normalisé dit "de Hamming".

Or, quatre bits utiles dans un octet de format ne permettent pas de définir un nombre supérieur à seize. On résolvait le
10 problème d'une manière assez lourde d'emploi, en désignant non pas un format effectif, mais un numéro de format, chaque format correspondant à un nombre d'octets particulier, et en coupant les paquets ne comportant pas un nombre d'octets correspondant à un format existant.

15 La présente invention se fonde sur plusieurs remarques. La première est que les données transmises seront le plus souvent des pages de texte où les caractères alphanumériques seront transmis tout simplement suivant le code binaire normalisé ASCII, c'est-à-dire le code actuellement utilisé pour les téléimprimeurs.
20 Dans ce code, tous les caractères alphanumériques, ainsi qu'un certain nombre de codes de commande, sont des octets comportant sept bits utiles et un bit de parité. Parmi les codes de commande, il existe un code dit "code NUL", qui est simplement un code de bourrage qui n'a aucune signification particulière, mais qui peut
25 être utilisé pour remplir des durées d'émission dans lesquelles aucune information n'est transmise. Il paraît donc possible d'ores et déjà de supprimer l'octet de format si on sait que les données numériques transmises sont des pages de télétexte en code ASCII.

Une deuxième remarque est que les systèmes connus s'ins-
30 pirent de la transmission de textes dans lesquels toutes les données sont découpées en octets, et ils ne peuvent travailler que dans ces conditions. Si le découpage est différent, les procédés utilisés ne permettent plus le décodage des données transmises. Un exemple d'inadéquation du découpage en octets est la transmission
35 musicale sous forme numérique : il est couramment admis qu'il faut entre onze et quatorze bits pour définir des niveaux de son per-

mettant une reproduction sonore avec une dynamique suffisante. D'autres exemples pourraient être trouvés, et d'ailleurs, même dans le cas de traitements de données de texte, on s'oriente actuellement de plus en plus vers une transmission par chaînes de 5 bits quelconques.

Il est donc souhaitable d'envisager une procédure de transmission permettant aussi bien la réception et la reconnaissance de données découpées en octets que la réception et la reconnaissance de données découpées autrement ou transmises tout
10 simplement en chaînes de bits.

Une troisième remarque est que, dans les systèmes proposés, on a pris grand soin de protéger les informations de l'en-tête des paquets, puisqu'on les a codées en octets redondants de Hamming, mais on n'a rien prévu pour protéger les données elles-
15 mêmes. Or, comme on l'a dit, les canaux de transmission sont bruyants et il est souhaitable de terminer chaque paquet de données par un code de détection et de correction d'erreurs.

On admet qu'un code correcteur capable de corriger une erreur dans un paquet de données nécessite environ dix bits, soit
20 en pratique deux octets si le découpage des données est fait en octets. Mais, l'utilisation de deux octets réduit encore la capacité de transmission de chaque paquet. Un des buts de l'invention est d'essayer d'optimiser la capacité de transmission et la protection contre les erreurs aussi bien dans l'en-tête des paquets
25 que dans les données transportées elles-mêmes.

Enfin, un but de l'invention est de proposer un procédé de diffusion de données qui rende le réseau de transport des données aussi "transparent" que possible vis-à-vis de la structure des données transportées. Ceci signifie que l'on doit pouvoir en-
30 voyer des données de structures différentes avec un minimum de conventions implicites entre la source de données et les récepteurs. Un exemple de convention implicite que la présente invention cherche à éliminer pour rendre le réseau plus "transparent" est la convention selon laquelle les données sont découpées en
35 octets.

Pour atteindre ces différents buts, et pour éliminer un

certain nombre d'inconvénients des systèmes de diffusion de données proposés jusqu'à maintenant, la présente invention propose un procédé de transmission de paquets de données, avec un agencement particulier des paquets les uns par rapport aux autres et un agencement particulier des en-têtes de ces paquets (et plus précisément de la seconde partie d'en-tête), cet agencement étant différent selon les structures de données transportées dans les paquets, afin que l'on puisse transmettre soit des informations agencées en octets de télétexte selon le code ASCII, soit des informations agencées autrement, et notamment en chaînes de bits.

Selon l'invention, on transmet, dans une même voie numérique, les paquets de la manière suivante :

- a) les paquets sont transmis par "wagons" ne comprenant que des paquets correspondant à des données de même type ;
- 15 b) chaque wagon comprend d'abord un ou plusieurs paquets dits "de procédure" spécifiques au wagon et indiquant notamment le type de données transmises dans le wagon, des paquets de procédure spécifiques étant prévus notamment pour les wagons de transmission de télétexte en code ASCII et pour les wagons de transmission de 20 données par chaînes de bits ;
- c) chaque seconde partie d'en-tête d'un paquet de procédure comprend d'abord un octet ou plusieurs non susceptibles d'être confondus avec un des symboles du code ASCII utilisés effectivement dans la transmission de données à sens unique ; la 25 première partie d'en-tête peut être celle qu'on a déjà mentionnée ;
- d) les paquets de données correspondant aux wagons de transmission de texte selon le code ASCII comprennent la première partie de l'en-tête suivie immédiatement des données à transmettre agencées en octets selon le code ASCII et utilisant le code de 30 bourrage "NUL" pour remplir les fins de paquets non pleins ;
- e) les paquets de données correspondant aux wagons de transmission de données agencées en chaînes de bits comprennent une seconde partie d'en-tête comportant au moins :
 - un bit indiquant si le paquet est plein ou non,
 - 35 - et si le paquet n'est pas plein, un nombre de bits fixe suffisant pour indiquer combien de bits de données sont contenus

dans le paquet.

L'invention permet donc de transmettre des données de types différents ; le paquet de procédure qui se trouve en tête de chaque wagon définit la structure des données qui suivent. S'il 5 indique que les données sont en code ASCII, aucun octet de format ne sera prévu dans les paquets de données du wagon ; en tant que correcteur d'erreurs, on utilisera à la fin du paquet un code de parité longitudinale en un seul octet. Si le paquet de procédure indique qu'il s'agit d'une transmission en chaînes de bits, les 10 paquets de données suivants seront dans la grande majorité des cas des paquets pleins, auquel cas il ne sera pas nécessaire d'indiquer un format particulier ; c'est seulement si le paquet n'est pas plein qu'on indiquera, par une instruction de format, combien de bits sont contenus dans le paquet.

15 Par ailleurs, selon une autre caractéristique de l'invention, on divise les wagons en malles correspondant à des groupes d'informations à distinguer les uns des autres, surtout pour les informations émises en chaînes de bits. Chaque malle comprend plusieurs paquets. Dans ce cas, un octet de Hamming est prévu dans la 20 seconde partie d'en-tête des paquets de données agencées en chaînes de bits, cet octet comprenant quatre bits utiles qui sont :

- un bit indiquant si le paquet est un début de malle ou non,
- un bit indiquant si le paquet est une fin de malle ou 25 non,
- un bit indiquant si le paquet est plein ou non,
- un bit indiquant le chiffre de poids supérieur du nombre binaire représentant le nombre de bits de données du paquet, ce bit étant forcé à 1 si le paquet est plein.

30 D'autres caractéristiques et avantages de l'invention apparaîtront à la lecture de la description détaillée qui suit et qui est faite en référence aux dessins annexés dans lesquels :

la figure 1 représente un modèle de paquets de données numériques à transmettre pendant la durée d'une ligne de balayage 35 de télévision ;

la figure 2 représente des paquets de données découpés en

octets ;

la figure 3 représente des paquets de données qui sont des suites de caractères du code ASCII ;

la figure 4 représente des paquets de données en chaînes 5 de bits ;

la figure 5 représente des paquets de procédure.

Les paquets de données comprennent chacun un en-tête destiné à leur identification, suivi des données numériques proprement dites (figure 1). On peut considérer que l'en-tête est divisé 10 en une première partie et une deuxième partie.

La première partie peut être celle qui a déjà été mentionnée à propos des systèmes antérieurs : elle peut comprendre deux octets de synchronisation SY1 et SY2, un octet de départ SO, un ou plusieurs octets d'adresse, SA1, SA2, SA3 (cette adresse 15 peut inclure un numéro de voie si les paquets sont transmis sur plusieurs voies avec multiplexage de ces voies), et enfin un indice de continuité IC qui termine la première partie de l'en-tête des paquets.

La deuxième partie de l'en-tête comprend une séquence caractéristique SC à laquelle s'intéresse la présente invention. Dans les systèmes déjà proposés, cette deuxième partie n'existait pas ou comprenait seulement un octet de format indiquant le nombre d'octets de données qui suivent dans le paquet.

La deuxième partie est suivie par les données proprement 25 dites, et on peut terminer ces données par un code correcteur CC comprenant un nombre de bits plus ou moins important selon la puissance de correction désirée.

Selon l'invention, les paquets peuvent transporter des données agencées en octets ou agencées autrement, et la séquence 30 caractéristique SC dépend de cet agencement. De même, le code correcteur CC peut dépendre aussi de cet agencement.

Toutefois, ce n'est pas la séquence caractéristique SC d'un paquet de données qui définit la structure des données qui arrivent ensuite. Au contraire, cette structure est définie pour 35 tout un ensemble de paquets qui arrivent successivement sur une même voie, ensemble que l'on appellera un "wagon" de données.

Ainsi, sur une voie, les données arrivent par wagons comprenant chacun une série de paquets dans lesquels les données transportées ont la même structure. On peut diviser les wagons en malles comportant plusieurs paquets chacune, afin de séparer des groupes d'informations selon une articulation logique pour l'utilisateur : par exemple, on pourrait imaginer qu'un wagon correspond à un magazine et que les malles correspondent aux pages de ce magazine. Cette notion de malle a surtout un aspect pratique. De plus, on peut faire coïncider la notion de malle avec la notion d'adresse demandée par l'utilisateur. On peut aussi s'en servir pour faire en sorte que si une perte de paquets se produit, la malle toute entière soit recherchée à nouveau (dans un système où les informations sont émises cycliquement sur le canal de télévision).

Chaque wagon de données comprend donc plusieurs malles et la première malle qui se présente dans un wagon est une malle dite de procédure, qui a essentiellement pour fonction de permettre la reconnaissance du type de wagon dont il s'agit (notamment wagon de télétexte ou wagon de chaînes de bits). La malle de procédure peut comprendre plusieurs paquets, ou en pratique un seul paquet de procédure ; elle est suivie d'une ou plusieurs malles de données.

Les séquences caractéristiques des paquets de données seront très différentes selon qu'elles concernent des paquets de données de télétexte ou des paquets de données en chaînes de bits, ou encore d'autres types de paquets, et toutes ces séquences caractéristiques devront être différentes des séquences caractéristiques SCP concernant les paquets de procédure.

La difficulté est de trouver des séquences caractéristiques aussi peu encombrantes que possible et permettant de distinguer les divers cas de figures.

Tout d'abord, étant donné la nécessité de protéger fortement contre les erreurs ces séquences caractéristiques, on les découpera en octets protégés individuellement. Le nombre d'octets est variable de 0 à 3 dans les divers types de séquences.

On a représenté à la figure 2 des paquets de données appartenant à des wagons transportant des données dont la seule ca-

caractéristique est qu'elles sont découpées en octets. Il s'agit du découpage effectué dans les systèmes qui sont actuellement proposés. On l'a représenté d'une part pour faire comprendre ce qui a déjà été proposé, et d'autre part pour montrer que le procédé de transmission selon l'invention peut accepter, si on le désire, ce type de découpage.

La ligne a représente la structure d'un paquet plein, et la ligne b la structure d'un paquet non plein. A la figure 2 comme aux figures suivantes, on a supprimé du paquet la première partie de l'en-tête dont la structure peut être celle décrite en référence à la figure 1. Les paquets sont donc supposés commencer par la séquence caractéristique SC.

Cette séquence comprend un octet de format FH à quatre bits utiles et quatre bits de redondance codés selon le code de Hamming. Cet octet FH désigne un numéro de format parmi seize, chaque numéro de format correspondant à un nombre d'octets bien déterminé du paquet de données.

Les données proprement dites suivent l'octet FH et sont terminées de préférence par deux octets de correction d'erreurs CCl et CC2. Si N max est le nombre d'octets maximum du paquet, y compris la séquence caractéristique, le nombre d'octets de données utiles est $N_{\max} - 3$.

L'octet FH est toujours le même pour tous les paquets pleins.

La ligne b représente un paquet de données qui n'est pas plein. Il a la même structure que celui de la ligne a, mais il est plus court et l'octet FH qui sert de séquence caractéristique désigne un format quelconque parmi seize. Le nombre N d'octets du paquet doit correspondre à un des quinze formats possibles. Sur les N octets, $N - 3$ sont des données utiles si, comme à la ligne a, le paquet est terminé par un code correcteur d'erreurs à deux octets CCl et CC2.

On peut remarquer ici que dans les wagons de ce type on ne rencontrera jamais de paquets dont la séquence caractéristique soit un code de Hamming nul, c'est-à-dire dont les quatre bits utiles sont 0000. En effet, ce code particulier désignerait, dans

les systèmes proposés actuellement, un paquet ne comportant pas d'octets, ce qui n'a pas de signification.

A la figure 3, on a représenté des paquets de données correspondant à une transmission de télétexte en code ASCII. Dans ce cas, la séquence caractéristique d'en-tête est réduite à zéro et les paquets de données présentent un coefficient de remplissage maximal.

A la ligne a, on a représenté un paquet de données situé en début de malle de télétexte : comme cela est courant en télétexte, on peut définir le début de malle par des codes ASCII particuliers, tels que SOH ("Start Of Heading"), et RS ("Record Separator"). Ces caractères sont des octets avec sept bits utiles et un bit de parité comme tous les caractères du code ASCII. Ils sont suivis d'autres octets du code ASCII, qui constituent le texte à transmettre. Ces octets représentent des lettres, chiffres, symboles, caractères de commande, désignés sur la figure 3 par la référence ISO. Les paquets de données transportant ainsi des octets du code ASCII ont une longueur constante et le dernier octet de chaque paquet est un octet de parité longitudinale PL qui sert de code correcteur d'erreurs à l'ensemble du paquet.

On peut remarquer en effet qu'un code de parité longitudinale, dont chaque bit représente la parité de la somme des bits de même rang de tous les octets qui précèdent, constitue un excellent code correcteur d'erreurs n'occupant qu'un espace restreint, car il permet, en combinaison avec le fait que chaque octet ASCII comporte lui-même un bit de parité, de détecter une erreur dans un octet quelconque et de localiser dans cet octet le bit erroné.

La ligne b de la figure 3 représente un paquet de données en milieu de malle de télétexte. La séquence caractéristique de ce paquet est réduite à zéro. Le remplissage du paquet en données utiles est maximal, seul le dernier octet étant encore un code de parité longitudinale PL.

La ligne c représente un paquet de données qui est une fin de malle de télétexte. Ce paquet comprend d'abord la fin du télétexte, c'est-à-dire des octets ISO, puis des octets qui sont des codes de commande du code ASCII et qui indiquent une fin de

texte, par exemple un octet ETX du code ASCII ("End of Text") et un octet EOT ("End Of Transmission"). Ces codes sont couramment utilisés en télétextes et font partie tout naturellement de la séquence transmise. Si le paquet comporte encore des emplacements 5 d'octets libres, on utilisera ici le code de bourrage NUL pour donner au paquet de fin de malle la même longueur que les autres. Le dernier octet sera encore un octet de parité longitudinale.

Il faut noter ici que le récepteur doit être capable de diriger les caractères reçus après la première partie de l'en-tête 10 des paquets de données vers un décodeur apte à décoder les caractères ASCII, alors que dans le cas de la figure 2, le récepteur devait diriger le premier octet vers un décodeur capable de décoder un code de Hamming. C'est le paquet de procédure ou la malle de procédure placée en tête du wagon correspondant qui prépare le 15 récepteur à recevoir les codes de l'un ou l'autre type et à les aiguiller vers les décodeurs appropriés selon le type de wagon et selon la position des octets dans le paquet.

A la figure 4, on a représenté des paquets de données correspondant à des wagons de transmission de données en chaînes 20 de bits. Là encore, le paquet de procédure qui est en tête du wagon prépare le récepteur à recevoir des paquets de ce type.

Ces paquets comportent comme séquence caractéristique SC essentiellement un octet à quatre bits utiles et quatre bits de redondance codés selon le code de Hamming.

25 Etant donné que la plupart des paquets seront pleins, on ne définit un format de paquets dans la séquence caractéristique SC que lorsque le paquet n'est pas plein. Un des bits de l'octet de Hamming définira si le paquet est plein ou non plein. Un autre bit de l'octet de Hamming définira le chiffre de plus fort poids 30 du nombre de bits du paquet dans les cas où le paquet n'est pas plein. Dans le cas où le paquet est plein, ce bit sera forcé à 1.

Il reste deux bits utiles dans l'octet de Hamming. On peut s'en servir pour définir les début et fin de malle : le premier bit est à 1 si le paquet de données constitue un début de 35 malle, il est à 0 dans le cas contraire. Le deuxième bit est à 1 si le paquet est une fin de malle ; il est à 0 dans le cas con-

traire.

La ligne a de la figure 4 montre un paquet plein qui constitue un début de malle. Le premier bit utile de la séquence caractéristique est à 1 ; le deuxième bit est à 0 si on suppose 5 que la malle ne se termine pas dans ce paquet ; le troisième bit est à 0 et indique qu'il s'agit d'un paquet plein. Le quatrième bit utile est forcé à 1, mais il n'a pas de signification particulière dans ce cas.

L'octet de Hamming est suivi des données transmises en 10 chaînes de bits, deux octets étant réservés à la fin pour constituer des codes correcteurs d'erreurs CCl et CC2. On considère que deux octets permettent une bonne correction d'erreur pour ce type de paquet.

La ligne b de la figure 4 montre un paquet correspondant 15 à un milieu de malle, c'est-à-dire ni un début, ni une fin : le premier bit utile qui est le bit de début de malle est à 0, le second qui est le bit de fin de malle est également à 0 ; le troisième est à 0 et le quatrième à 1 car un paquet de milieu de malle est en principe plein. Les codes correcteurs d'erreur CCl et CC2 20 terminent le paquet après la transmission des données.

La ligne c montre un paquet de fin de malle : le deuxième bit utile est à 1. Le premier est ici à 0, mais il peut être à 1 si la malle ne comporte qu'un paquet qui est à la fois un début et une fin de malle.

25 La ligne d montre un paquet plein, quelle que soit sa place dans la malle : le troisième bit utile est à 0 et le quatrième est forcé à 1.

Enfin, la ligne e montre la constitution d'un paquet non plein : le troisième bit utile est à 1. Le quatrième bit utile 30 finit le chiffre de plus fort poids du nombre de bits contenus dans le paquet ; les autres chiffres de ce nombre sont indiqués dans les deux octets qui suivent et qui sont également codés selon un code de Hamming, de sorte que neuf bits permettent de désigner le nombre de bits du paquet, ce qui est suffisant pour les long- 35 gueurs de paquets considérées (environ 40 octets).

Le paquet est encore terminé par deux octets CCl et CC2

de correction d'erreurs.

Ainsi, c'est seulement dans le cas rare où les paquets ne seront pas pleins que l'on utilisera trois octets pour la séquence caractéristique, un seul octet étant suffisant dans la majorité des cas, cet octet servant en plus à définir des séparations entre malles.

La figure 5 représente la constitution des paquets de procédure : ceux-ci sont tous divisés en octets à quatre bits utiles et quatre bits de redondance codés selon le code de Hamming afin d'assurer une sécurité élevée contre les erreurs. La séquence caractéristique SC de la deuxième partie d'en-tête de ces paquets comprend au moins un octet (de préférence deux) de Hamming.

Bien entendu, il faut que cette séquence caractéristique, indiquant qu'il s'agit d'un paquet de procédure et non pas d'un paquet de données d'un type ou d'un autre, soit reconnaissable des autres séquences caractéristiques déjà décrites, afin que le récepteur puisse distinguer qu'il s'agit d'un paquet de procédure en tête d'un wagon de données dont le type est justement défini par le contenu de ce paquet de procédure.

Les séquences caractéristiques déjà utilisées ou susceptibles d'être utilisées dans les paquets de données déjà décrits sont évidemment indisponibles pour définir une séquence caractéristique de paquets de procédure puisqu'il y aurait confusion possible.

On remarquera cependant que, selon l'invention, on s'est arrangé dans les paquets de données en chaînes de bits pour ne pas utiliser, comme premier octet de la séquence caractéristique, l'octet de Hamming dont les quatre bits utiles sont 0000. De même, dans les paquets de données divisés en octets de la figure 2, il n'apparaîtra jamais l'octet de Hamming 0000 qui indiquerait un paquet de données de longueur 0.

Par contre, dans les malles de données de télétexte pour lesquelles on n'a prévu aucune séquence caractéristique puisqu'il n'y a pas de format à indiquer et que les divisions des paquets en malles peuvent se faire par des caractères de commande faisant justement partie des caractères ASCII, il peut se trouver que

n'importe quel octet du code ASCII se situe en début de paquet. Il faut donc que la séquence caractéristique des paquets de procédure, qu'elle comporte un ou deux octets ou plus, ne se confonde pas avec un octet ou plusieurs octets du code ASCII.

5 Or, justement, l'octet de Hamming nul, qui était le seul pratiquement que l'on puisse utiliser si l'on voulait pouvoir transmettre des paquets du type de la figure 2, correspond à un octet du code ASCII. Cet octet est le caractère NACK du code ASCII. En effet, l'octet de Hamming nul et l'octet NACK du code
10 ASCII s'écrivent tous deux 10101000.

Bien qu'il y ait cette possibilité de confusion, on utilise selon l'invention comme premier octet de la séquence caractéristique des paquets de procédure le code Hamming nul H0, car, en transmission de télétexte à sens unique sur canal de télévision,
15 le symbole NACK ne sera jamais utilisé puisqu'il a justement comme signification, en télétexte, un accusé de réception négatif que le système de transmission de données ne sera pas amené à utiliser (NACK = "Non Acknowledge").

De ce fait, l'apparition du code de Hamming nul, à la
20 première position d'octet de la seconde partie d'en-tête d'un paquet de données, signifie qu'il s'agit d'un paquet de procédure. Toutefois, par sécurité, pour éviter qu'une erreur sur un ou deux bits ne transforme le code de Hamming nul en un octet lisible selon le code ASCII, on renforce la séquence caractéristique des pa-
25 quets de procédure en prévoyant qu'elle comprend soit deux octets de Hamming nul, soit un octet de Hamming nul suivi d'un octet de Hamming 0001 qui, lui aussi, correspond à un octet ASCII qui ne sera normalement pas utilisé en transmission de données : il s'agit de l'octet STX ("Start of Text").

30 La figure 5 montre à la ligne a un premier paquet de procédure dont la séquence caractéristique comprend un octet de Hamming H0 (0000) et un octet de Hamming H1 (0001). Les octets qui suivent, désignés par HX, sont des octets de Hamming quelconques auxquels on donne des significations concernant le décodage des
35 paquets de données qui vont suivre. Ces indications de procédure concernent évidemment, d'une part, le type de données transmises

ultérieurement, de manière que le récepteur soit prêt à recevoir des séquences caractéristiques ultérieures sous une forme bien déterminée ; elles peuvent concerner, d'autre part, toutes sortes d'indications de procédure, telles que la présence, la nature et la position du ou des codes correcteurs d'erreurs utilisés, la longueur maximum des paquets de données, des informations sur la structure des malles, etc.

La ligne b de la figure 5 montre une autre séquence caractéristique possible pour un paquet de procédure ; cette séquence comprend tout simplement deux codes de Hamming nul H0. On peut utiliser la séquence caractéristique H0, H1 par exemple pour un premier paquet d'une malle de procédure et la séquence H0, H0 pour les paquets suivants de la malle de procédure du wagon. On peut prévoir, à la fin de la malle de procédure, un octet HS caractéristique qui indique cette fin.

Il pourra être nécessaire de prévoir un instant de silence à l'émission après la fin des paquets de procédure pour laisser au récepteur le temps de l'interprétation de la procédure avant la réception du premier paquet de données.

On aura remarqué dans tout ce qui précède que toutes les secondes parties d'en-tête des paquets de données ou de procédure sont découpées en octets pour faciliter le travail de décodage du récepteur, sans que les données transmises elles-mêmes soient obligatoirement agencées en octets.

Note : Les codes Hamming et le code ASCII sont bien connus ; on rappelle cependant qu'un octet de Hamming s'écrit b1, b2, b3, b4, b5, b6, b7, b8 où b1 est le premier bit transmis, où b2, b4, b6 et b8 sont les bits utiles de l'octet et b1, b3, b5, b7 sont les bits redondants, avec :

- . $b7 = b8 + b6 + b4$
- . $b5 = b6 + b4 + b2^*$
- . $b3 = b4 + b2^* + b8$
- . $b1 = b2^* + b8 + b6$

où b2* est le complément de b2 et où le signe + désigne une opération OU exclusive.

Ainsi, le code de Hamming nul (0000) correspond à l'octet

2509553

16

10101000 et le code de Hamming 0001 correspond à l'octet 01000000.

REVENDECATIONS.

1. Procédé de transmission à sens unique de données numériques par paquets comprenant chacun un en-tête suivi de données, l'en-tête comprenant une première partie suivie d'une séquence caractéristique (SC), caractérisé par le fait que, dans le but de
5 transmettre des types de paquets différents correspondant respectivement à différents types d'agencement de données numériques qui sont :

- soit des informations agencées en octets codés selon le code ASCII avec bit d'imparité exclusivement,

10 - soit des informations agencées autrement, et notamment en chaînes de bits,

on transmet les paquets de la manière suivante :

a) les paquets sont transmis par wagons ne comprenant que des paquets correspondant à des données de même type,

15 b) chaque wagon comprend d'abord un ou plusieurs paquets dits "de procédure", spécifiques au wagon et indiquant notamment le type de données transmises dans le wagon, des paquets de procédure spécifiques étant prévus notamment pour les wagons de transmission de télétexte en code ASCII et pour les wagons de
20 transmission de données par chaînes de bits,

c) chaque séquence caractéristique (SC) d'un paquet de procédure comprend d'abord un octet (H0) ou plusieurs non susceptibles d'être confondus avec un des symboles du code ASCII utilisés effectivement dans la transmission de données à sens unique,

25 d) les paquets de données correspondant aux wagons de transmission de texte selon le code ASCII comprennent la première partie de l'en-tête suivie immédiatement des données à transmettre agencées en octets selon le code ASCII et utilisant le code de bourrage NUL pour remplir les fins de paquets non pleins,

30 e) les paquets de données correspondant aux wagons de transmission de données agencées en chaînes de bits comprennent une seconde partie d'en-tête comportant au moins :

- un bit indiquant si le paquet est plein ou non,
- et, si le paquet n'est pas plein, un nombre de bits fixe suffisant pour indiquer combien de bits de données sont contenus dans le paquet.

5 2. Procédé selon la revendication 1 caractérisé par le fait que l'octet utilisé au début de la séquence caractéristique des paquets de procédure est le code NACK (10101000).

3. Procédé selon la revendication 2 caractérisé par le fait que l'octet NACK est suivi d'un autre octet NACK ou d'un octet STX (01000000).

4. Procédé selon l'une des revendications 1 à 3 caractérisé par le fait qu'un paquet de procédure spécifique est prévu pour indiquer que le wagon dont il constitue un début va transmettre des données découpées en octets avec un octet de format en seconde partie de chaque paquet de données du wagon, cet octet comportant quatre bits utiles et quatre bits de redondance codés selon un code de Hamming et indiquant selon une table particulière le nombre d'octets de données contenus dans le paquet.

5. Procédé selon l'une des revendications 1 à 4 caractérisé par le fait que les paquets de procédure sont découpés en octets (HX) ayant chacun quatre bits utiles et quatre bits de redondance codés selon un code de Hamming.

6. Procédé selon l'une des revendications 1 à 5 caractérisé par le fait que toutes les séquences caractéristiques de paquets sont découpées en octets.

7. Procédé selon l'une des revendications 1 à 6 caractérisé par le fait que la séquence caractéristique des paquets de données des wagons transportant des données agencées en chaînes de bits comprend un octet à quatre bits utiles et quatre bits de redondance codés selon le code Hamming, deux des bits utiles étant respectivement :

- le bit indiquant si le paquet est plein ou non,
- un bit qui indique le chiffre de poids supérieur du nombre binaire représentant le nombre de bits de données du paquet, ce bit étant forcé à 1 si le paquet est plein.

8. Procédé selon la revendication 7 caractérisé par le

fait que le bit indiquant le chiffre de poids supérieur du nombre de bits du paquet est suivi, uniquement dans le cas où le paquet n'est pas plein, de deux octets de format indiquant le nombre de bits du paquet et codés selon le code de Hamming.

- 5 9. Procédé selon l'une des revendications 7 et 8 caractérisé par le fait que les deux autres bits utiles de l'octet de Hamming du début de la séquence caractéristique des paquets des wagons de chaînes de bits servent à indiquer une division des wagons en malles correspondant à des groupes d'informations à distinguer les uns des autres, et pour cela

- l'un de ces deux autres bits utiles indique si le paquet est un début de malle ou non,
- l'autre indique si le paquet est une fin de malle ou non.

- 15 10. Procédé selon l'une des revendications 1 à 9 caractérisé par le fait que chaque paquet de données d'un wagon de transmission de texte selon le code ASCII est terminé par un octet de parité longitudinale (PL) en tant que code correcteur d'erreur des octets de données qui précèdent.

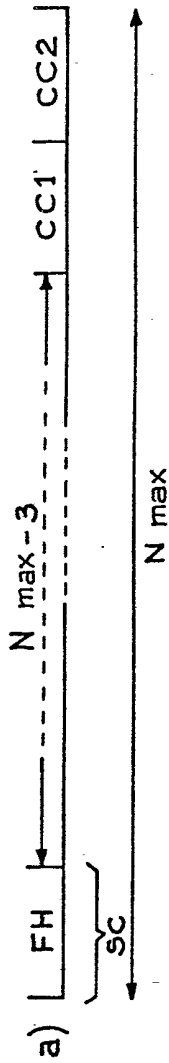
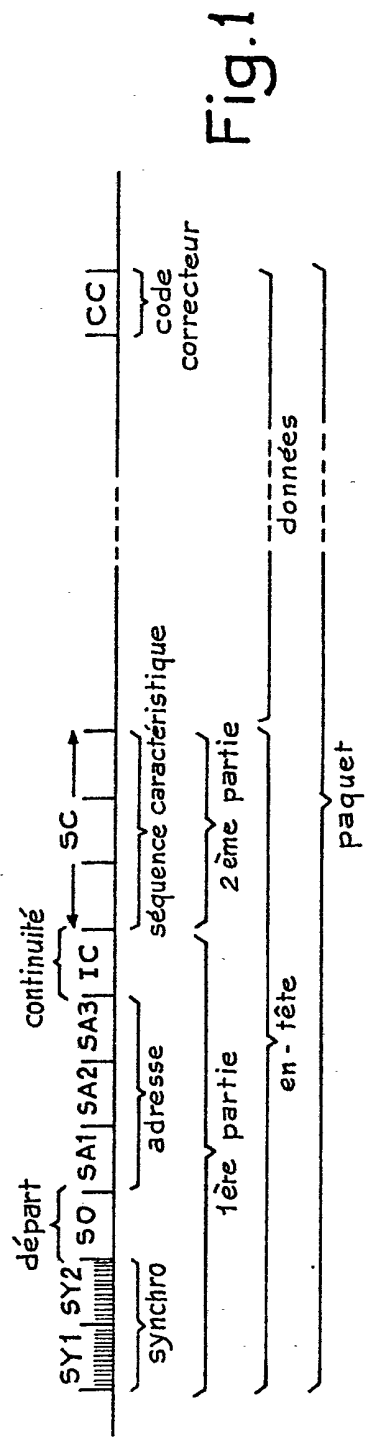


Fig.2

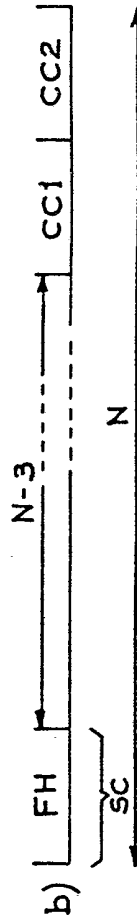
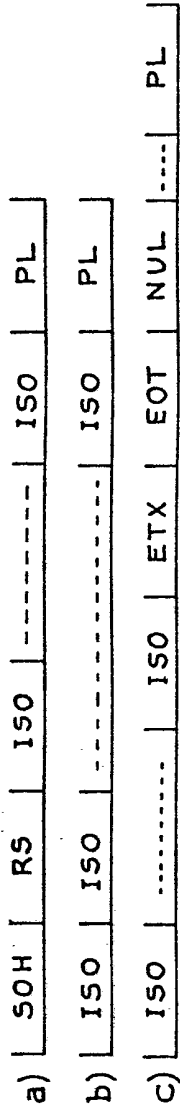


Fig.3



2/2

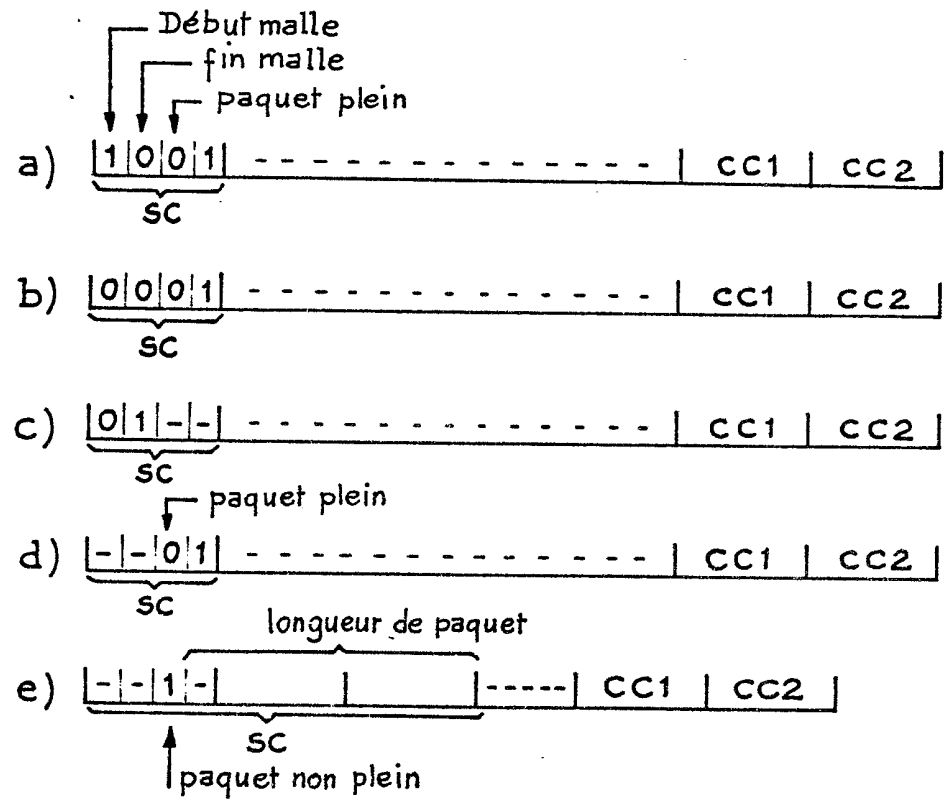


Fig. 4

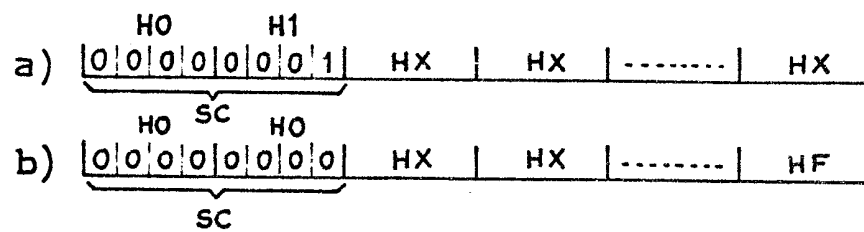


Fig. 5