

12)

DEMANDE DE BREVET D'INVENTION

A1

22) Date de dépôt : 14.01.08.

30) Priorité :

43) Date de mise à la disposition du public de la demande : 17.07.09 Bulletin 09/29.

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

60) Références à d'autres documents nationaux apparentés :

71) Demandeur(s) : CANON KABUSHIKI KAISHA — JP.

72) Inventeur(s) : VISA PIERRE, FROUIN LAURENT et EL KOLLI YACINE.

73) Titulaire(s) :

74) Mandataire(s) : CABINET PATRICE VIDON.

54) PROCÉDE DE TRANSMISSION DE DONNEES PAR UN NOEUD EMETTEUR DANS UN CYCLE DE TRANSMISSION DE DONNEES D'UN RESEAU DE COMMUNICATION SYNCHRONE, PRODUIT PROGRAMME D'ORDINATEUR, MOYEN DE STOCKAGE ET NOEUD EMETTEUR.

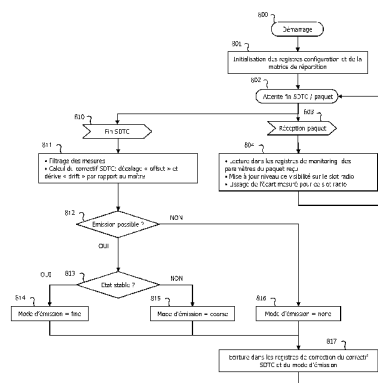
57) L'invention concerne un procédé de transmission de données par un noeud émetteur dans un cycle de transmission de données d'un réseau de communication synchrone (2000) comprenant une pluralité de noeuds dont au moins un noeud esclave (1a) et un noeud maître (5a) mettant en oeuvre un cadencement qui définit le cycle de transmission, le cycle de transmission comprenant un nombre prédéterminé d'intervalles de temps, dits temps de parole, chacun des temps de parole étant associé à un noeud de la pluralité de noeuds, le noeud émetteur possédant un premier mode d'émission. Selon l'invention, un tel noeud émetteur effectue les étapes suivantes :

- obtention (612) d'un début effectif de temps de parole et d'un début estimé de temps de parole pour un noeud de la pluralité de noeuds;

- à partir des débuts estimé et effectif, détermination (811) d'une correction à apporter à une représentation locale au noeud émetteur du cadencement défini par le noeud maître;

- sélection (813, 814, 815) d'un mode d'émission courant, parmi le premier mode d'émission et un second mode d'émission parmi un ensemble de modes d'émission, en

fonction de la correction.



FR 2 926 426 - A1



Procédé de transmission de données par un nœud émetteur dans un cycle de transmission de données d'un réseau de communication synchrone, produit programme d'ordinateur, moyen de stockage et nœud émetteur.

1. DOMAINE DE L'INVENTION

5 Le domaine de l'invention est celui des transmissions de paquets de données dans un réseau de communication synchrone sans-fil par exemple du type maillé (ou MESH en anglais).

Plus précisément, l'invention concerne le contrôle d'accès à un medium de transmission pour un nœud émetteur mis en œuvre dans un tel réseau de communication.

2. ART ANTÉRIEUR

10 Les systèmes de transmission radio à 60 GHz (bande d'ondes RF millimétriques) sont particulièrement bien adaptés pour la transmission de données à très haut débit sur courtes distances. Par exemple, un tel système de transmission est bien adapté à la connectivité entre les différents éléments d'un « home cinema » (ou « cinéma à domicile
15 » en français). En effet, pour ce cas d'utilisation, la portée de la transmission est limitée à une dizaine de mètres, mais les débits mis en jeux sont très élevés, parfois au-delà du gigabit par seconde, de par la nature (aussi bien audio que vidéo) et la haute résolution de l'information transmise.

20 On se place dans la suite dans le cadre d'un réseau de communication synchrone sans-fil par exemple du type maillé comprenant une pluralité de nœuds. Les nœuds opèrent dans la bande d'onde millimétrique (c'est-à-dire autour de 60GHz).

25 Pour permettre à chaque nœud du réseau de transmettre de l'information dans des intervalles de temps régulièrement espacés, on peut utiliser un medium (aussi appelé support de transmission) à accès multiplexé suivant un mode de multiplexage du type TDMA (pour « Time Division Multiple Access » en anglais, et « accès multiple à
30 répartition dans le temps » en français). Selon le multiplexage TDMA, le domaine temporel est divisé en une pluralité de cycles de transmission de données synchrones (ou cycle « SDTC » pour « Synchronous Data Transmission Cycle » en anglais). Chaque cycle SDTC est divisé en une pluralité d'intervalles de temps. Dans un réseau sans-fil utilisant un multiplexage TDMA, à chaque nœud est associé un intervalle de temps (ou « time slot » en anglais) déterminé (aussi appelés par la suite temps de parole) destiné à

l'émission de données par ledit nœud. Pendant les autres temps de paroles (distincts de celui ou ceux affecté(s) à la transmission de ses données), chaque nœud doit orienter son antenne en réception de manière à pouvoir recevoir les données émises par les autres nœuds du réseau de communication durant ces intervalles de temps.

5 Du fait du caractère aléatoire du support de transmission, il est nécessaire d'effectuer de multiples transmissions des données afin d'en garantir la bonne réception au-delà d'un taux d'erreur résiduel prédéfini. En effet, ce type de support de transmission est notamment sensible aux masquages causés, par exemple, par un individu traversant le champ de transmission. Ces multiples transmissions sont
10 préférentiellement assurées par l'intermédiaire de nœuds relais (diversité spatiale), un nœud relais retransmettant les données reçues d'un autre nœud du réseau. Ceci permet de palier à la présence d'obstacles permanents (ou quasi-permanents) dans le réseau de communication qui rendrait inefficace la retransmission de données par le nœud les ayant générées (diversité temporelle).

15 Un tel réseau de communication requiert donc un synchronisme parfait entre le ou les émetteurs et les récepteurs, notamment dans le cas d'un système de diffusion de contenus audio à canaux multiples (« surround sound system » en anglais). A titre d'exemple, un système de diffusion du type « surround sound system » peut comprendre huit hauts parleurs.

20 Généralement dans un système de diffusion du type « surround sound system », l'émetteur (comprenant aussi un décodeur audio) transmet de manière parfaitement synchrone différents canaux audio (issus d'une seule source) à un ensemble de récepteurs, comprenant chacun un haut parleur. Cet ensemble de récepteurs doit restituer globalement le son de manière parfaitement synchronisée afin d'apporter au son un effet
25 de spatialisation.

Le synchronisme au niveau de la couche applicative peut être obtenu grâce au synchronisme des nœuds au niveau de la couche physique. En raison du mode d'accès au medium (de type TDMA), il est impératif de synchroniser les nœuds du réseau afin qu'ils puissent accéder au medium sans perturber le système. En particulier, il faut éviter
30 les collisions entre des paquets de données (aussi appelés trames de données) émis par des nœuds distincts.

Pour assurer la synchronisation du système, un des nœuds du réseau, appelé nœud maître, est chargé de cadencer les cycles SDTC, en émettant des paquets de données à intervalle de temps régulier. Les autres nœuds du réseau, appelés nœuds esclaves, sont capables de reproduire le cadencement du nœud maître à partir de la
5 réception des paquets de données émis par le nœud maître. En cas de masquage, un nœud esclave peut ne pas recevoir les paquets de données du nœud maître. Dans un tel cas, le nœud esclave masqué doit récupérer le cadencement du nœud maître par l'intermédiaire d'un autre nœud esclave synchronisé sur le nœud maître, appelé nœud de référence.

10 Le phénomène de masquage varie en fonction du temps, ainsi un nœud esclave ne se référencera pas continûment avec le même nœud de référence.

On note que cette propriété du réseau interdit l'utilisation de la technique consistant à détecter la fin de réception d'un paquet, puis d'attendre qu'un intervalle de temps entre deux paquets se soit écoulé pour commencer à transmettre. On choisit pour
15 nœud de référence le nœud du réseau qui offre la meilleure précision de synchronisation. Le changement de nœud de référence peut intervenir de façon très fréquente (de l'ordre de la milliseconde) et dépend de la durée du cycle de transmission.

Cette première étape de synchronisation, bien que nécessaire, n'est pas suffisante dès lors que l'on cherche à atteindre un niveau de précision fin (par exemple, de l'ordre
20 de la microseconde). La difficulté réside dans le fait que chaque nœud du réseau est cadencé par sa propre horloge (dite horloge locale) issue d'un oscillateur possédant des caractéristiques qui lui sont propres. En effet, un oscillateur délivre un signal à une fréquence nominale qui peut fluctuer, au cours du temps et en fonction de l'environnement, entre une valeur minimale et une valeur maximale. Il est donc
25 nécessaire que chaque nœud du réseau connaisse en permanence la dérive de son horloge locale par rapport à celle du nœud maître.

Dans le cas où les conditions du réseau le permettraient, il est possible d'obtenir la valeur d'une dérive à un instant donné, par exemple, en mesurant le nombre de coups
30 d'horloge locale entre deux réceptions successives de paquets de données émis par un nœud donné. Dans ce cas, la dérive est l'écart entre le nombre théorique de coups d'horloge locale calculé entre les deux réceptions successives de paquets de données et

le nombre de coups d'horloge locale mesuré. Généralement, cette dérive comprend une erreur induite par les variations aléatoires des différents oscillateurs. En conséquence, un nœud esclave introduit une erreur dans la restitution du cadencement du nœud maître, et donc dans la détermination de son début de temps de parole à l'intérieur d'un cycle SDTC. Cette erreur est fluctuante au cours du temps et crée une gigue (« jitter » en anglais) qui s'ajoute à la gigue initiale du cadencement du nœud maître. Cette gigue est d'autant plus grande qu'il y a de nœuds intermédiaires entre le nœud maître (masqué par rapport au nœud esclave) et le nœud esclave, ce qui affecte d'autant plus la précision globale du système.

Il sera nécessaire de prévoir un intervalle de garde suffisamment large entre les paquets de données successifs pour permettre la compensation de la gigue et ainsi écarter tout risque de collision entre paquets. Par exemple dans le cas d'un réseau de communication sans fil, l'intervalle de garde minimum doit permettre la configuration d'un module de radiocommunication, et notamment l'orientation d'une antenne directive dans la direction d'un émetteur.

Il existe donc un réel besoin de synchroniser ce type de système en minimisant l'incertitude portant sur la reproduction du cadencement du nœud maître dans chaque nœud esclave.

On connaît plusieurs techniques permettant d'assurer une synchronisation d'un nœud esclave sur un nœud maître avec une précision fine et garantie.

Parmi les différentes techniques connues, une technique consiste à mettre en œuvre des échanges de données de synchronisation entre le nœud esclave et le nœud de référence. Plus précisément, les données temporelles obtenues sont utilisées pour calculer le décalage (« shift » en anglais) et la dérive (« drift » en anglais) entre l'horloge du nœud esclave et l'horloge du nœud de référence. Une telle technique est notamment présentée dans le document de brevet US 7,072,432. Cette technique présente deux inconvénients majeurs. Tout d'abord, l'échange de données de synchronisation réduit la bande passante utile disponible, ce qui est préjudiciable dans un réseau de communication sans fil destiné à supporter des applications très haut débit. Un autre inconvénient de cette technique connue réside dans le fait qu'elle est sensible aux

erreurs causées par les variations ponctuelles des oscillateurs des nœuds. Cette technique ne permet donc pas d'assurer un basculement rapide sur un nouveau nœud de référence.

3. OBJECTIFS DE L'INVENTION

5 L'invention, dans au moins un mode de réalisation, a notamment pour objectif de pallier ces différents inconvénients de l'état de la technique.

Plus précisément, un objectif de l'invention, dans au moins un de ses modes de réalisation, est de fournir une technique qui permette de réduire la taille de l'intervalle de temps entre deux paquets de données successifs pour permettre la transmission de paquets de données de taille plus grande.

10 Un autre objectif de l'invention, dans au moins un de ses modes de réalisation, est de mettre en œuvre une telle technique qui permette d'améliorer la détermination du début de temps de parole de chaque nœud esclave à l'intérieur d'un cycle SDTC.

15 Un autre objectif de l'invention, dans au moins un de ses modes de réalisation, est de mettre en œuvre une telle technique qui permette de minimiser les risques de collision entre paquets émis par différents nœuds du réseau.

Un autre objectif de l'invention, dans au moins un de ses modes de réalisation, est de mettre en œuvre une telle technique qui permette de limiter la réduction des intervalles de temps réservés pour le traitement des données au sein des nœuds.

20 Au moins un mode de réalisation de l'invention a également pour objectif de fournir une telle technique qui soit notamment bien adaptée aux réseaux mettant en œuvre un médium à accès multiplexé suivant un mode de multiplexage du type TDMA.

L'invention, dans au moins un de ses modes de réalisation, a encore pour objectif de fournir une telle technique qui soit simple à mettre en œuvre et pour un faible coût.

4. EXPOSÉ DE L'INVENTION

25 Dans un mode de réalisation particulier de l'invention, il est proposé un procédé de transmission de données par un nœud émetteur dans un cycle de transmission de données d'un réseau de communication synchrone comprenant une pluralité de nœuds dont au moins un nœud esclave et un nœud maître mettant en œuvre un cadencement qui
30 définit ledit cycle de transmission, ledit cycle de transmission comprenant un nombre prédéterminé d'intervalles de temps, dits temps de parole, chacun desdits temps de

parole étant associé à un nœud de ladite pluralité de nœuds, ledit nœud émetteur étant un nœud esclave, ledit nœud émetteur possédant un premier mode d'émission associé à un premier intervalle de garde s'inscrivant dans un desdits temps de parole pendant lequel ledit nœud émetteur ne transmet pas de donnée.

5 Selon l'invention, le nœud émetteur effectue les étapes suivantes :

- obtention d'un début effectif de temps de parole et d'un début estimé de temps de parole pour un nœud, dit nœud de référence, de ladite pluralité de nœuds, ledit nœud de référence étant distinct du nœud émetteur ;
- à partir desdits débuts estimé et effectif, détermination d'une correction à apporter à
10 une représentation locale au nœud émetteur dudit cadencement défini par le nœud maître ;
- sélection d'un mode d'émission courant, parmi ledit premier mode d'émission et un second mode d'émission parmi un ensemble de modes d'émission, en fonction de ladite correction, chaque mode dudit ensemble étant associé à un intervalle de garde
15 pendant lequel ledit nœud émetteur ne transmet pas de donnée distinct dudit premier intervalle de garde.

L'invention propose donc de mettre en œuvre au moins deux modes d'émission distincts dans un nœud. Pour ce faire, on prévoit avantageusement d'utiliser un intervalle de garde spécifique pour chaque mode d'émission. La sélection du mode d'émission
20 courant se fait en fonction de l'analyse de la correction. Par exemple, on vérifie si la variation de la correction reste inférieure à un nombre de coups (ou ticks) d'horloge déterminé pendant une durée déterminée.

Comme on le verra dans la suite de la description, dans un mode de réalisation particulier, on utilise la correction pour détecter un état stable de synchronisation du
25 nœud émetteur dans le cycle de transmission. Dans le cas où on a un premier et un second mode d'émission, le basculement du premier mode vers le second mode peut être conditionné par la détection d'un état stable de synchronisation.

Avantageusement, la durée dudit premier intervalle de garde est supérieure à la durée de l'intervalle de garde dudit second mode.

30 Ainsi, moins de bande passante est utilisée par le nœud émetteur afin de compenser la gigue entre le nœud maître et lui-même.

De façon avantageuse, le procédé comprend en outre les étapes suivantes :

- détermination, en fonction dudit mode d'émission courant, d'un ensemble de données à insérer dans un paquet de données à émettre pendant ledit temps de parole associé audit nœud émetteur ;
- 5 - transmission dudit paquet de données selon ledit mode d'émission courant après insertion dudit ensemble de données déterminé dans ledit paquet de données.

Ainsi, le second mode d'émission correspond à un mode « fin » dans lequel le nœud émetteur peut transmettre un ou plusieurs blocs de données supplémentaires. Dans le second mode d'émission le nœud émetteur peut ainsi transmettre des paquets de données de taille plus grande.

Selon un premier mode de réalisation particulier de l'invention, ledit ensemble de données déterminé comprend une information de synchronisation relative audit mode d'émission courant.

Ainsi, grâce à cette information de synchronisation les autres nœuds du réseau peuvent connaître le mode d'émission dans lequel le nœud émetteur a transmis son paquet, et ainsi obtenir une estimation précise du début de temps de parole utilisé par le nœud émetteur.

Selon un second mode de réalisation particulier de l'invention, ledit ensemble de données déterminé comprend, dans le cas du second mode d'émission, des données de répétition correspondant à une retransmission de données émises par un nœud de ladite pluralité de nœuds distinct du nœud émetteur, lesdites données de répétition n'étant pas comprises dans ledit ensemble de données déterminé dans le cas du premier mode d'émission.

Les blocs de données supplémentaires apportés par le second mode d'émission peuvent par exemple servir à apporter plus de redondance de données dans un système synchrone à diversité spatiale. En conséquence, on augmente la capacité de correction d'erreurs dans le ou les nœuds récepteurs. En d'autres termes, on augmente les chances de succès du décodage des données, en augmentant le nombre de données de redondance transmises.

De façon avantageuse, l'étape d'obtention d'un début estimé de temps de parole comprend les étapes suivantes :

- réception d'un paquet de données comprenant une information de synchronisation relative à un mode d'émission utilisé par un nœud distinct dudit nœud émetteur pour transmettre ledit paquet ;
- extraction de ladite information de synchronisation ;
- 5 - calcul dudit début estimé de temps de parole, en tenant compte de ladite information de synchronisation extraite.

Ainsi, on va pouvoir choisir parmi les débuts estimés dans les premier ou second modes d'émission, celui qui sera considéré pour la mesure de l'écart, et ainsi prendre en compte le mode d'émission sélectionné par le nœud ayant émis le paquet de données dans la détermination de la correction. Ceci permet de ne pas interpréter un changement
10 de mode d'émission par le nœud ayant émis le paquet de données, notamment au passage du premier au second mode d'émission, comme une instabilité de la synchronisation avec le nœud maître. Cela permet donc audit nœud émetteur de pouvoir plus rapidement passer lui aussi dans le second mode d'émission.

15 Avantageusement, ladite étape de détermination d'une correction comprend les étapes suivantes :

- détermination d'un décalage entre le cadencement défini par le nœud maître et la représentation locale dudit cadencement, à partir desdits débuts estimé et effectif ;
- détermination d'une dérive entre le cadencement défini par le nœud maître et la
20 représentation locale dudit cadencement, à partir desdits débuts estimé et effectif ;
- détermination de ladite correction à partir dudit décalage et de ladite dérive.

De cette façon, on connaît l'allongement ou le raccourcissement (c'est-à-dire la correction) à appliquer au prochain cycle SDTC pour maintenir un cadencement de cycle SDTC au plus proche de celui du nœud maître, tout en s'affranchissant au mieux,
25 des incertitudes de mesures (décalages) et des variations aléatoires (dérives) des oscillateurs. Dans certains cas, on s'attache à limiter la correction à apporter.

Par exemple, lorsque le signal de début de transmission doit être généré plus tôt que prévu pour corriger une dérive, il est avantageux que la correction soit la plus petite possible afin de limiter la réduction du temps de traitement des données au sein du
30 nœud.

Dans un autre exemple, lorsque la durée du cycle de transmission doit être raccourcie pour corriger une dérive, il est avantageux que la correction soit la plus petite possible afin de limiter la réduction du temps de traitement avant délivrance à la couche applicative.

5 De façon avantageuse, ladite étape de détermination d'une correction comprend une étape de détermination d'une courbe de régression linéaire à partir desdits débuts estimé et effectif, et en ce que ledit décalage et ladite dérive sont évalués à partir de ladite courbe de régression linéaire.

10 Ainsi, on met en œuvre une analyse statistique pour déduire rapidement les paramètres de correction, dans le cas où un nœud reçoit plusieurs paquets émis par plusieurs autres nœuds du réseau distincts du nœud maître.

Avantageusement, ladite étape de détermination d'une correction comprend en outre une étape de sélection de nœud, comprenant elle-même les étapes suivantes, pour chaque nœud de ladite pluralité de nœuds :

- 15
- détermination d'un niveau de visibilité dudit nœud par rapport au nœud maître ;
 - comparaison dudit niveau de visibilité à un seuil déterminé ;
 - si ledit niveau de visibilité est supérieur audit seuil déterminé, sélection dudit nœud pour la détermination de ladite correction.

20 Ainsi, il est possible de garder les nœuds les plus fiables, c'est-à-dire les nœuds présentant un fort taux de visibilité, pour le calcul de la correction. Par exemple, le nœud esclave ne conserve que les mesures obtenues avec le nœud maître et avec les nœuds ayant reçu le paquet radio du nœud maître au cours du cycle SDTC précédent. Avec les nœuds présentant un fort taux de visibilité, on pourra donc faire une comparaison pertinente entre le dernier écart mesuré (entre le temps estimé et le début effectif du paquet) et les mesures précédentes. De cette façon, toute mesure faussée par une
25 variation brutale d'un oscillateur est écartée.

30 Avantageusement, le nœud émetteur sélectionne le second mode d'émission lorsque ladite correction déterminée est stable sur un nombre prédéterminé de cycles de transmission, et en ce que le nœud émetteur sélectionne le premier mode d'émission lorsque ladite correction déterminée est instable.

Ainsi, dans le cas où la synchronisation des nœuds esclaves avec le nœud maître est instable, c'est-à-dire dans le cas d'une mise en marche ou d'une réinitialisation du système, ou dans le cas de fortes interférences perturbant le canal de transmission, les intervalles de garde sont renforcés, alors que quand la synchronisation des nœuds esclaves avec le nœud maître est stable, c'est-à-dire dans le cas d'un régime de marche nominal, les intervalles de garde sont minimisés et la bande passante disponible sur le canal de transmission est maximisée.

Dans un autre mode de réalisation, l'invention concerne un produit programme d'ordinateur téléchargeable depuis un réseau de communication et/ou enregistré sur un support lisible par ordinateur et/ou exécutable par un processeur, ledit produit programme d'ordinateur comprenant des instructions de code de programme pour la mise en oeuvre du procédé de transmission précité, lorsque ledit programme est exécuté sur un ordinateur.

Dans un autre mode de réalisation, l'invention concerne un moyen de stockage, éventuellement totalement ou partiellement amovible, lisible par un ordinateur, stockant un jeu d'instructions exécutables par ledit ordinateur pour mettre en oeuvre le procédé de transmission précité.

Dans un autre mode de réalisation, l'invention concerne un nœud émetteur comprenant des moyens de transmission de données dans un cycle de transmission de données d'un réseau de communication synchrone comprenant une pluralité de nœuds dont au moins un nœud esclave et un nœud maître mettant en oeuvre un cadencement qui définit ledit cycle de transmission, ledit cycle de transmission comprenant un nombre prédéterminé d'intervalles de temps, dits temps de parole, chacun desdits temps de parole étant associé à un nœud de ladite pluralité de nœuds, ledit nœud émetteur étant un nœud esclave, ledit nœud émetteur possédant un premier mode d'émission associé à un premier intervalle de garde s'inscrivant dans un desdits temps de parole pendant lequel ledit nœud émetteur ne transmet pas de donnée.

Selon l'invention, le nœud émetteur comprend :

- des moyens d'obtention d'un début effectif de temps de parole et d'un début estimé de temps de parole pour un nœud, dit nœud de référence, de ladite pluralité de nœuds, ledit nœud de référence étant distinct du nœud émetteur ;

- des moyens de détermination d'une correction à apporter à une représentation locale au nœud émetteur dudit cadencement défini par le nœud maître ;
- des moyens de sélection d'un mode d'émission courant, parmi ledit premier mode d'émission et un second mode d'émission parmi un ensemble de modes d'émission, en fonction de ladite correction, chaque mode dudit ensemble étant associé à un intervalle de garde pendant lequel ledit nœud émetteur ne transmet pas de donnée distinct dudit premier intervalle de garde.

5

Les avantages des produit programme d'ordinateur, moyen de stockage et nœud émetteur sont sensiblement les mêmes que ceux du procédé de transmission et ne sont donc pas repris ci-après.

10

Avantageusement, la durée dudit premier intervalle de garde est supérieure à la durée de l'intervalle de garde dudit second mode.

De façon avantageuse, le nœud émetteur comprend en outre :

- des moyens de détermination d'un ensemble de données à insérer dans un paquet de données à émettre pendant ledit temps de parole associé audit nœud émetteur ;
- des moyens de transmission dudit paquet de données comprenant ledit ensemble de données, selon ledit mode d'émission courant.

15

Selon un premier mode de réalisation particulier de l'invention, ledit ensemble de données comprend une information de synchronisation relative audit mode d'émission courant.

20

Selon un second mode de réalisation particulier de l'invention, ledit ensemble de données comprend, dans le cas du second mode d'émission, des données de répétition correspondant à une retransmission de données émises par un nœud de ladite pluralité de nœuds distinct du nœud émetteur, lesdites données de répétition n'étant pas comprises dans ledit ensemble de données dans le cas du premier mode d'émission.

25

De façon avantageuse, lesdits moyens d'obtention d'un début estimé de temps de parole comprennent :

- des moyens de réception d'un paquet de données comprenant une information de synchronisation relative à un mode d'émission utilisé par un nœud distinct dudit nœud émetteur pour transmettre ledit paquet ;
- des moyens d'extraction de ladite information de synchronisation ;

30

- des moyens de calcul dudit début estimé de temps de parole, lesdits moyens de calcul tenant compte de ladite information de synchronisation.

Avantageusement, lesdits moyens de détermination d'une correction comprennent :

- 5 - des moyens de détermination d'un décalage entre le cadencement défini par le nœud maître et la représentation locale dudit cadencement, lesdits moyens de détermination déterminant ledit décalage à partir desdits débuts estimé et effectif ;
- des moyens de détermination d'une dérive entre le cadencement défini par le nœud maître et la représentation locale dudit cadencement, lesdits moyens de
- 10 détermination déterminant ladite dérive à partir desdits débuts estimé et effectif ;
- des moyens de détermination de ladite correction, lesdits moyens de détermination déterminant ladite correction à partir dudit décalage et de ladite dérive.

De façon avantageuse, lesdits moyens de détermination d'une correction comprennent :

- 15 - des moyens de détermination d'une courbe de régression linéaire, lesdits moyens de détermination déterminant ladite courbe de régression linéaire à partir desdits débuts estimé et effectif ;
- des moyens d'évaluation dudit décalage et de ladite dérive utilisant ladite courbe de régression linéaire.

20 Avantageusement, lesdits moyens de détermination d'une correction comprennent en outre des moyens de sélection d'un nœud comprenant eux-mêmes :

- des moyens de détermination d'un niveau de visibilité dudit nœud par rapport au nœud maître ;
- des moyens de comparaison dudit niveau de visibilité à un seuil déterminé.

25 De façon avantageuse, le nœud émetteur sélectionne le second mode d'émission lorsque ladite correction est stable sur un nombre prédéterminé de cycles de transmission, et en ce que le nœud émetteur sélectionne le premier mode d'émission lorsque ladite correction est instable.

5. LISTE DES FIGURES

30 D'autres caractéristiques et avantages de modes de réalisation de l'invention apparaîtront à la lecture de la description suivante, donnée à titre d'exemple indicatif et

non limitatif (tous les modes de réalisation de l'invention ne sont pas limités aux caractéristiques et avantages des modes de réalisation décrits ci-après), et des dessins annexés, dans lesquels :

- 5 - la figure 1 présente un exemple de système de communication radio à 60 GHz selon un mode de réalisation préférentiel de l'invention ;
- la figure 2 présente un schéma d'un cycle complet de transmission de données synchrones, selon un mode de réalisation particulier de l'invention ;
- la figure 3 présente un schéma d'un cycle complet de transmission de données synchrones avec des temps d'émission correspondant aux modes « grossier » et
10 « fin », selon un mode de réalisation particulier de l'invention ;
- la figure 4a présente un exemple d'architecture du module de communication synchrone apparaissant sur la figure 1, selon un mode de réalisation particulier de l'invention ;
- la figure 4b présente un exemple d'architecture du contrôleur SDTC apparaissant
15 sur la figure 4a, selon un mode de réalisation particulier de l'invention ;
- la figure 5 présente un exemple de matrice de répartition de la bande passante synchrone, selon un mode de réalisation particulier de l'invention ;
- la figure 6a présente les étapes principales d'un algorithme de traitement mis en œuvre par le bloc d'émission, compris dans un nœud esclave et apparaissant sur
20 la figure 4a, selon un mode de réalisation particulier de l'invention ;
- la figure 6b présente les étapes principales d'un algorithme de traitement mis en œuvre par le bloc de réception, compris dans un nœud esclave et apparaissant sur la figure 4a, selon un mode de réalisation particulier de l'invention ;
- la figure 7 présente les étapes principales d'un algorithme de traitement mis en œuvre par le contrôleur SDTC, compris dans un nœud esclave et apparaissant sur
25 la figure 4b, selon un mode de réalisation particulier de l'invention ;
- la figure 8 présente les étapes principales d'un algorithme de traitement mis en œuvre par l'unité centrale, compris dans un nœud esclave et apparaissant sur la figure 4b, selon un mode de réalisation particulier de l'invention ;
- 30 - la figure 9 présente un exemple de courbe de régression, selon un mode de réalisation particulier de l'invention ;

- la figure 10 présente les étapes principales d'un algorithme de traitement mis en œuvre par le contrôleur SDPC, compris dans un nœud esclave et apparaissant sur la figure 4a, selon un mode de réalisation particulier de l'invention ;
- la figure 11a présente un exemple d'architecture d'un compteur de cycles SDTC, selon un mode de réalisation particulier de l'invention ;
- la figure 11b présente un exemple d'architecture du compteur temps avant transmission, selon un mode de réalisation particulier de l'invention ;
- la figure 11c présente un exemple d'architecture du compteur de cycles SDPC, selon un mode de réalisation particulier de l'invention ; et
- la figure 11d présente un exemple d'architecture d'un dispositif comprenant un compteur temps avant intervalle de temps et un compteur de numéro d'intervalle de temps courant, selon un mode de réalisation particulier de l'invention.

6. DESCRIPTION DÉTAILLÉE

Sur toutes les figures du présent document, les éléments et étapes identiques sont désignés par une même référence numérique.

6.1 Exemple de système de communication radio à 60 GHz (système audio 8 voies)

On décrit, en relation avec la **figure 1**, un exemple de système de communication radio à 60 GHz selon un mode de réalisation préférentiel de l'invention.

Dans ce mode de réalisation, le système de communication 2000 selon l'invention comprend neuf nœuds de communication. Plus précisément, le système de communication 2000 comprend huit nœuds 1a, 2a, 3a, 4a, 6a, 7a, 8a, et 9a de type « WAR » (pour « Wireless Audio Renderer » en anglais ou « récepteur audio sans fil » en français) comprenant chacun des moyens de restitution de canal audio numérique (ou « Digital Audio Channel Amplifier » en anglais) 1b, 2b, 3b, 4b, 6b, 7b, 8b, et 9b. Ces moyens de restitution comprennent, par exemple, un haut-parleur (ou « speaker » en anglais).

Le système de communication 2000 comprend en outre un nœud 5a de type « WAD » (pour « Wireless Audio Decoder » en anglais ou « décodeur audio sans fil » en français) comprenant lui-même un décodeur audio multi voies (ou « Surround sound decoder » en anglais) 5b. Ce décodeur audio multi voies 5b est intégré, par exemple,

dans un écran plat et est susceptible de transmettre de manière synchronisée, via le système de communication 2000, les différents canaux audio associés à la vidéo affichée sur l'écran.

5 Chacun des nœuds 1a, 2a, 3a, 4a, 5a, 6a, 7a, 8a, et 9a comprend un module de communication synchrone, noté « SCM » (pour « Synchronous Communication Module » en anglais).

6.2 Principe général du système

10 L'ensemble de la bande passante disponible délivrée par le système est découpé en canaux virtuels synchrones. Le débit utile est caractérisé par la fréquence de traitement des canaux virtuels (par exemple 8KHz) ainsi que par la taille des échantillons (par exemple 48 bits). Dans le présent exemple, un canal virtuel (« VC » pour « Virtual Channel » en anglais) a un débit constant de 384 Kbps (kilobits par seconde). La séquence complète, qui comprend un échantillon de chacun des canaux virtuels disponibles, constitue un cycle complet de traitement de données synchrones
15 (appelé cycle SDPC pour « Synchronous Data Processing Cycle » en anglais). Dans le présent exemple, où la fréquence de traitement des canaux virtuels est de 8KHz, la durée d'un cycle SDPC est de 125µs.

Un canal audio de résolution 96KHz-24bits utilisera donc 6 canaux virtuels soit, par exemple pour un système audio 8 voies (classiquement appelé « home cinema 7.1 »),
20 un total de 48 canaux virtuels. Le débit utile nécessaire pour le transfert de l'information audio est donc de 18,432 Mbps (mégabit par seconde). Si l'on ajoute 10 canaux virtuels supplémentaires (alloués à l'ensemble des nœuds de transmission du système) pour le transfert d'informations supplémentaires (protocole de contrôle du système, commande utilisateur,...), alors le cycle SDPC comprend 58 canaux virtuels, soit un débit utile de
25 22,272 Mbps.

Ainsi, parmi les canaux virtuels 0 à 47 qui sont transmis par le module de communication synchrone 5 (« SCM#0 ») du nœud 5a :

- les canaux virtuels 0 à 5 sont décodés puis traités par le module de communication synchrone 9 (« SCM#1 ») du nœud 9a, qui correspond au haut
30 parleur avant-central (« Front-Center Speaker » en anglais) ;

- les canaux virtuels 6 à 11 sont décodés puis traités par le module de communication synchrone 4 (« SCM#2 ») du nœud 4a, qui correspond au haut parleur avant-gauche (« Front-Left Speaker» en anglais) ;
- 5 - les canaux virtuels 12 à 17 sont décodés puis traités par le module de communication synchrone 7 (« SCM#3 ») du nœud 7a, qui correspond au haut parleur avant-droit (« Front-Right Speaker» en anglais) ;
- les canaux virtuels 18 à 23 sont décodés puis traités par le module de communication synchrone 3 (« SCM#4 ») du nœud 3a, qui correspond au haut parleur latéral-gauche (« Side-Left Speaker» en anglais) ;
- 10 - les canaux virtuels 24 à 29 sont décodés puis traités par le module de communication synchrone 8 (« SCM#5 ») du nœud 8a, qui correspond au haut parleur latéral-droit (« Side-Right Speaker» en anglais) ;
- les canaux virtuels 30 à 35 sont décodés puis traités par le module de communication synchrone 2 (« SCM#6 ») du nœud 2a, qui correspond au haut parleur arrière-gauche (« Rear-Left Speaker» en anglais) ;
- 15 - les canaux virtuels 36 à 41 sont décodés puis traités par le module de communication synchrone 1 (« SCM#7 ») du nœud 1a, qui correspond au haut parleur arrière-droit (« Rear-Right Speaker» en anglais) ;
- les canaux virtuels 42 à 47 sont décodés puis traités par le module de communication synchrone 6 (« SCM#8 ») du nœud 6a, qui correspond au caisson de graves (« Subwoofer » en anglais).
- 20

Les canaux virtuels 48 à 57 sont décodés puis traités par tous les nœuds du système pour l'échange de données supplémentaires de manière synchrone. Ainsi, à chaque module de communication synchrone est alloué au moins un canal virtuel pour l'émission de ces données supplémentaires vers les autres nœuds. A titre d'exemple, on peut avoir le schéma d'allocation suivant :

- 25

- au module de communication synchrone 5 (« SCM#0 ») du nœud 5a, on alloue les canaux virtuels 48 et 49,
- au module de communication synchrone 9 (« SCM#1 ») du nœud 9a, on alloue le canal virtuel 50,
- 30

- au module de communication synchrone 4 (« SCM#2 ») du nœud 4a, on alloue le canal virtuel 51,
- au module de communication synchrone 7 (« SCM#3 ») du nœud 7a, on alloue le canal virtuel 52,
- 5 - au module de communication synchrone 3 (« SCM#4 ») du nœud 3a, on alloue le canal virtuel 53,
- au module de communication synchrone 8 (« SCM#5 ») du nœud 8a, on alloue le canal virtuel 54,
- au module de communication synchrone 2 (« SCM#6 ») du nœud 2a, on alloue
10 le canal virtuel 55,
- au module de communication synchrone 1 (« SCM#7 ») du nœud 1a, on alloue le canal virtuel 56,
- au module de communication synchrone 6 (« SCM#8 ») du nœud 6a, on alloue le canal virtuel 57.

15 Une séquence de traitement d'un conteneur de canal virtuel (« VC Chunk » pour « Virtual Channel Chunk » en anglais) comprend un ensemble d'échantillons d'un même canal virtuel (« Virtual Channel Samples » en anglais). Cette séquence de traitement doit être effectuée pendant un cycle complet de transmission de données synchrone (appelé cycle SDTC). La séquence de transmission de ce conteneur de canal
20 virtuel (« VC Chunk ») doit être effectuée pendant la durée du cycle SDTC suivant. En conséquence, la durée d'un cycle SDTC est un multiple de la durée d'un cycle SDPC. Le rapport du cycle de transmission synchrone (cycle SDTC) sur le cycle de traitement synchrone (cycle SDPC) est appelé « STPR » (pour « Synchronous Transmission to
25 Processing Ratio » en anglais). Ce rapport (nombre entier) définit ainsi la taille de la séquence de transmission d'un conteneur de canal virtuel, de manière commune quels que soient les canaux virtuels. Dans un mode de réalisation particulier, le rapport « STPR » est égal à 16. Dans ce cas, la séquence de transmission d'un conteneur de canal virtuel est constituée de 96 octets (c'est-à-dire la taille des échantillons (48 bits) que multiplie le rapport « STPR » (16)).

30 Pendant un cycle SDTC, chaque nœud de transmission envoie au moins un paquet de données (aussi appelé trame de données). Dans un mode de réalisation

particulier, dix paquets de données (de taille égale) sont émis selon la séquence suivante :

- les deux premiers paquets de données du cycle SDTC sont émis par le module de communication synchrone (SCM#0) du nœud 5a de type « WAD » ;
- 5 - les huit paquets de données suivants sont émis par les huit nœuds 1a, 2a, 3a, 4a, 6a, 7a, 8a, et 9a de type « WAR » (un paquet de données par nœud « WAR »), par exemple, selon l'ordre suivant : SCM#1, SCM#2, SCM#3, SCM#4, SCM#5, SCM#6, SCM#7 et SCM#8.

6.3 Structure des paquets de données découpés en blocs de données radio

10 On présente maintenant, en relation avec la **figure 2**, un schéma d'un cycle complet de transmission de données synchrones (cycle SDTC).

Dans le présent mode de réalisation, dix paquets de données 10 à 19 (pas tous représentés sur la figure 2) (générés par les modules de communication synchrone (SCM) embarqués dans les nœuds du système) sont transmis pendant un cycle SDTC
15 d'une durée de 2ms. Le temps de transmission d'un paquet de données est légèrement inférieur à $200\mu\text{s}$, ce qui permet de transmettre 3456 octets de données utiles (soit un débit moyen de 138.24Mbps) pendant cette durée. Comme illustré, le paquet de données est transmis à l'intérieur d'un intervalle de temps (référéncé « Radio Slot » sur la figure 2) comprenant :

- 20 - un intervalle de configuration (référéncé « RF setup » sur la figure 2), pendant lequel le module de communication synchrone est configuré pour la prochaine émission ou réception ;
- un intervalle d'incertitude (référéncé « Gap » sur la figure 2) permettant de palier à l'incertitude portant sur le début de transmission du paquet.

25 On s'intéresse désormais au traitement des données. Dans le cas où le rapport « STPR » est égal à 16, chaque module de communication synchrone (SCM) doit traiter 16 cycles SDPC 1058 à 2558 (pas tous représentés sur la figure 2) pendant un même cycle SDTC. Dans le présent exemple, la durée de chaque cycle SDPC est de $125\mu\text{s}$. Pendant chaque cycle SDPC, le module de communication synchrone (SCM) effectue le
30 traitement des 58 échantillons des canaux virtuels. Comme illustré sur la figure 2, les échantillons 1000, 1030, 1031 et 1057 sont traités pendant le cycle SDPC 1058,

l'échantillon 1100 pendant le cycle SDPC 1158, et les échantillons 2500, 2530 et 2557 pendant le cycle SDPC 2558.

Dans le mode de réalisation illustré, chaque paquet de données 11 comprend :

- un préambule 54 ;
- 5 - un champ en-tête 50, noté « RPH » (pour « Radio Packet Header » en anglais) ;
et
- un champ de données utiles 55, noté « RPP » (pour « Radio Packet Payload » en anglais).

10 Plus précisément, le champ en-tête 50 (formé par exemple de 36 octets) comprend :

- un champ « hop_id » 51. Le champ « hop_id » indique le numéro de l'intervalle de temps affecté au paquet de données dans le cycle SDTC ;
- un champ « tx_mode » 52. Le champ « tx_mode » indique le niveau de synchronisation du nœud qui émet le paquet de données. Comme on le verra par
15 la suite, le nœud selon l'invention met en œuvre deux niveaux de synchronisation distincts correspondants à deux modes d'émission distincts :
 - un niveau de synchronisation grossier, dans ce cas le nœud met en œuvre un premier mode de synchronisation, appelé par la suite « mode grossier » (ou « coarse mode » en anglais) ;
 - 20 ○ un niveau de synchronisation fin, dans ce cas le nœud met en œuvre un deuxième mode de synchronisation, appelé par la suite « mode fin » (ou « fine mode » en anglais) ;

25 Il est possible de mettre en œuvre plus de deux niveaux de synchronisation distincts chacun correspondants à un mode d'émission distinct. Ceci permet, de manière dépendante d'un niveau de confiance dans la stabilité (en fonction du nombre de cycles SDTC consécutifs où la synchronisation est considérée comme stable), de faire évoluer le mode de synchronisation sélectionné et par conséquent le mode d'émission correspondant; et

- un champ « hop_cpt » 53. Comme indiqué précédemment, un nœud esclave pour
30 lequel le nœud maître est masqué peut récupérer le cadencement du nœud maître par l'intermédiaire d'un ou plusieurs autres nœuds esclaves (dits nœuds de

référence) synchronisés (directement ou indirectement) sur le nœud maître. Le champ « hop_cpt » indique le nombre de nœuds qui se trouve entre le nœud qui émet le paquet de données et le nœud maître du réseau du point de vue de la synchronisation. On note que ce nombre reflète la situation au cours du cycle SDTC précédent. Le champ « hop_cpt » est nul dans le cas d'un paquet émis par le nœud maître.

Le champ de données utiles 55 comprend un ensemble de blocs de données radio (« RDB » pour « Radio Data Block » en anglais) (numérotés de 20 à 49 sur la figure 2) d'une longueur de 114 octets après encodage. Cet ensemble comprend des données propres au nœud émetteur et des données de répétition provenant d'autres nœuds du réseau. Dans un mode de réalisation particulier, chaque paquet de données comprend un champ de données utiles 55 comprenant 30 blocs de données radio (numérotés de 20 à 49 sur la figure 2).

On présente désormais, en relation avec la **figure 3**, un schéma d'un cycle SDTC avec des temps d'émission correspondant aux modes « grossier » et « fin ».

Par souci de simplification de la description, on se limitera, dans toute la suite de ce document, à décrire le cas particulier où les intervalles de temps associés à chacun des nœuds sont identiques (c'est-à-dire de même durée).

Sur la figure 3, le mode « grossier » correspond à la ligne 3001 et le mode « fin » à la ligne 3002. Le cycle SDTC est délimité par les deux repères 300 et 301. Le cycle SDTC est divisé en 10 intervalles de temps (Slot#0 à Slot#9).

Les repères 310 à 319 (sur la ligne 3001) indiquent le début nominal de transmission des paquets de données 320 à 329 dans le mode « grossier ».

Les repères 330 à 339 (sur la ligne 3001) indiquent le début nominal de transmission des paquets de données 340 à 349 dans le mode « fin ».

A titre d'exemple, pour le module de communication synchrone 8 (« SCM#5 ») du nœud 8a, qui utilise l'intervalle de temps « Slot#6 » pour transmettre :

- quand ce module fonctionne dans le mode « grossier », il transmet son paquet de données à partir du repère 316 ;
- en revanche, quand ce module fonctionne dans le mode « fin », il transmet son paquet de données à partir du repère 336.

En réalité, à cause des variations des oscillateurs des nœuds du réseau et de leur dérive respective, le début de transmission peut avoir lieu dans une plage de temps qui est égale à deux fois l'intervalle entre le début de l'intervalle de temps et le début nominal de transmission. Ceci définit l'intervalle de garde (« Guard Interval » en anglais) entre deux paquets de données successifs (zone non hachurée sur la figure 3).

Dans un mode de réalisation particulier de l'invention, chaque nœud esclave fonctionne dans le mode « grossier » au démarrage du système. Il en va de même lorsqu'on détecte un état de synchronisation imprécise, c'est-à-dire que le nœud esclave est susceptible de produire une erreur importante sur la détermination de son temps de parole. Comme on le verra par la suite, lorsque la synchronisation devient stable, le nœud esclave bascule dans le mode « fin ».

Au démarrage du système, le nœud maître fonctionne dans le mode « fin » si les variations de son oscillateur sont inférieures à l'intervalle de garde prévu dans le mode « fin ». Si cette condition n'est pas remplie, il est possible d'envisager de rallonger la durée des intervalles de temps réservés au nœud maître avec un intervalle de garde supplémentaire. Grâce à cet intervalle de garde supplémentaire, le nœud maître peut constamment émettre dans le mode « fin ». Cependant, cette solution n'est pas optimale en terme de bande passante. En effet, chaque intervalle de temps du nœud maître est rallongé de l'intervalle de garde supplémentaire.

Pour remédier à cet inconvénient, selon un mode de réalisation particulier de l'invention, on applique uniquement l'intervalle de garde supplémentaire au premier intervalle de temps du nœud maître. Les autres intervalles de temps du nœud maître sont gérés de la même façon qu'un nœud esclave avec le processus de surveillance du réseau. En conséquence, sur la figure 3, seul le premier intervalle de temps du nœud maître (Slot#0) est un peu plus long que les autres intervalles de temps.

Dans un mode de réalisation particulier de l'invention, si on considère les hypothèses suivantes :

- une horloge système de 24.576MHz avec une précision de ± 100 ppm ;
 - une durée de cycle SDTC de 2ms, soit 49152 coups (ou ticks) d'horloge ;
- et

- une incertitude de mesure sur l'instant de réception de paquet de $\pm 120\text{ns}$, soit ± 3 coups (ou ticks) d'horloge ;

alors :

- 5 - la variation possible d'un oscillateur sur la durée du cycle SDTC est de $\pm 203\text{ns}$, soit ± 5 coups (ou ticks) d'horloge ;
- l'écart maximum entre un nœud esclave synchronisé et le nœud maître sur la durée du cycle SDTC est de $\pm 406\text{ns}$, soit ± 10 coups (ou ticks) d'horloge ;
- 10 - avec 10 intervalles de temps, le nombre de nœuds de référence entre le nœud maître et un nœud esclave est au maximum de 8, ce qui donne un écart maximum entre un nœud esclave et le nœud maître de $\pm 3654\text{ns}$, soit ± 90 coups (ou ticks) d'horloge. En additionnant cet écart maximum avec l'incertitude de mesure sur l'instant de réception de paquet, on obtient une durée minimum de l'intervalle pour le mode « grossier » de $\pm 3774\text{ns}$, soit ± 93 coups (ou ticks) d'horloge ;
- 15 - la durée minimum de l'intervalle pour le mode « fin » est de $\pm 120\text{ns}$, soit ± 3 coups (ou ticks) d'horloge (la durée minimum de l'intervalle pour le mode « fin » est égale à l'incertitude de mesure sur l'instant de réception de paquet).

20 Ainsi, le gain entre le mode « grossier » et le mode « fin » est d'environ $7,30\mu\text{s}$. A titre d'exemple, pour une durée de transmission de bloc de données radio de $6,08\mu\text{s}$, il est possible de transmettre dans le mode « fin » au moins un bloc de données radio supplémentaire par paquet de données.

25 On note qu'en réservant 22 coups (ou ticks) d'horloge d'intervalle de garde supplémentaire, pour couvrir la variation de l'oscillateur du nœud maître, il reste 49130 coups (ou ticks) d'horloge comme temps de transmission possible, soit 4913 coups (ou ticks) (c'est-à-dire $199,91\mu\text{s}$) par intervalle de temps. Pour le premier intervalle de temps du nœud maître, la durée est de $2000 - 9 \times 199,91 = 200,81\mu\text{s}$.

30 Dans le présent mode de réalisation, la durée du préambule 54 est de $7,68\mu\text{s}$, la durée du champ en-tête 50 de $5,76\mu\text{s}$, et la durée de l'intervalle de configuration de

3,5 μ s. Ainsi, pour chaque intervalle de temps, la durée maximale du champ de données utiles 55 plus l'intervalle est de 182,97 μ s.

On note également que :

- dans le mode « grossier », un intervalle minimum de 7,55 μ s permet de laisser une durée maximale de 175,42 μ s pour le champ de données utiles 55, soit 28,85 blocs de données radio ;
- dans le mode « fin », un intervalle minimum de 0,244 μ s permet de laisser une durée maximale de 182,97 μ s pour le champ de données utiles 55, soit 30,09 blocs de données radio ;

Dans l'exemple numérique précité, on peut donc émettre dans le mode « grossier » des paquets contenant 28 blocs de données radio et dans le mode « fin » des paquets contenant 30 blocs de données radio.

6.4 Architecture d'un module de communication synchrone

On présente, en relation avec la **figure 4a**, un exemple d'architecture du module de communication synchrone (SCM) selon un mode de réalisation particulier de l'invention.

Comme illustré, le module de communication synchrone 3000 comprend une unité centrale de traitement 87 (qui est par exemple un microprocesseur référencé CPU) coopérant avec un bloc de mémoire d'exécution 86 (RAM) et un bloc de mémoire non volatile 85 (ROM). L'unité centrale 87 communique avec un module radio 84 via une interface 93 (CPU IF). L'interface 93 gère notamment les interruptions à destination du microprocesseur (CPU), ainsi que les échanges de données entre les différents blocs du module radio 84 et l'unité centrale 87.

Ainsi, à l'initialisation du système, l'unité centrale 87 effectue le transfert des informations de configuration depuis la mémoire non volatile 85 vers l'interface 93.

Dans une variante de réalisation, il est possible d'envisager de sélectionner une configuration parmi une pluralité de configurations système (par exemple, stockée préalablement dans la mémoire non volatile 85), en fonction d'informations (par exemple, sur l'application et l'environnement du système) provenant d'un utilisateur.

Après l'initialisation du système, l'unité centrale 87 et la mémoire non volatile 85 sont utilisés pour mettre en oeuvre l'analyse du réseau et la détermination de la dérive de l'horloge locale du nœud esclave par rapport à celle du nœud maître.

5 Le module radio 84 coopère avec un bloc d'interface radio 60GHz 99 et un bloc d'interface audio synchrone 98. Le bloc d'interface audio synchrone 98 communique avec un module de traitement de canaux audio 82. Le module radio 84 permet donc de transférer des données synchrones depuis le bloc d'interface radio 60GHz 99 vers le module de traitement de canaux audio 82, et inversement.

10 Dans un premier mode de réalisation, le module de traitement de canaux audio 82 est un moyen de restitution de canal audio numérique (« Digital Audio Channel Amplifier » en anglais).

Dans un deuxième mode de réalisation, le module de traitement de canaux audio 82 est un décodeur audio multi voies (« Surround sound decoder » en anglais).

15 Dans le module radio 84 on trouve, côté traitement des données synchrones, des blocs d'écriture 89 et de lecture 88 de conteneurs de canal virtuel (« VC Chunk »). Le bloc d'écriture 89 est chargé de construire les blocs de données radio (RDB) associés à un canal virtuel, au fur et à mesure de l'écriture de chacun des échantillons du canal virtuel (« VC sample » en anglais) pendant le cycle SDPC auquel l'échantillon est associé. A l'inverse, le bloc de lecture 88 est chargé de l'écriture de chaque échantillon
20 extrait des blocs de données radio (RDB) reçus pendant le cycle SDPC.

Dans le cas où le rapport « STPR » est égal à 16, le module radio 84 doit effectuer, pendant un cycle SDTC, la lecture et l'écriture des échantillons associés aux canaux virtuels qu'il doit traiter pendant les 16 cycles SDPC.

25 Dans le module radio 84 on trouve, côté transmission des données synchrones, des blocs d'émission 95 et de réception 94 de paquets de données. Le bloc d'émission 95 intègre notamment les fonctions de modulation (par exemple, une modulation du type OFDM) et d'insertion de préambule, celui-ci permettant d'indiquer le début d'émission d'un paquet de données. Le bloc de réception 94 réalise les fonctions inverses de celles mises en oeuvre dans le bloc d'émission 95, à savoir notamment les fonctions de
30 démodulation et de détection de préambule.

Dans le mode de réalisation illustré, le module radio 84 comprend en outre :

- 5 - un bloc d'encodage 92 permettant de coder les blocs de données radio (RDB) à transmettre. Après codage, les blocs de données radio codés sont stockés dans une mémoire tampon (ou buffer) d'émission de paquets de données. Cette mémoire tampon (ou buffer) est, par exemple, compris dans le bloc d'émission 95 ;
- un bloc de décodage 90 permettant de décoder les blocs de données radio (RDB) reçus. Après décodage, les blocs de données radio décodés sont stockés dans une mémoire tampon (ou buffer) de réception de paquets de données. Cette mémoire tampon (ou buffer) est, par exemple, compris dans le bloc de réception 94 ;
- 10 - un bloc de retransmission 91 de blocs de données radio (RDB) permettant de récupérer certains des blocs de données radio reçus par le bloc de réception 94, puis de les stocker (dans la mémoire tampon d'émission de paquets de données précité) avant qu'ils soient retransmis par le bloc d'émission 95 ;
- un premier bloc de synchronisation 97 (aussi appelé par la suite contrôleur SDTC) permettant de contrôler l'enchaînement régulier des cycles SDTC. Le contrôleur SDTC 97 permet également de définir l'instant d'accès (c'est-à-dire le début de temps de parole) au canal de transmission (ou médium) pour l'envoi de paquets de données (par le bloc d'émission 95). Dans le présent exemple, ce contrôleur SDTC 97 permet le traitement en émission/réception de 10 paquets de données pendant un cycle SDTC. Dans le cas d'un nœud esclave, le contrôleur SDTC 97 coopère avec l'unité centrale 87 pour définir le cadencement du cycle SDTC en fonction des paquets radio reçus par le bloc de réception 94.

On note que le contrôleur SDTC 97 contrôle un contrôleur SDPC 96 (aussi appelé par la suite contrôleur SDPC) qui permet de contrôler l'enchaînement régulier des cycles SDPC. Ainsi, on obtient pour chaque canal virtuel un transfert parfaitement synchrone des échantillons entre les blocs d'écriture 89 et de lecture 88 et le bloc d'interface audio synchrone 98.

6.5 Architecture des blocs de synchronisation

On présente maintenant, en relation avec la **figure 4b**, un exemple d'architecture du contrôleur SDTC 97 (embarqué dans chaque nœud esclave) selon un mode de réalisation particulier de l'invention.

Le contrôleur SDTC 97 communique avec l'unité centrale 87 via l'interface 93 (CPU IF). Les échanges de données se font par l'intermédiaire de trois groupes de registres :

- des registres de configuration 405, dans lesquels sont stockés des paramètres statiques, par exemple :
 - 5 ○ le nombre d'intervalles de temps par cycle SDTC ;
 - les numéros des intervalles de temps affectés en transmission au nœud considéré ;
 - la durée initiale d'un cycle SDTC ;
 - 10 ○ la durée initiale d'un cycle SDPC ;
 - la durée initiale des intervalles de temps ;
 - la durée de l'intervalle de garde ;
 - le délai entre le début d'un intervalle de temps et le début de transmission en mode « grossier » ;
 - 15 ○ le délai entre le début d'un intervalle de temps et le début de transmission en mode « fin » ;
 - le temps de propagation des données à travers les différents blocs du module de communication synchrone en charge de l'émission et de la réception des paquets ;
- 20 - des registres de surveillance (« monitoring registers » en anglais) 404, dans lesquels sont stockés, par exemple, les paramètres suivants, après chaque réception de paquet :
 - la valeur d'un signal Rx_hop_id 421 (généralisé par le bloc de réception 94) indiquant le numéro de l'intervalle de temps concerné ;
 - 25 ○ la valeur d'un signal Rx_hop_cpt 423 (généralisé par le bloc de réception 94) indiquant le nombre de nœuds de référence entre le nœud esclave émetteur du paquet et le nœud maître ;
 - la valeur d'un signal Rx_mode 422 (généralisé par le bloc de réception 94) indiquant le mode de synchronisation (aussi appelé par la suite mode d'émission) dans lequel travail le nœud esclave émetteur : mode « grossier »
 - 30 ou « fin » (ou respectivement « coarse » ou « fine ») ;

- la valeur de l'écart mesuré, en nombre de coups (ou ticks) d'horloge locale, entre le temps estimé et le début effectif du paquet ;
- des registres de correction 406, dans lesquels sont stockés, par exemple, les paramètres suivants, à la fin de chaque cycle SDTC :
 - 5 ○ le paramètre correctif de décalage (« shift » en anglais) à apporter au cadencement du cycle SDTC ;
 - le paramètre correctif de dérive (« drift » en anglais) à apporter au cadencement du cycle SDTC ;
 - le mode d'émission pour le prochain cycle SDTC : mode « grossier », mode
10 « fin », ou aucune émission autorisée.

Le contrôleur SDTC 97 génère et transmet deux types de notifications (aussi appelée interruptions) vers l'unité centrale 87 :

- une notification de nouvelles données dans les registres de surveillance 404. Cette notification est transmise vers l'unité centrale 87 via le signal Rx 401. Ce
15 signal Rx 401 est activé après réception d'un paquet de données et après écriture des données dans les registres de surveillance 404 ;
- une notification de fin de cycle SDTC. Cette notification est transmise vers l'unité centrale 87 via le signal Req 402. Ce signal Req 402 est activé pendant le dernier intervalle de temps pour indiquer la fin du cycle SDTC. Cette
20 notification permet de fournir à l'unité centrale 87 les paramètres de correction du prochain cycle SDTC.

Par ailleurs, l'unité centrale 87 informe le contrôleur SDTC 97, via un signal Ack 403, que des nouvelles données sont disponibles dans les registres de correction 406. Ce signal Ack 403 est activé après calcul des paramètres de correction et après écriture des
25 paramètres dans les registres de correction 406.

Comme indiqué précédemment, le contrôleur SDTC 97 contrôle le contrôleur SDPC 96.

Le contrôleur SDPC 96 permet de calculer la durée du cycle de traitement de données synchrones (cycle SDPC). Le calcul tient compte de la dérive du nœud esclave par rapport au nœud maître. Le calcul est déclenché après réception d'un signal
30 SDTC_start 410 (généré par le contrôleur SDTC 97) indiquant un nouveau cycle SDTC

et d'un signal SDTC_duration 415 indiquant la durée du nouveau cycle SDTC. Après calcul, on génère le signal de début de cycle de traitement de données synchrone, utilisé par les blocs d'écriture 89 et de lecture 88 (décrit précédemment en relation avec la figure 4a).

5 Le contrôleur SDTC 97 communique avec le bloc d'émission 95 via les signaux suivants :

- le signal Tx_start 411 indiquant le début de transmission d'un paquet de données ;
- le signal Tx_hop_id 412 indiquant le numéro de l'intervalle de temps utilisé ;
- 10 - le signal Tx_mode 413 indiquant le mode d'émission dans lequel travail le nœud esclave : mode « grossier » ou « fin » ;
- le signal Tx_hop_cpt 414 indiquant le nombre de nœuds de référence entre le nœud esclave et le nœud maître, pour le cycle SDTC précédent.

15 Le bloc de réception 94 communique avec le contrôleur SDTC 97 via les signaux suivants :

- le signal Rx_start 420 indiquant le début de réception d'un paquet de données ;
- le signal Rx_hop_id 421 indiquant le numéro de l'intervalle de temps utilisé par le nœud de référence ;
- le signal Rx_mode 422 indiquant le mode de synchronisation dans lequel travail le nœud esclave émetteur : mode « grossier » ou « fin » ;
- 20 - le signal Rx_hop_cpt 423 indiquant le nombre de nœuds de référence entre le nœud esclave émetteur et le nœud maître, pour le cycle SDTC précédent.

25 Comme on le verra ci-après, les blocs d'émission 95 et de réception 94 ont accès à une matrice de répartition de la bande passante synchrone 407 (cf. figure 5) qui est initialisée au démarrage du système par l'unité centrale 87.

La **figure 6a** présente les étapes principales d'un algorithme de traitement mis en œuvre par le bloc d'émission 95 (décrit en relation avec la figure 4a), compris dans un nœud esclave, selon un mode de réalisation particulier de l'invention.

30 Après démarrage du nœud esclave (étape 600), on attend dans une étape 601 un signal (aussi appelé événement) Tx_start 411 indiquant le début d'une transmission.

Dans une étape 602, on reçoit le signal Tx_start 411.

Puis, dans une étape 603, on construit le champ en-tête 50 du paquet de données à transmettre. Dans cette étape 603, le bloc d'émission 95 construit le champ en-tête 50 en copiant :

- 5 - la valeur du signal Tx_hop_id 412 dans le champ « hop_id » 51 (le champ « hop_id » indique le numéro de l'intervalle de temps affecté au paquet de données) ;
- la valeur du signal Tx_mode 413 dans le champ « tx_mode » 52 (le champ « tx_mode » indique le niveau de synchronisation du nœud qui émet le paquet de données) ; et
- 10 - la valeur du signal Tx_hop_cpt 414 dans le champ « hop_cpt » 53 (le champ « hop_cpt » indique le nombre de nœuds de référence qui se trouvent entre le nœud esclave qui émet le paquet de données et le nœud maître du réseau).

Toujours dans cette étape 603, le bloc d'émission 95 transmet tout d'abord au bloc d'interface radio 99 l'en-tête 50 du paquet de données, puis les données applicatives agencées en blocs de données radio. Comme indiqué précédemment, le nombre de blocs de données radio à transmettre dépend du mode d'émission sélectionné. Par exemple, on peut transmettre dans le mode « grossier » des paquets contenant 28 blocs de données radio et dans le mode « fin » des paquets contenant 30 blocs de données radio. Le bloc d'émission 95 effectue en outre une opération de lecture de la matrice de répartition 407 de façon à connaître la nature des blocs de données radio à transmettre. Par exemple, s'il s'agit d'un bloc de données radio contenant des données propres, alors ces données sont fournies par le bloc d'encodage 92. En revanche, s'il s'agit d'un bloc de données radio contenant des données reçues d'un autre nœud et à répéter (c'est-à-dire à relayer dans le paquet émis par le nœud considéré), alors ces données sont fournies par le bloc de retransmission 91. Ensuite, on revient à l'étape 601.

La **figure 6b** présente les étapes principales d'un algorithme de traitement mis en œuvre par le bloc de réception 94 (décrit en relation avec la figure 4a), compris dans un nœud esclave, selon un mode de réalisation particulier de l'invention.

Après démarrage du nœud esclave (étape 610), on attend dans une étape 611 un préambule radio 54 marquant le début d'un paquet de données.

Dans une étape 612, on détecte le préambule radio 54.

Dans une étape 613, on récupère dans l'en-tête du paquet de données reçu :

- l'identifiant de l'intervalle de temps. En d'autres termes, le signal Rx_hop_id 421 prend la valeur du champ « hop_id » 51 du paquet de données reçu ;
- le niveau de synchronisation du nœud distant. En d'autres termes, le signal Rx_mode 422 prend la valeur du champ « Tx_mode » 52 du paquet de données reçu ;
- le nombre de nœuds de référence qui se trouvaient entre le nœud esclave qui émet le paquet de données et le nœud maître du réseau au cours du cycle SDTC précédent. En d'autres termes, le signal Rx_hop_cpt 423 prend la valeur du champ « hop_cpt » 53 du paquet de données reçu.

Dans une étape 614, on notifie la réception d'un paquet de données au contrôleur SDTC 97. Pour ce faire, le bloc de réception 94 génère et transmet un signal Rx_start 420 vers le contrôleur SDTC 97.

Puis, dans une étape 615, le bloc de réception 94 lit les données du champ de données utiles 55 du paquet de données reçu. Ces données sont agencées en blocs de données radio. Le nombre de blocs de données radio reçu dépend du mode d'émission qui a été utilisé par le nœud émetteur (c'est-à-dire du signal Tx_mode 413). La lecture de la matrice de répartition 407 permet d'identifier la nature des blocs de données radio présents dans le paquet de données. Par exemple, s'il s'agit d'un bloc de données radio (RDB) contenant des données destinées au nœud, alors ces données sont transmises au bloc de décodage 90. En revanche, s'il s'agit d'un bloc de données radio contenant des données à répéter, alors ces données sont transmises au bloc de retransmission 91. Il est important de noter qu'un même bloc de données radio peut réunir les deux conditions : « RDB en réception » et « RDB à répéter ». Ensuite, on revient à l'étape 611.

6.6 Description de la matrice de répartition de la bande passante synchrone

On présente désormais, en relation avec la **figure 5**, un exemple de matrice de répartition de la bande passante synchrone 407 (aussi appelée matrice de répartition de capacité du canal) (cf. figure 4b) selon un mode de réalisation particulier de l'invention.

Dans cet exemple, le cycle complet de transmission de données synchrones SDTC est découpé en 300 blocs de données radio. Ainsi, jusqu'à 10 paquets de données sont transmis par cycle SDTC, chaque paquet de données comprenant jusqu'à 30 blocs

de données radio. Comme déjà indiqué, le paquet de données contient 28 blocs de données radio lorsqu'il est transmis en mode « grossier », et 30 blocs de données radio lorsqu'il est transmis en mode « fin ».

5 Les lignes 100, 110, 120, 130, 140, 150, 160, 170, 180, 190 de la matrice décrivent chacune le contenu des champs RDB des 10 paquets de données émis respectivement par les modules de communication synchrone SCM#0 (qui est le seul à émettre deux paquets consécutifs et qui est le nœud maître), SCM#1, SCM#2, SCM#3, SCM#4, SCM#5, SCM#6, SCM#7, SCM#8. Chaque ligne de la matrice comprend 30 éléments. Chaque élément est représentatif d'un bloc de données radio selon sa position
10 dans le paquet de données. Par souci de clarté de la figure 5, les champs plus larges décrivent 6 éléments de la matrice (par exemple, le champ 101 contient les éléments représentatifs de 6 blocs RDB(0, j), avec j variant de 0 à 5), alors que les autres champs ne décrivent qu'un seul élément (par exemple, le champ 105 contient un seul élément représentatif du bloc RDB(0, 24)).

15 La matrice de répartition décrit la signification des différents champs RDB(i,j), ($i < 10$; $j < 30$), au cours d'un cycle SDTC :

- Pour 58 de ces blocs de données radio, l'information utile transportée contient un conteneur de canal virtuel (« VC Chunk ») nouvellement transmis. Il s'agit des champs 101 à 105, 111 à 115, ainsi que des champs 125, 135, 145, 155, 165,
20 175, 185, et 195 ;
- Pour les 242 autres blocs de données radio, l'information utile transportée contient un conteneur de canal virtuel préalablement reçu puis retransmis, ou bien issu du même cycle SDTC(N), ou bien issu d'un ou plusieurs cycles SDTC(N-m) précédents. Dans le présent exemple, on limitera la retransmission
25 uniquement au cycle précédent ($m=1$). Ainsi, les champs 131 à 134, 143, 144, 157, 158, 162, 172, 184, 192, 197, 198, 199, 251 et 281 sont représentatifs de 67 blocs RDB retransmis pendant le même cycle SDTC(N) que celui de leur première transmission ;
- Les champs restants, représentatifs des 175 blocs RDB restants, identifient des
30 blocs de données radio retransmis, dont la première transmission a été effectuée lors du cycle précédent SDTC(N-1).

Dans un mode de réalisation particulier, on impose que chaque bloc de données radio RDB soit transmis jusqu'à 6 fois avec un minimum de 4 fois, ce qui caractérise en partie la répartition des blocs de données radio RDB au sein de la matrice de répartition 407. Les données audio qui correspondent aux champs 101 à 104 et 111 à 114 sont transmises 5 fois. Les données supplémentaires de contrôle qui correspondent aux champs 105, 115, 125, 135, 145, 155, 165, 175, 185, et 195 sont transmises 4 fois si tous les paquets contiennent 28 blocs de données radio ou 6 fois s'ils contiennent 30 blocs de données radio.

Pour chaque élément de la matrice de répartition 407, on trouve au minimum les informations suivantes :

- un champ « VCB » indiquant un numéro de bande synchrone (« Virtual Channel Bank » en anglais) à laquelle est associé le canal virtuel du conteneur de canal virtuel (« VC Chunk ») ; c'est un identifiant commun aux canaux virtuels affectés à la transmission d'un même contenu. Dans le présent exemple, on a un VCB par voie audio, c'est-à-dire 8 VCB au total. Un VCB est donc composé de 6 canaux virtuels. On note que les canaux virtuels qui appartiennent à un même VCB indiquent une même valeur prédéfinie pour ce champ VCB. Ce champ est optionnel et a, par exemple, une valeur indéfinie pour les canaux virtuels 48 à 57, destinés au transport de données synchrones supplémentaires ;
- un champ « VC » indiquant le numéro de canal virtuel du conteneur de canal virtuel (« VC Chunk ») ;
- un champ « Rx » pouvant prendre la valeur « 1 » ou « 0 ». Lorsque le champ « Rx » prend la valeur « 1 », cela signifie que le nœud est destinataire du canal virtuel. Lorsque le champ « Rx » prend la valeur « 0 », cela signifie que le nœud n'est pas destinataire du canal virtuel ;
- un champ « STDC » indiquant le cycle au cours duquel le conteneur de canal virtuel (« VC Chunk ») a été transmis pour la première fois. Le champ « STDC » prend la valeur « 0 » pour indiquer un cycle courant et la valeur « 1 » pour indiquer un cycle précédent ;
- un champ « Repeat » pouvant prendre la valeur « 1 » ou « 0 ». Lorsque le champ « Repeat » prend la valeur « 1 », cela signifie qu'il s'agit d'un bloc de données

radio (RDB) à répéter. Lorsque le champ « Repeat » prend la valeur « 0 », cela signifie qu'il ne s'agit pas d'un bloc de données radio (RDB) à répéter.

A titre d'exemple, pour l'élément de la matrice correspondant au sixième bloc RDB du premier paquet de données émis dans un cycle SDTC, on a l'écriture suivante :

5 RDB(1 ;6)= { VCB =6 ; VC=30 ; Rx=INDEFINI; SDTC=0 ; Repeat = 0 ; }. Par souci de clarté, les champs « Rx », « SDTC » et « Repeat » ne sont pas représentés sur la figure 5.

6.7 Fonctionnement du contrôleur de synchronisation

On présente, en relation avec la **figure 7**, les étapes principales d'un algorithme de traitement mis en œuvre par le contrôleur SDTC 97 (décrit en relation avec la figure 4b), compris dans un nœud esclave, selon un mode de réalisation particulier de l'invention.

10

Après démarrage du nœud esclave (étape 700), le contrôleur SDTC 97 attend dans une étape 701 la réception d'un paquet de données.

15 Dans une étape 702, on reçoit un paquet de données.

Dans une étape 703, le contrôleur SDTC 97 lit les valeurs des signaux Rx_hop_id 421, Rx_mode 422 et Rx_hop_cpt 423.

Dans une étape 704, on vérifie si le nœud émetteur est un nœud fiable. En cas de vérification positive, on passe à une étape 705, sinon on retourne à l'étape 701. Dans un mode de réalisation particulier, on considère que le nœud émetteur est fiable lorsque le signal Rx_hop_cpt est égal à « 0 » (c'est-à-dire lorsque le nœud émetteur est le nœud maître) ou « 1 » (c'est-à-dire lorsque le nœud émetteur a reçu un paquet de données émis par le nœud maître).

20

Dans l'étape 705, le contrôleur SDTC 97 calcule tout d'abord le temps d'attente « Time_before_SDTC » (en nombre de coups (ou ticks) d'horloge locale) avant le début du prochain cycle SDTC. Le temps d'attente « *Time_before_SDTC* » est défini par :

25

$$Time_before_SDTC = Tsdtc_init - \sum_{k=0}^{Rx_hop_id-1} Tslot_init(k) - rx_latency$$

où :

- *Tsdtc_init* est la durée initiale d'un cycle SDTC ;
 - *Tslot_init(k)* est la durée initiale d'un intervalle de temps *k* ;
- 30

- *rx_latency* est une constante permettant de prendre en compte le temps de propagation à travers le bloc d'émission 95 et le bloc d'interface radio 99 du nœud émetteur, le temps de propagation sur le medium, et le temps de propagation à travers le bloc d'interface radio 99 et le bloc de réception 94 du nœud récepteur.

5

Dans un mode de réalisation particulier, les paramètres *Tsdtc_init*, *Tslot_init(k)* et *rx_latency* sont lus dans les registres de configuration 405.

La valeur « *Time_before_SDTC* » permet de configurer le compteur décrit ci-après en relation avec la figure 11a.

10

On présente, en relation avec la **figure 11a**, un exemple d'architecture du compteur de cycles SDTC selon un mode de réalisation particulier de l'invention. Le compteur de cycles SDTC 4000 génère le signal SDTC_start 410 indiquant le début d'un nouveau cycle SDTC.

Le compteur de cycles SDTC 4000 comprend :

15

- un compteur 1101 qui délivre en sortie « Out » un signal de sortie 1106, représentatif d'une durée exprimée en nombre de coups (ou ticks) d'horloge. Plus précisément, le compteur 1101 reçoit sur une première entrée « Reset » un signal de réinitialisation « Reset1 » 1102, sur une deuxième entrée « Clock ENA » un signal de commande « CENa1 » 1103, et sur une troisième entrée

20

« Clock » un signal d'horloge « Clk » 1104 ;

- un comparateur 1105 recevant sur une première entrée le signal de sortie 1106 et sur une deuxième entrée le signal « *Time_before_SDTC* » 1107 représentatif de la durée du prochain cycle SDTC (exprimée en nombre de coups (ou ticks) d'horloge locale). Le comparateur 1105 délivre en sortie le signal SDTC_start 410 correspondant au résultat de la soustraction entre le signal de sortie 1106 et le signal « *Time_before_SDTC* ». Ainsi, le comparateur 1105 génère en sortie une impulsion lorsque la valeur du compteur 1101 a atteint la valeur « *Time_before_SDTC* ».

25

De nouveau en référence à la figure 7, dans l'étape 705, on fixe ensuite le signal

30

Tx_mode 413 (indiquant le mode d'émission) au mode « grossier », puis on initialise le

compteur de nœuds de référence Tx_hop_cpt 414 avec la valeur Rx_hop_cpt du paquet reçu incrémenté de 1.

Dans une étape 706, le contrôleur SDTC 97 attend la notification d'un début de cycle de transmission SDTC.

5 Dans une étape 707, on active le signal SDTC_start 410 indiquant un début de cycle de transmission SDTC.

Dans une étape 708, le contrôleur SDTC 97 bascule dans un mode d'attente de début d'intervalle de temps.

10 Dans une étape 709, on détecte le début d'un intervalle de temps. On note que le début du premier intervalle de temps coïncide avec le début du cycle SDTC.

A ce stade, on active le dispositif 7000 décrit ci-après en relation avec la figure 11d. Ce dispositif 7000 permet de signaler le début de l'intervalle de temps suivant.

15 On présente désormais, en relation avec la **figure 11d**, un exemple d'architecture du dispositif 7000 précité selon un mode de réalisation particulier de l'invention. Le dispositif 7000 comprend un compteur temps avant intervalle de temps 7001 et un compteur de numéro d'intervalle de temps courant 7002.

Le compteur 7001 comprend :

- un compteur 1131 qui délivre en sortie « Out » un signal de sortie 1136, représentatif d'une durée exprimée en nombre de coups (ou ticks) d'horloge. Plus précisément, le compteur 1131 reçoit sur une première entrée « Reset » un signal de réinitialisation « Reset4 » 1132, sur une deuxième entrée « Clock ENA » un signal de commande « CENa4 » 1133, et sur une troisième entrée « Clock » un signal d'horloge « Clk » 1134 ;
 - un comparateur 1135 recevant sur une première entrée le signal de sortie 1136 et sur une deuxième entrée le signal « Slot_duration » 1137 représentatif de la durée de l'intervalle de temps courant. Le comparateur 1135 délivre en sortie un signal « Slot_start » 1144 correspondant au résultat de la soustraction entre le signal de sortie 1136 et le signal « Slot_duration ». Ainsi, le comparateur 1135 génère en sortie une impulsion lorsque la valeur du compteur 1131 a atteint la valeur « Slot_duration ». Il est important de noter qu'au démarrage du système $slot_duration = Tslot_init(k)$, où $Tslot_init(k)$ est la durée initiale de
- 20
- 25
- 30

l'intervalle de temps k (à ce stade $k=0$).

Le compteur 7002 comprend un compteur 1141 qui délivre en sortie « Out » un signal de sortie « Slot_number » 1150 indiquant le numéro du slot courant. Plus précisément, le compteur 1141 reçoit sur une première entrée « Reset » un signal de réinitialisation « Reset5 » 1142, sur une deuxième entrée « Clock ENA » un signal de commande « CENa5 » 1143, et sur une troisième entrée « Clock » le signal « Slot_start » 1144 généré en sortie du comparateur 1145.

De nouveau en référence à la figure 7, dans une étape 710, on vérifie si l'intervalle de temps (détekté à l'étape 709) est utilisé pour une émission de paquet ou pour une réception de paquet (dans ce dernier cas, le paquet est émis par un autre nœud). Si l'intervalle de temps, appelé par la suite « intervalle d'émission » est utilisé pour une émission de paquet, alors on passe à une étape 711. En revanche, si l'intervalle de temps, appelé par la suite « intervalle de réception » est utilisé pour une réception de paquet, alors on passe à une étape 713.

Dans l'étape 711, on vérifie si la transmission est bien possible sans risque de collision. En d'autres termes, on vérifie si la transmission est autorisée. Si le signal Tx_mode 413 indique un mode d'émission « grossier » ou « fin », alors la transmission est autorisée et on passe à une étape 712, sinon on passe à une étape 715. On note que le signal de délai avant transmission « tx_delay » prend la valeur T_{coarse} si on est en mode « grossier » ou la valeur T_{fine} si on est en mode « fin ». La valeur du signal « tx_delay » permet de configurer le compteur décrit ci-après en relation avec la figure 11b.

On présente désormais, en relation avec la **figure 11b**, un exemple d'architecture du compteur temps avant transmission 5000 selon un mode de réalisation particulier de l'invention. Le compteur temps avant transmission 5000 génère le signal Tx_start 411 indiquant le début de la transmission.

Le compteur temps avant transmission 5000 comprend :

- un compteur 1111 qui délivre en sortie « Out » un signal de sortie 1116, représentatif d'une durée exprimée en nombre de coups (ou ticks) d'horloge. Plus précisément, le compteur 1111 reçoit sur une première entrée « Reset » un signal de réinitialisation « Reset2 » 1112, sur une deuxième entrée « Clock

ENA » un signal de commande « CENa2 » 1113, et sur une troisième entrée « Clock » un signal d'horloge « Clk » 1114 ;

- un comparateur 1115 recevant sur une première entrée le signal de sortie 1116 et sur une deuxième entrée le signal « tx_delay ». Le comparateur 1115 délivre en sortie le signal Tx_start 411 correspondant au résultat de la soustraction entre le signal de sortie 1116 et le signal « tx_delay ». Ainsi, le comparateur 1115 génère en sortie une impulsion lorsque la valeur du compteur 1111 a atteint la valeur « tx_delay ».

De nouveau en référence à la figure 7, dans l'étape 712, on envoie au bloc d'émission 95 le signal Tx_start 411. Ensuite, on revient à l'étape 708.

Dans l'étape 713 (cas d'un intervalle de réception), on vérifie si il y a un masquage au moyen d'un compteur temps. Plus précisément, si aucun paquet de données n'est reçu après que le compteur temps indique qu'une durée d'attente déterminée est écoulée depuis l'activation du compteur temps, alors cela signifie que le nœud distant n'a pas émis son paquet ou qu'il y a un obstacle qui bloque la transmission sans fil. Si le paquet de données est reçu avant la fin du comptage du compteur temps, alors on passe à une étape 714, sinon on passe à l'étape 715.

Dans l'étape 714, le contrôleur SDTC 97 réalise les opérations suivantes :

- lecture des paramètres du paquet via les signaux Rx_hop_id, Rx_mode, et Rx_hop_cpt ;
- calcul de l'écart « Diff » (en nombre de coups (ou ticks) d'horloge) entre le temps de réception estimé et la réception effective du paquet.

Lorsque le signal Rx_mode indique que le mode de synchronisation dans lequel travail le nœud émetteur est le mode « grossier », l'écart « Diff » est défini par :

$$Diff(Rx_hop_id) = Trx - \sum_{i=0}^{Rx_hop_id-1} Tslot(i) - rx_latency - Tcoarse$$

- Lorsque le signal Rx_mode indique que le mode de synchronisation dans lequel travail le nœud esclave est le mode « fin », l'écart « Diff » est défini par :

$$Diff(Rx_hop_id) = Trx - \sum_{i=0}^{Rx_hop_id-1} Tslot(i) - rx_latency - Tfine$$

où

- *Trx* est la valeur de la sortie 1106 du compteur 1101 lors de la réception du signal *Rx_start* 420 (indiquant le début de réception du paquet de données) ;
 - *Tslot(i)* est la durée de l'intervalle de temps *i*. Dans un mode de réalisation particulier, au démarrage du système on a $Tslot(i) = Tslot_init(i)$.
 - *rx_latency* est le temps de propagation des données à travers les différents blocs du module de communication synchrone (SCM) en charge de l'émission et de la réception des paquets
- écriture dans les registres de surveillance 404 de l'écart « Diff » et des paramètres du paquet reçu, à savoir : les signaux *Rx_hop_id*, *Rx_mode*, et *Rx_hop_cpt*;
 - transmission à l'unité centrale 87 d'un signal d'interruption. En d'autres termes, on notifie la réception du paquet à l'unité centrale 87 via le signal *Rx* 401 ;
 - mise à jour de la valeur du signal *Tx_hop_cpt* 414 qui sera utilisé pour le prochain cycle SDTC. On note que s'il s'agit du premier paquet reçu, alors on incrémente d'une unité la valeur *Rx_hop_cpt* (qui devient la première valeur temporaire), sinon on compare la valeur temporaire (préalablement déterminé) avec la valeur *Rx_hop_cpt*. Si la valeur *Rx_hop_cpt* est strictement inférieure à la valeur temporaire, alors on met à jour la valeur du signal *Tx_hop_cpt* 414 avec la valeur *Rx_hop_cpt* incrémentée d'une unité (en d'autres termes, $Tx_hop_cpt = Rx_hop_cpt + 1$).
- Dans l'étape 715, on vérifie si l'intervalle de temps courant est le dernier intervalle de temps du cycle SDTC. Pour ce faire, on analyse le signal « Slot_number » 1150 généré en sortie du dispositif 7000 (cf. figure 11d). En cas de vérification positive, on passe à une étape 716, sinon on revient à l'étape 708.

Dans l'étape 716, le contrôleur SDTC 97 réalise les opérations suivantes :

- transmission d'une requête de lecture (via le signal Req 402) vers l'unité centrale 87 ; attente d'une réponse de l'unité centrale (via le signal Ack 403) ; puis lecture dans les registres de correction 406 des paramètres de correction du cycle SDTC (à savoir, la valeur du décalage « shift » et la valeur de la dérive « drift ») et du mode d'émission (« Tx_mode ») pour le prochain cycle SDTC ;

5

- calcul de la nouvelle durée du cycle SDTC « SDTC_duration ». Dans un mode de réalisation particulier, au démarrage du système « SDTC_duration » est égale à T_{sdtc_init} la valeur initiale stockée dans les registres de configuration 405. la nouvelle durée « SDTC_duration » est définie par :

10

$$SDTC_duration = SDTC_duration + shift + drift$$

On en déduit les nouvelles durées des intervalles de temps $Tslot(k)$:

$$Tslot = \frac{SDTC_duration - shift - GI}{Nb_slot_per_sdtc}$$

où

GI représente l'intervalle de garde qui couvre une éventuelle variation de la durée du cycle SDTC dans le nœud maître.

15

Du fait que l'on ne gère que des nombres entiers, il convient de prendre la partie entière du résultat de l'équation ci-dessus et de répartir le reste « r » sur quelques intervalles de temps. On a donc « r » intervalles de temps qui ont une durée égale à $Int(Tslot)$, en revanche les autres intervalles de temps ont une durée égale à $Int(Tslot) + 1$.

20

On note également que la durée du premier intervalle de temps doit être modulée par la valeur du décalage « shift ». On a donc : $Tslot(0) = Tslot(0) + shift$.

On note par ailleurs que la durée du premier intervalle de temps du nœud maître doit être étendue de la valeur de l'intervalle de garde « GI ». On a donc : $Tslot(master) = Tslot(master) + GI$.

25

Dans une étape 717, le contrôleur SDTC 97 attend le début du prochain cycle SDTC.

Dans une étape 718, on détecte l'activation du signal SDTC_start 410 indiquant un début de cycle de transmission SDTC. Dans cette étape 718, le contrôleur SDTC 97 réinitialise et redémarre le compteur de cycles SDTC 4000 de la figure 11a avec

$Time_before_sdtc = SDTC_duration$. Dans cette même étape 718, le contrôleur SDTC 97 réinitialise et redémarre également le dispositif 7000 de la figure 11d avec $slot_duration = Tslot(0)$. En outre, on note que le signal SDTC_start 410 est envoyé au contrôleur SDPC 96 en même temps que le signal SDTC_duration 415 (indiquant la nouvelle durée du cycle SDTC). Ensuite, on revient à l'étape 708.

6.8 Fonctionnement de la tâche CPU dédiée à la synchronisation

On présente, en relation avec la **figure 8**, les étapes principales d'un algorithme de traitement mis en œuvre par l'unité centrale 87 (décrit en relation avec les figures 4a et 4b), compris dans un nœud esclave, selon un mode de réalisation particulier de l'invention.

Après démarrage du nœud esclave (étape 800), l'unité centrale 87 initialise dans une étape 801 les registres de configuration 405 et la matrice de répartition 407 avec les données stockées dans le bloc de mémoire non volatile 85 (ROM).

Dans une étape 802, l'unité centrale 87 attend l'un des deux événements suivants :

- Rx 401 : signal permettant de notifier la réception d'un paquet de données et l'écriture de nouvelles données dans les registres de surveillance 404 ;
- Req 402 : signal permettant de notifier la fin du cycle SDTC.

6.8.1 Réception d'un événement « Rx »

Dans une étape 803, on reçoit le signal Rx 401 (émis par le contrôleur SDTC 97).

Dans une étape 804, l'unité centrale 87 réalise les opérations suivantes :

- lecture dans les registres de surveillance 404 des paramètres du paquet reçu ;
- mise à jour du niveau de visibilité pour l'intervalle de temps correspondant à la valeur Rx_hop_id. Le niveau de visibilité est égal au nombre de réceptions de paquet pour l'intervalle de temps considéré divisé par le nombre de cycles SDTC écoulé(s) depuis le démarrage du système. Le nombre de réceptions de paquet correspond au nombre de fois où le signal Rx 401 a été activé pour l'intervalle de temps considéré ;
- lissage de l'écart mesuré pour l'intervalle de temps considéré. En d'autres termes, l'écart mesuré est pondéré pour tenir compte de l'incertitude sur la

mesure et de(s) l'éventuelle(s) variation(s) brusque(s) des oscillateurs. Dans un exemple de réalisation, l'écart mesuré est ajusté d'un coup (ou tick) d'horloge pour tenir compte d'une incertitude de mesure qui typiquement serait de plus ou moins un coup (ou tick) d'horloge. On note que l'écart mesuré peut être comparé à la moyenne des écarts, calculée en tenant compte des N dernières mesures pour l'intervalle de temps considéré. Par ailleurs, on note que si la valeur mesurée s'écarte largement de la valeur moyenne, alors soit on prend en compte cette valeur mesurée dans le calcul de correction en prenant, par exemple, la médiane entre la valeur mesurée et la valeur moyenne, soit on ne la prend pas en compte dans le calcul de correction. Ensuite, on revient à l'étape 802.

6.8.2 Réception d'un événement « Req »

Dans une étape 810, on reçoit le signal Req 402 (émis par le contrôleur SDTC 97).

Dans une étape 811, l'unité centrale 87 réalise les opérations suivantes :

- 15 - filtrage des mesures. Plus précisément, les différentes mesures d'écart sont filtrées en ne gardant que les mesures obtenues avec Rx_hop_cpt égal à « 0 » ou « 1 », et avec un niveau de visibilité supérieur à un seuil prédéterminé (par exemple 0,5) ;
- 20 - calcul des paramètres de correction (valeurs de décalage « shift » et de dérive « drift ») du cycle SDTC :
 - si on reçoit un paquet émis par le nœud maître, alors on utilise uniquement l'écart obtenu pour le ou les intervalles de temps correspondants. Les valeurs de décalage « shift » et de dérive « drift » sont donc égales à l'écart obtenu (ou à la moyenne des écarts si plusieurs intervalles de temps sont affectés au nœud émetteur, comme c'est le cas pour le nœud maître dans l'exemple de la figure 2) ;
 - 25 ○ si on reçoit un paquet émis par un nœud distant avec Rx_hop_cpt égal à « 1 » (donc distinct du nœud maître) et qu'aucun paquet en provenance du nœud maître n'est reçu, alors on utilise l'écart obtenu pour l'intervalle de temps correspondant. Les valeurs de décalage « shift » et de dérive
 - 30 « drift » sont donc égales à l'écart obtenu ;

- si on reçoit plusieurs paquets émis par plusieurs nœuds distants avec Rx_hop_cpt égal à « 1 » (donc distincts du nœud maître) et qu'aucun paquet en provenance du nœud maître n'est reçu, alors on met en œuvre une analyse statistique pour déduire les paramètres de correction de la durée SDTC. Pour ce faire, on obtient à partir des écarts obtenus une courbe de régression linéaire du type $y = \alpha \times x + \beta$, en utilisant la méthode des moindres carrés. Si on note « y_i » l'écart obtenu pour l'intervalle de temps numéro « x_i », et si on suppose une incertitude de mesure identique pour chaque valeur, alors les coefficients α, β sont définis par :

$$\alpha \sum x_i^2 + \beta \sum x_i = \sum x_i y_i$$

$$\alpha \sum x_i + \beta = \sum y_i$$

Le décalage « shift » correspond à la valeur de $\alpha \times x + \beta$ pour le premier intervalle de temps du prochain cycle SDTC ($x=10$, si on a 10 intervalles de temps par cycle SDTC numérotés de 0 à 9). Le décalage « shift » est exprimé en nombre de coups (ou ticks) d'horloge. Pour ce paramètre « shift », on prend le nombre entier le plus proche de la valeur obtenue.

La dérive « drift » correspond à la différence entre la valeur de $\alpha \times x + \beta$ pour le premier intervalle de temps du prochain cycle SDTC ($x=10$) et la valeur de $\alpha \times x + \beta$ pour le premier intervalle de temps du cycle SDTC courant ($x=0$). La dérive « drift » est exprimée en nombre de coups (ou ticks) d'horloge. Pour ce paramètre « drift », on prend le nombre entier le plus proche de la valeur obtenue.

Il est important de noter que pour éviter d'effectuer une correction de grande amplitude, notamment sur les premiers cycles SDTC, les valeurs de décalage « shift » et de dérive « drift » (transmises au contrôleur SDTC) sont bornées.

La **figure 9** présente un exemple de courbe de régression 900. Sur cette figure, on représente en abscisse les numéros des intervalles de temps, et en ordonnée le

nombre de coups (ou ticks) d'horloge. Dans le présent exemple, on se place au niveau du module de communication synchrone 2 (SCM#6) du nœud 2a. On suppose également que le module de communication synchrone 2 (SCM#6) du nœud 2a n'a reçu aucun paquet du nœud maître (SCM#0) et que les paquets qu'il a reçus avec Rx_hop_id égal à

5 « 1 » sont les paquets des nœuds 4a (SCM#2), 3a (SCM#4), 8a (SCM#5) et 1a (SCM#7). Sur la figure 9, les écarts obtenus sont matérialisés par les repères 901, 902, 903, 904. La courbe de régression 900 est définie par l'équation : $y = 0.0769 \times x - 0.4231$. On obtient donc les valeurs réelles de décalage et de dérive suivantes : $shift = 0.3462$ et $drift = 0.7692$. Comme indiqué précédemment, pour

10 chaque paramètre, on prend le nombre entier le plus proche de la valeur réelle obtenue. Ainsi, on transmet au contrôleur SDTC 97 un paramètre de décalage ayant une valeur égale à « 0 » ($shift = 0$) et un paramètre de dérive ayant une valeur égale à « 1 » ($drift = 1$). La durée du prochain cycle SDTC sera donc allongée d'un coup (ou tick) d'horloge par rapport au cycle SDTC courant.

15 Dans une étape 812, on vérifie si la transmission est autorisée. En d'autres termes, on vérifie si un calcul de correction a été effectué à l'étape 811 (c'est-à-dire si on dispose de mesure avec Rx_hop_cpt égal à « 0 » ou « 1 »). En cas de vérification positive, on passe à une étape 813, sinon on passe à une étape 816.

On note que dans d'autres variantes de réalisation, il est possible d'envisager de :

- 20 - utiliser la valeur du signal Rx_mode pour donner un poids plus ou moins important à la mesure associée. Par exemple, la mesure sera considérée comme plus fiable si Rx_mode est égal au mode « fin » ;
- prendre en compte les mesures associées à une valeur Rx_hop_cpt supérieure à « 1 » pour calculer les paramètres de correction ;
- 25 - autoriser l'émission de paquet en mode « grossier » malgré un masquage total pendant un ou plusieurs cycles SDTC.

Dans l'étape 813, l'unité centrale 87 vérifie la stabilité de la synchronisation. En cas de vérification positive (état stable), on passe à une étape 814, sinon (état instable) on passe à une étape 815. Dans un mode de réalisation particulier, une stabilité de

30 synchronisation est détectée lorsqu'une correction de durée SDTC ($shit + drift$) reste

inférieure ou égale à un coup (ou tick) d'horloge pendant un nombre prédéterminé de cycles SDTC (par exemple 10).

Dans l'étape 814, on sélectionne le mode « fin » comme mode d'émission. Ensuite, on passe à l'étape 817.

5 Dans l'étape 815, on sélectionne le mode « grossier » comme mode d'émission. Ensuite, on passe à l'étape 817.

Dans l'étape 816, on considère que le nœud n'est plus synchronisé et on interdit la transmission de paquet, en sélectionnant un mode de non-transmission (i.e le mode d'émission prend la valeur « none »). Ensuite, on passe à l'étape 817.

10 Dans l'étape 817, l'unité centrale 87 écrit dans les registres de corrections 406 les paramètres de correction calculés et le mode d'émission (« Tx_mode ») sélectionné. Ces paramètres sont ensuite mis à disposition du contrôleur SDTC 97. Enfin, on revient à l'étape 802.

6.10 Fonctionnement du bloc de calcul de début de durée de cycle SDPC

15 La **figure 10** présente les étapes principales d'un algorithme de traitement mis en œuvre par le contrôleur SDPC 96 (décrit en relation avec les figures 4a et 4b), compris dans un nœud esclave, selon un mode de réalisation particulier de l'invention.

Après démarrage du nœud esclave (étape 1000), le contrôleur SDPC 96 attend dans une étape 1001 l'un des deux événements suivants :

- 20
- SDPC_start : signal permettant de notifier le début d'un nouveau cycle SDPC ;
 - SDTC_start 410 : signal permettant de notifier le début d'un nouveau cycle de transmission.

6.10.1 Réception d'un événement « SDTC_start »

25 Dans une étape 1020, on reçoit le signal SDTC_start 410 (indiquant un début de cycle SDTC).

Dans une étape 1021, le contrôleur SDPC 96 effectue les opérations suivantes :

- 30
- obtention du signal SDTC_duration 415 et lecture de la nouvelle durée de cycle SDTC ;
 - calcul, en nombre de coups (ou ticks) d'horloge locale, de la nouvelle durée « SDPC_duration » des cycles SDPC restants avant la fin du cycle SDTC. La nouvelle durée « SDPC_duration » est définie par :

$$SDPC_duration = \frac{SDTC_duration}{STPR}$$

où *STPR* désigne le nombre de cycles SDPC dans un cycle SDTC.

Du fait que l'on ne gère que des nombres entiers, il convient de prendre la partie entière du résultat de l'équation ci-dessus et de répartir le reste « r » sur quelques intervalles de temps. On a donc « r » intervalles de temps qui ont une durée égale à $\text{Int}(\text{SDPC_duration})+1$, en revanche les autres intervalles de temps ont une durée égale à $\text{Int}(\text{SDPC_duration})$;

5

- initialisation et démarrage d'un compteur de cycles SDPC 6000. Ensuite, on revient à l'étape 1001.

On présente désormais, en relation avec la **figure 11c**, un exemple d'architecture du compteur de cycles SDPC 6000 selon un mode de réalisation particulier de l'invention. Le compteur de cycles SDPC 6000 génère le signal SDPC_start 1131 indiquant le début d'un nouveau cycle SDPC. Le compteur de cycles SDPC 6000 comprend :

10

- un compteur 1121 qui délivre en sortie « Out » un signal de sortie 1126, représentatif d'une durée exprimée en nombre de coups (ou ticks) d'horloge. Plus précisément, le compteur 1121 reçoit sur une première entrée « Reset » un signal de réinitialisation « Reset3 » 1122, sur une deuxième entrée « Clock ENA » un signal de commande « CENa3 » 1123, et sur une troisième entrée « Clock » un signal d'horloge « Clk » 1124 ;

15

20

- un comparateur 1125 recevant sur une première entrée le signal de sortie 1126 et sur une deuxième entrée le signal « SDPC_duration » 1127 représentatif de la nouvelle durée des cycles SDPC restants avant la fin du cycle SDTC. Le comparateur 1125 délivre en sortie un signal résultant 1129 correspondant au résultat de la soustraction entre le signal de sortie 1126 et le signal « SDPC_duration ». Ainsi, le comparateur 1125 génère en sortie une impulsion lorsque la valeur du compteur 1121 a atteint la valeur « SDPC_duration » ;

25

- une porte logique de type « OU » 1130 recevant sur une première entrée le signal de sortie 1129 et sur une deuxième entrée le signal « SDTC_start » 410 indiquant le début d'un nouveau cycle SDTC. La porte logique 1130 délivre en sortie le signal SDPC_start 1131. On note que le début du premier cycle SDPC correspond au début de cycle SDTC.

30

6.10.2 Réception d'un événement « SDPC_start »

Dans une étape 1010, on reçoit le signal SDPC_start 1131 (généralisé en sortie du compteur de cycles SDPC 6000 décrit en relation avec la figure 11c).

5 Dans une étape 1011, le contrôleur SDPC 96 vérifie s'il s'agit du dernier cycle SDPC du cycle SDTC courant. En cas de vérification positive, on passe à une étape 1012. En revanche, en cas de vérification négative, on passe à une étape 1013.

Dans l'étape 1012, on arrête le compteur de cycles SDPC 6000. Ensuite, on revient à l'étape 1001.

10 Dans l'étape 1013, on met à jour la durée du prochain cycle SDPC à partir du calcul effectué à l'étape 1021, et on redémarre le compteur de cycles SDPC 6000. Ensuite, on revient à l'étape 1001.

REVENDICATIONS

1. Procédé de transmission de données par un nœud émetteur dans un cycle de transmission de données (SDTC) d'un réseau de communication synchrone (2000) comprenant une pluralité de nœuds dont au moins un nœud esclave (1a) et un nœud maître (5a) mettant en œuvre un cadencement qui définit ledit cycle de transmission, ledit cycle de transmission comprenant un nombre prédéterminé d'intervalles de temps, dits temps de parole, chacun desdits temps de parole étant associé à un nœud de ladite pluralité de nœuds, ledit nœud émetteur étant un nœud esclave, ledit nœud émetteur possédant un premier mode d'émission associé à un premier intervalle de garde s'inscrivant dans un desdits temps de parole pendant lequel ledit nœud émetteur ne transmet pas de donnée, caractérisé en ce que le nœud émetteur effectue les étapes suivantes :
- obtention (612) d'un début effectif de temps de parole et d'un début estimé de temps de parole pour un nœud, dit nœud de référence, de ladite pluralité de nœuds, ledit nœud de référence étant distinct du nœud émetteur ;
 - à partir desdits débuts estimé et effectif, détermination (811) d'une correction à apporter à une représentation locale au nœud émetteur dudit cadencement défini par le nœud maître ;
 - sélection (813, 814, 815) d'un mode d'émission courant, parmi ledit premier mode d'émission et un second mode d'émission parmi un ensemble de modes d'émission, en fonction de ladite correction, chaque mode dudit ensemble étant associé à un intervalle de garde pendant lequel ledit nœud émetteur ne transmet pas de donnée distinct dudit premier intervalle de garde.
2. Procédé selon la revendication 1, caractérisé en ce que la durée dudit premier intervalle de garde est supérieure à la durée de l'intervalle de garde dudit second mode.
3. Procédé selon l'une quelconque des revendications 1 à 2, caractérisé en ce qu'il comprend en outre les étapes suivantes :
- détermination, en fonction dudit mode d'émission courant, d'un ensemble de données à insérer dans un paquet de données à émettre pendant ledit temps de parole associé audit nœud émetteur ;
 - transmission dudit paquet de données selon ledit mode d'émission courant après

insertion dudit ensemble de données déterminé dans ledit paquet de données.

4. Procédé selon la revendication 3, caractérisé en ce que ledit ensemble de données déterminé comprend une information de synchronisation relative audit mode d'émission courant.
- 5 5. Procédé selon la revendication 3, caractérisé en ce que ledit ensemble de données déterminé comprend, dans le cas du second mode d'émission, des données de répétition correspondant à une retransmission de données émises par un nœud de ladite pluralité de nœuds distinct du nœud émetteur, lesdites données de répétition n'étant pas comprises dans ledit ensemble de données déterminé dans le cas du
10 premier mode d'émission.
6. Procédé selon l'une quelconque des revendications 1 à 5, caractérisé en ce que l'étape d'obtention (612) d'un début estimé de temps de parole comprend les étapes suivantes :
 - réception d'un paquet de données comprenant une information de synchronisation
15 relative à un mode d'émission utilisé par un nœud distinct dudit nœud émetteur pour transmettre ledit paquet ;
 - extraction (613) de ladite information de synchronisation ;
 - calcul dudit début estimé de temps de parole, en tenant compte de ladite information de synchronisation extraite.
- 20 7. Procédé selon l'une quelconque des revendications 1 à 6, caractérisé en ce que ladite étape de détermination (811) d'une correction comprend les étapes suivantes :
 - détermination d'un décalage (shift) entre le cadencement défini par le nœud maître et la représentation locale dudit cadencement, à partir desdits débuts estimé et effectif ;
 - détermination d'une dérive (drift) entre le cadencement défini par le nœud maître et
25 la représentation locale dudit cadencement, à partir desdits débuts estimé et effectif ;
 - détermination de ladite correction à partir dudit décalage et de ladite dérive.
8. Procédé selon la revendication 7, caractérisé en ce que ladite étape de détermination (811) d'une correction comprend une étape de détermination d'une courbe de régression linéaire à partir desdits débuts estimé et effectif, et en ce que ledit
30 décalage et ladite dérive sont évalués à partir de ladite courbe de régression linéaire.

- 9.** Procédé selon l'une quelconque des revendications 1 à 8, caractérisé en ce que ladite étape de détermination (811) d'une correction comprend en outre une étape de sélection de nœud, comprenant elle-même les étapes suivantes, pour chaque nœud de ladite pluralité de nœuds :
- 5 - détermination (804) d'un niveau de visibilité dudit nœud par rapport au nœud maître ;
- comparaison dudit niveau de visibilité à un seuil déterminé ;
- si ledit niveau de visibilité est supérieur audit seuil déterminé, sélection dudit nœud pour la détermination de ladite correction.
- 10 **10.** Procédé selon l'une quelconque des revendications 1 à 9, caractérisé en ce que le nœud émetteur sélectionne (814) le second mode d'émission lorsque ladite correction déterminée est stable sur un nombre prédéterminé de cycles de transmission, et en ce que le nœud émetteur sélectionne (815) le premier mode d'émission lorsque ladite correction déterminée est instable.
- 15 **11.** Produit programme d'ordinateur téléchargeable depuis un réseau de communication et/ou enregistré sur un support lisible par ordinateur et/ou exécutable par un processeur, caractérisé en ce qu'il comprend des instructions de code de programme pour l'exécution des étapes du procédé de synchronisation selon au moins une des revendications 1 à 10, lorsque ledit programme est exécuté sur un ordinateur.
- 20 **12.** Moyen de stockage lisible par un ordinateur, stockant un jeu d'instructions exécutables par ledit ordinateur pour mettre en œuvre le procédé de synchronisation selon au moins une des revendications 1 à 10.
- 13.** Nœud émetteur comprenant des moyens de transmission de données dans un cycle de transmission de données d'un réseau de communication synchrone comprenant
- 25 une pluralité de nœuds dont au moins un nœud esclave et un nœud maître mettant en œuvre un cadencement qui définit ledit cycle de transmission, ledit cycle de transmission comprenant un nombre prédéterminé d'intervalles de temps, dits temps de parole, chacun desdits temps de parole étant associé à un nœud de ladite pluralité de nœuds, ledit nœud émetteur étant un nœud esclave, ledit nœud émetteur possédant un

premier mode d'émission associé à un premier intervalle de garde s'inscrivant dans un desdits temps de parole pendant lequel ledit nœud émetteur ne transmet pas de donnée, caractérisé en ce que le nœud émetteur comprend :

- 5 - des moyens d'obtention d'un début effectif de temps de parole et d'un début estimé de temps de parole pour un nœud, dit nœud de référence, de ladite pluralité de nœuds, ledit nœud de référence étant distinct du nœud émetteur ;
- des moyens de détermination d'une correction à apporter à une représentation locale au nœud émetteur dudit cadencement défini par le nœud maître ;
- 10 - des moyens de sélection d'un mode d'émission courant, parmi ledit premier mode d'émission et un second mode d'émission parmi un ensemble de modes d'émission, en fonction de ladite correction, chaque mode dudit ensemble étant associé à un intervalle de garde pendant lequel ledit nœud émetteur ne transmet pas de donnée distinct dudit premier intervalle de garde.

14. Nœud émetteur selon la revendication 13, caractérisé en ce que la durée dudit premier intervalle de garde est supérieure à la durée de l'intervalle de garde dudit second mode.

15. Nœud émetteur selon l'une quelconque des revendications 13 à 14, caractérisé en ce que le nœud émetteur comprend en outre :

- 20 - des moyens de détermination d'un ensemble de données à insérer dans un paquet de données à émettre pendant ledit temps de parole associé audit nœud émetteur ;
- des moyens de transmission dudit paquet de données comprenant ledit ensemble de données, selon ledit mode d'émission courant.

16. Nœud émetteur selon la revendication 15, caractérisé en ce que ledit ensemble de données comprend une information de synchronisation relative audit mode d'émission courant.

17. Nœud émetteur selon la revendication 15, caractérisé en ce que ledit ensemble de données comprend, dans le cas du second mode d'émission, des données de répétition correspondant à une retransmission de données émises par un nœud de ladite pluralité de nœuds distinct du nœud émetteur, lesdites données de répétition n'étant pas comprises dans ledit ensemble de données dans le cas du premier mode d'émission.

18. Nœud émetteur selon l'une quelconque des revendications 13 à 17, caractérisé

en ce que lesdits moyens d'obtention d'un début estimé de temps de parole comprennent :

- des moyens de réception d'un paquet de données comprenant une information de synchronisation relative à un mode d'émission utilisé par un nœud distinct dudit nœud émetteur pour transmettre ledit paquet ;
- des moyens d'extraction de ladite information de synchronisation ;
- des moyens de calcul dudit début estimé de temps de parole, lesdits moyens de calcul tenant compte de ladite information de synchronisation.

19. Nœud émetteur selon l'une quelconque des revendications 13 à 18, caractérisé en ce que lesdits moyens de détermination d'une correction comprennent :

- des moyens de détermination d'un décalage entre le cadencement défini par le nœud maître et la représentation locale dudit cadencement, lesdits moyens de détermination déterminant ledit décalage à partir desdits débuts estimé et effectif ;
- des moyens de détermination d'une dérive entre le cadencement défini par le nœud maître et la représentation locale dudit cadencement, lesdits moyens de détermination déterminant ladite dérive à partir desdits débuts estimé et effectif ;
- des moyens de détermination de ladite correction, lesdits moyens de détermination déterminant ladite correction à partir dudit décalage et de ladite dérive.

20. Nœud émetteur selon la revendication 19, caractérisé en ce que lesdits moyens de détermination d'une correction comprennent :

- des moyens de détermination d'une courbe de régression linéaire, lesdits moyens de détermination déterminant ladite courbe de régression linéaire à partir desdits débuts estimé et effectif ;
- des moyens d'évaluation dudit décalage et de ladite dérive utilisant ladite courbe de régression linéaire.

21. Nœud émetteur selon l'une quelconque des revendications 13 à 20, caractérisé en ce que lesdits moyens de détermination d'une correction comprennent en outre des moyens de sélection d'un nœud comprenant eux-mêmes :

- des moyens de détermination d'un niveau de visibilité dudit nœud par rapport au nœud maître ;
- des moyens de comparaison dudit niveau de visibilité à un seuil déterminé.

22. Nœud émetteur selon l'une quelconque des revendications 13 à 21, caractérisé en ce que le nœud émetteur sélectionne le second mode d'émission lorsque ladite correction est stable sur un nombre prédéterminé de cycles de transmission, et en ce que le nœud émetteur sélectionne le premier mode d'émission lorsque ladite correction est instable.

1/12

2000

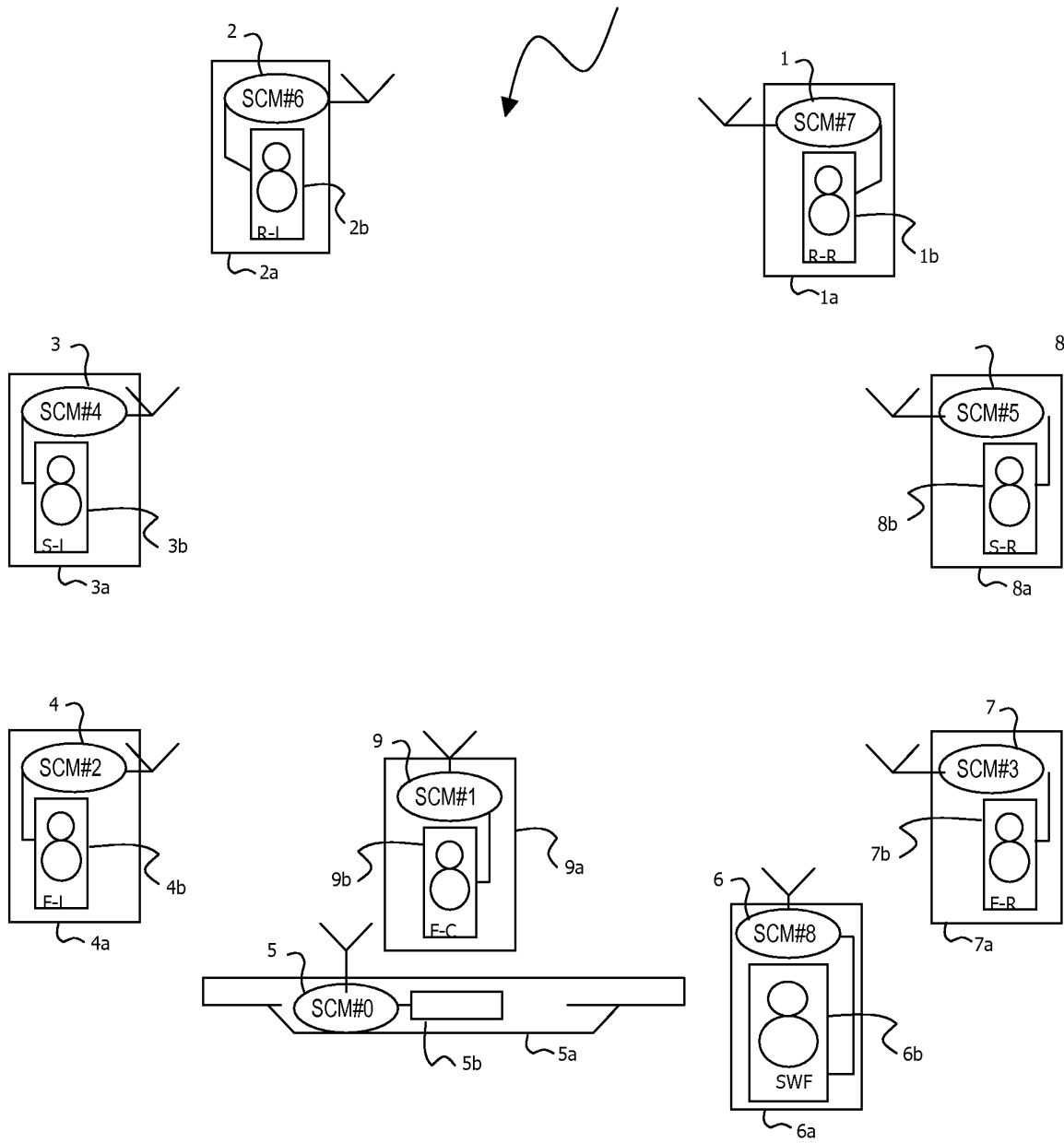


Figure 1

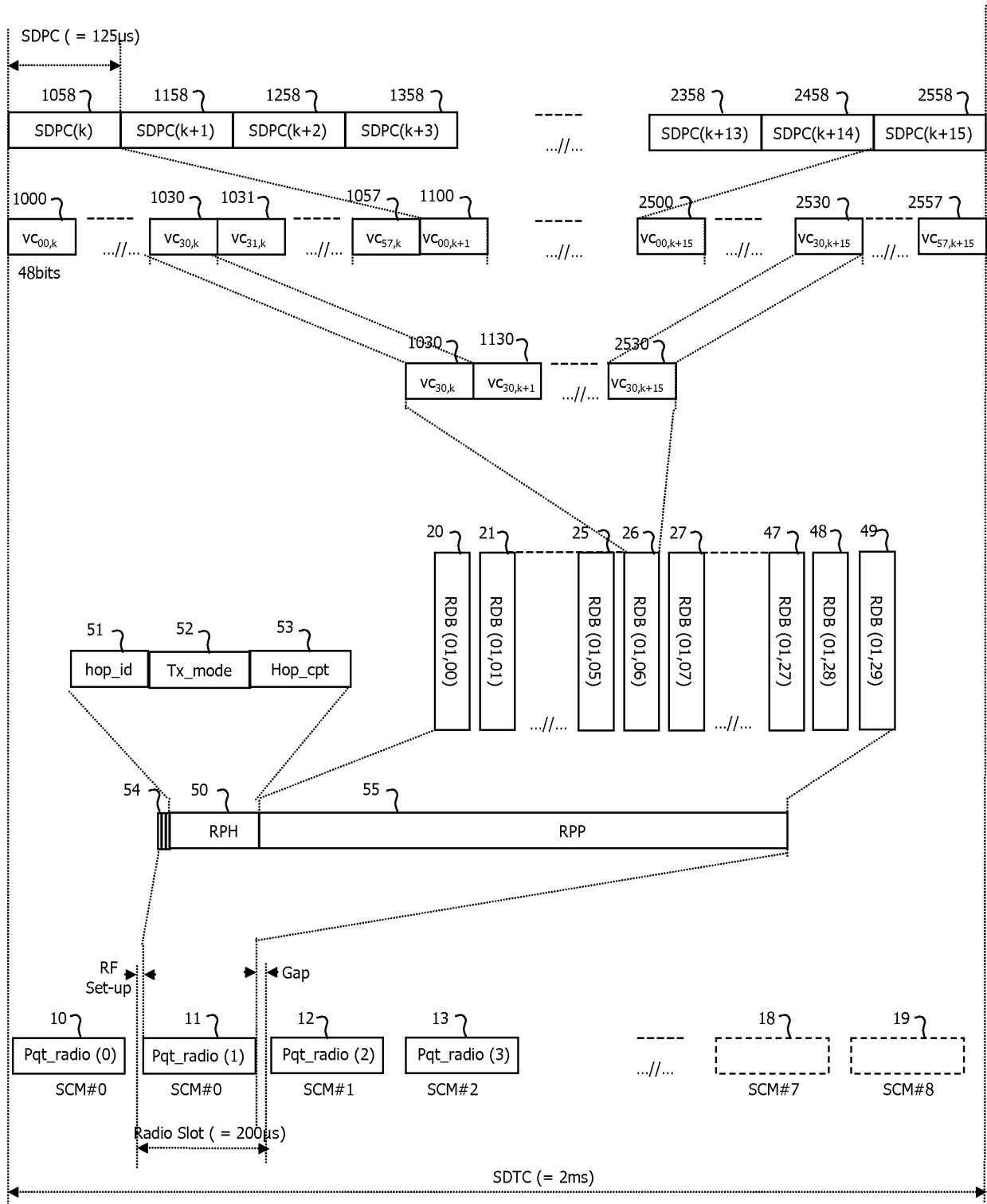


Figure 2

3/12

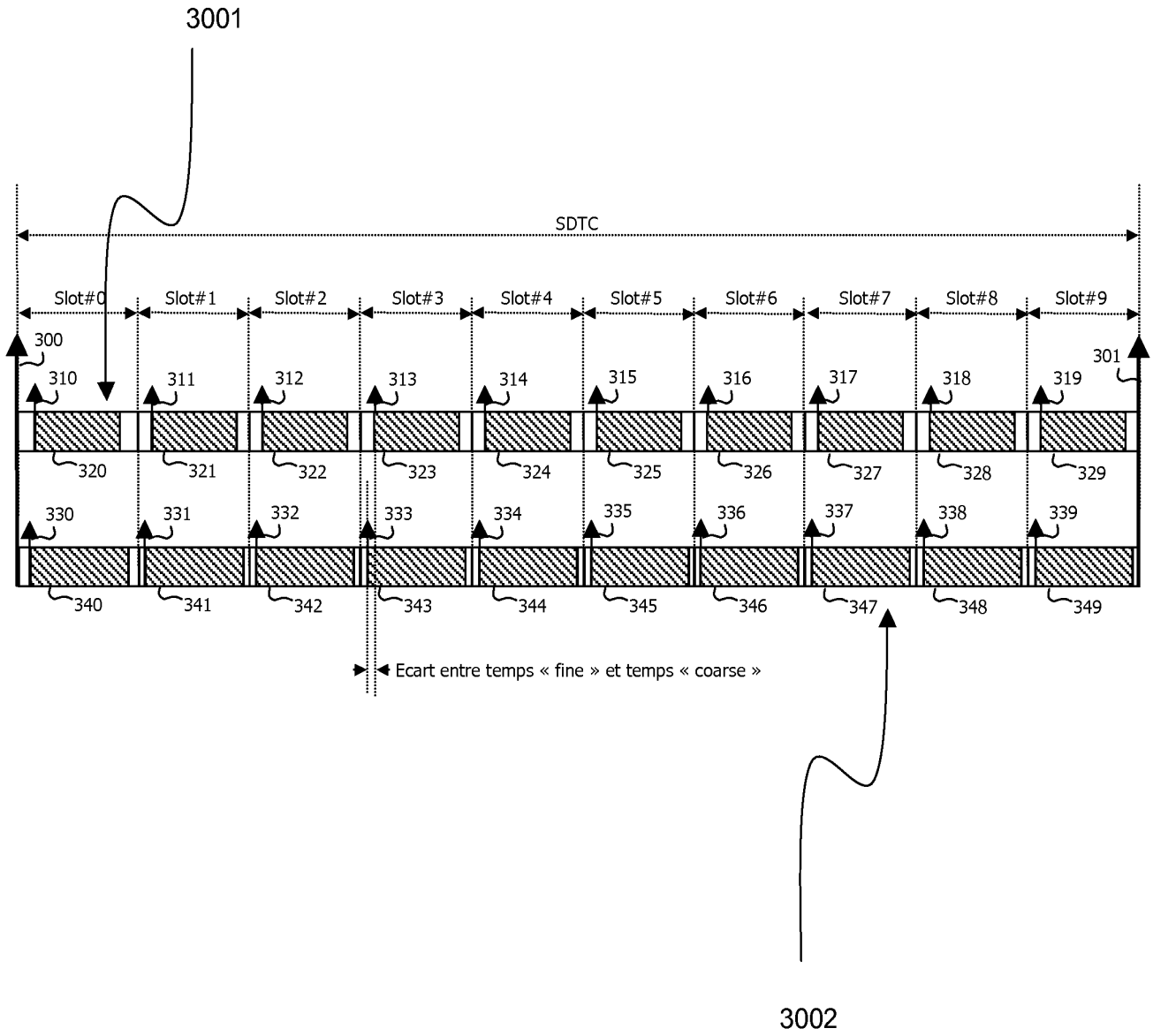


Figure 3

4/12

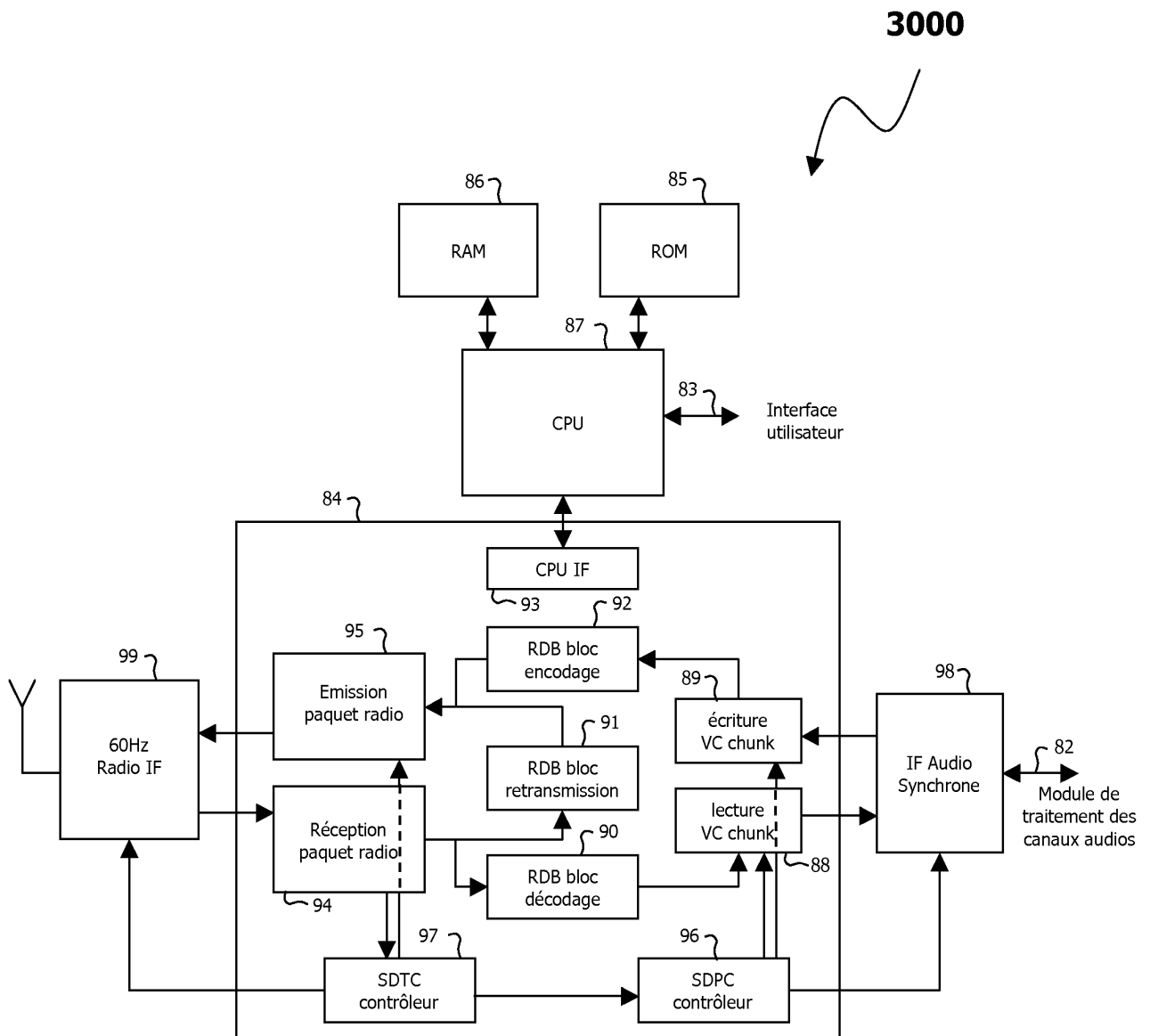


Figure 4a

5/12

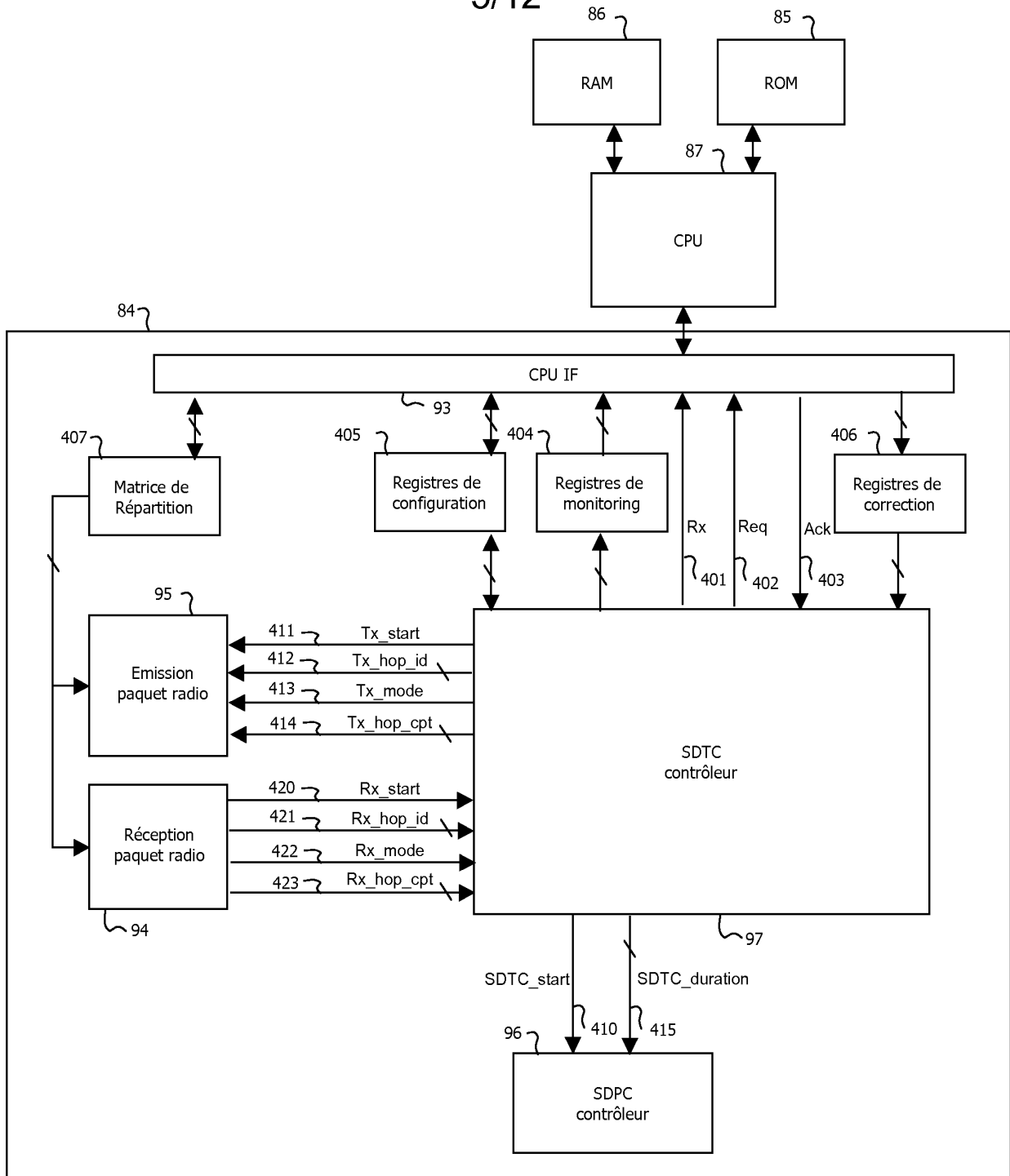


Figure 4b

6/12

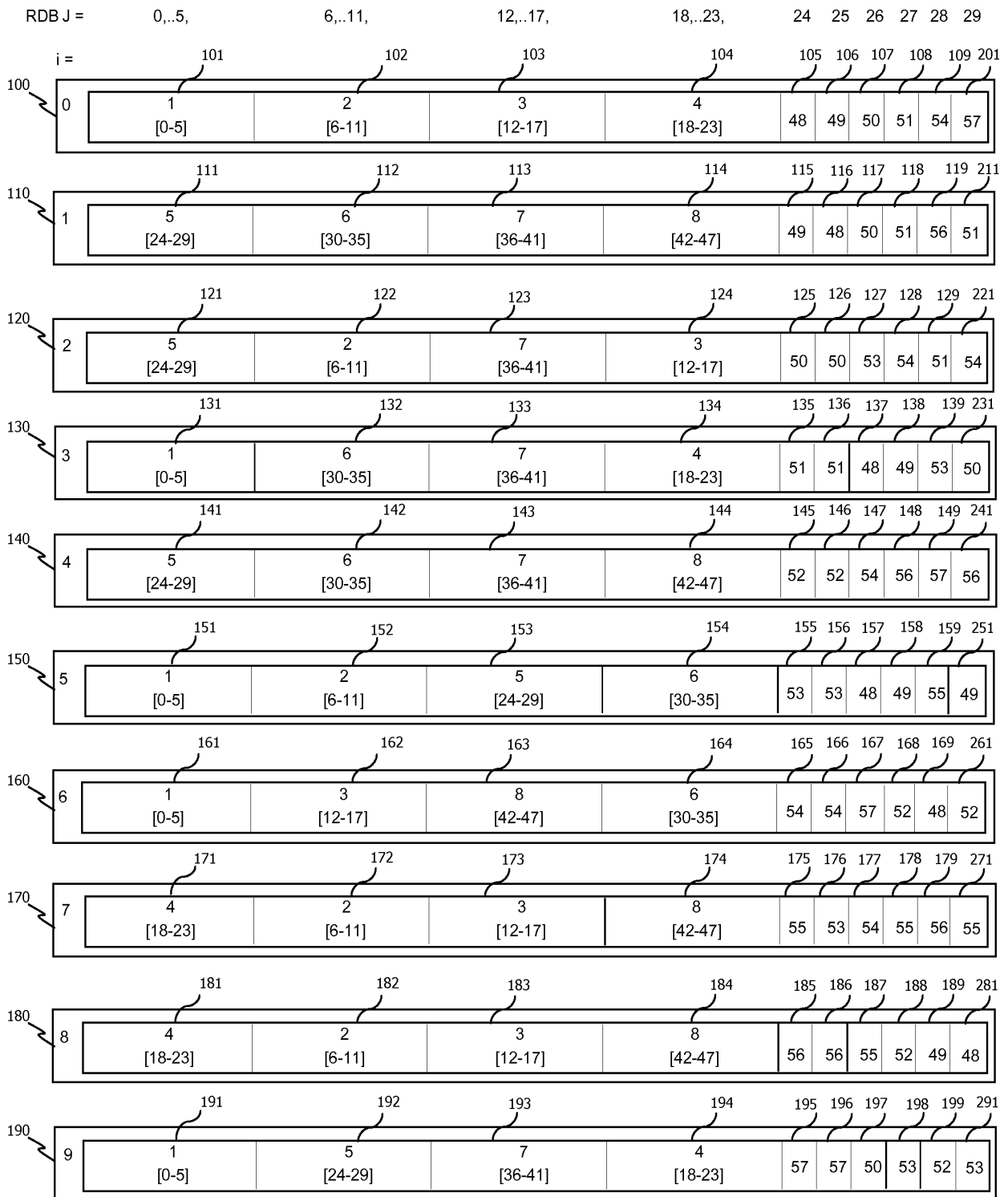


Figure 5

7/12

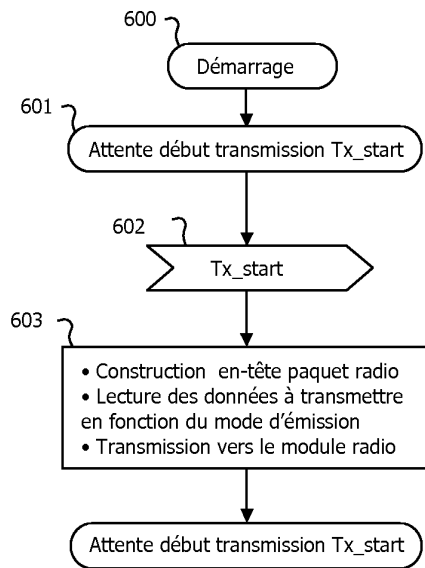


Figure 6a

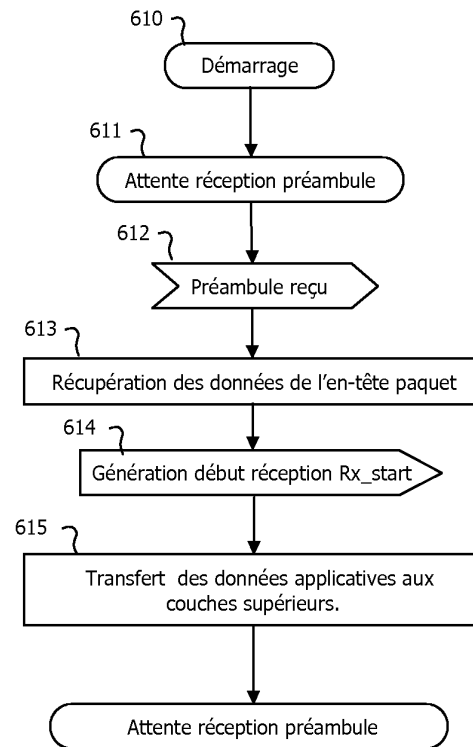


Figure 6b

8/12

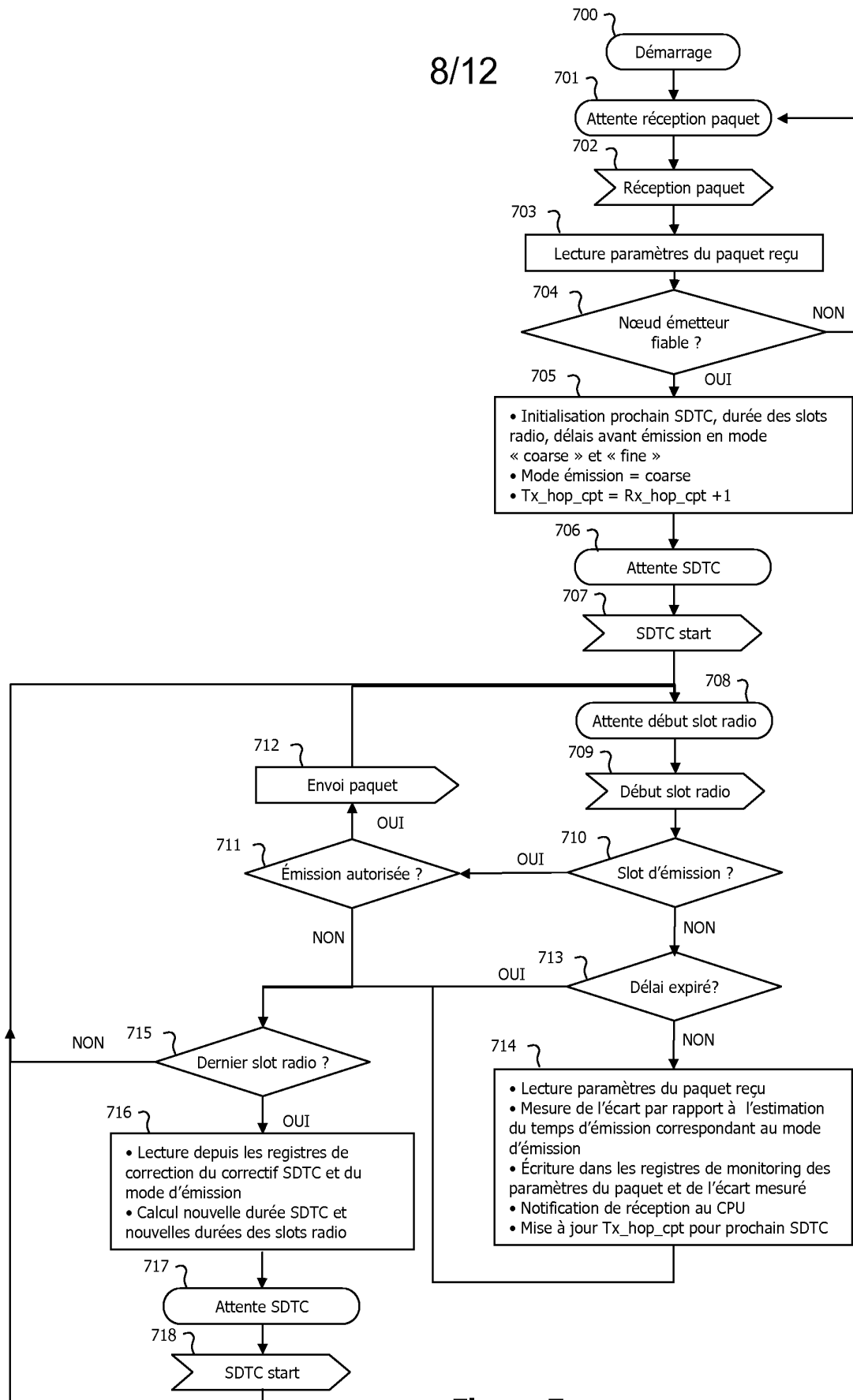


Figure 7

9/12

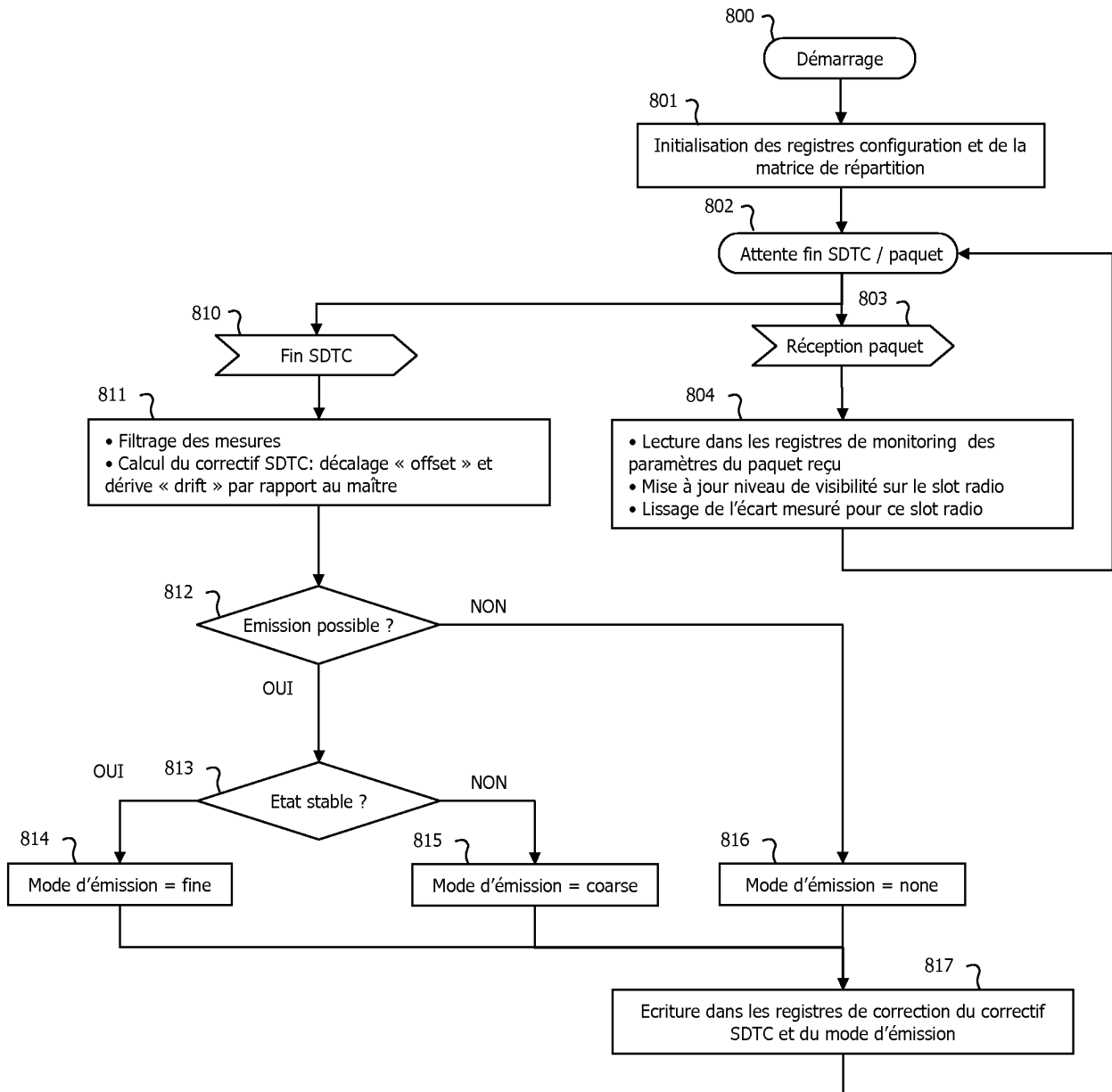
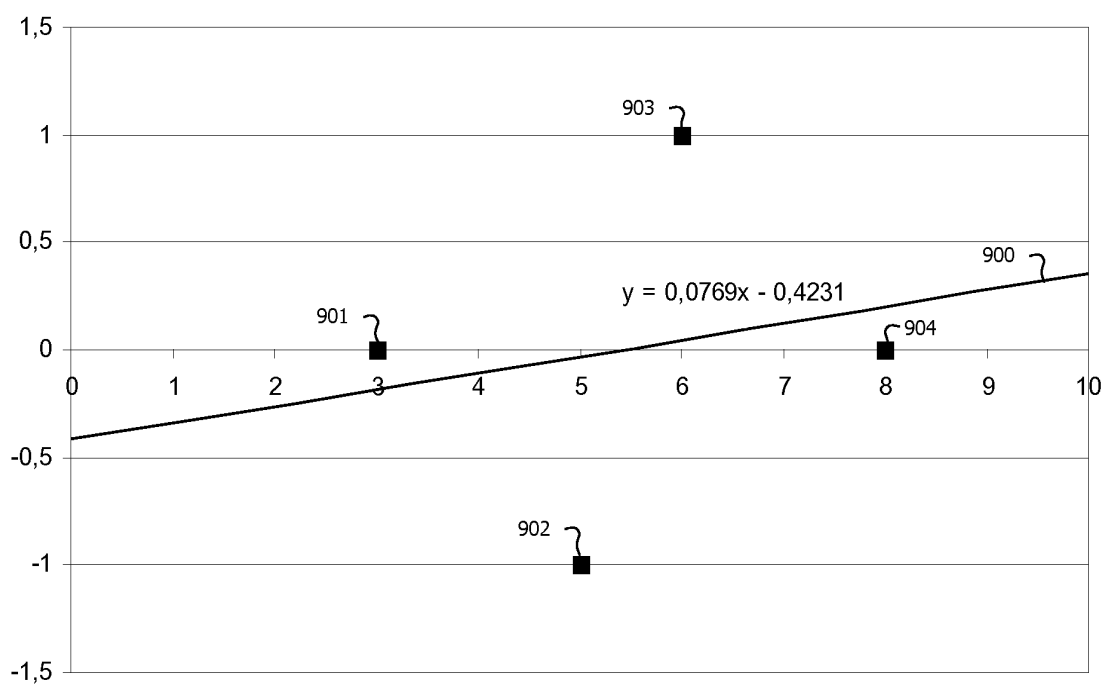


Figure 8

10/12

**Figure 9**

11/12

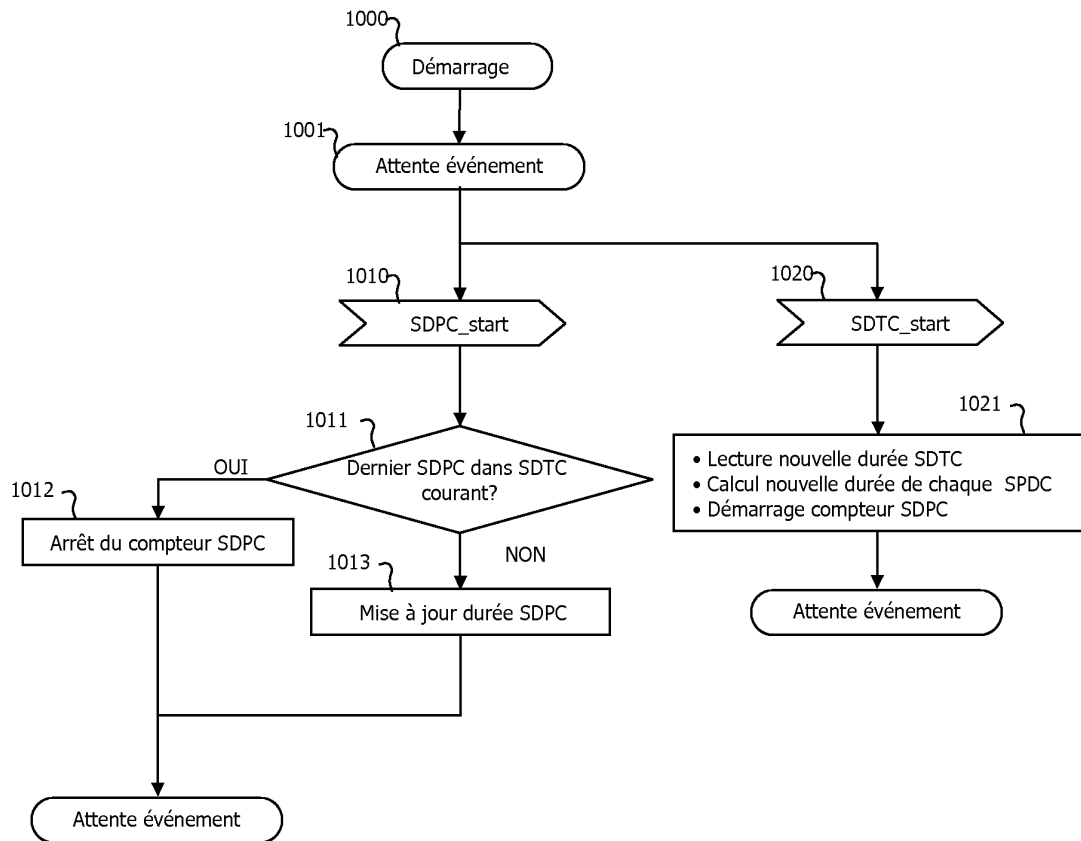


Figure 10

12/12

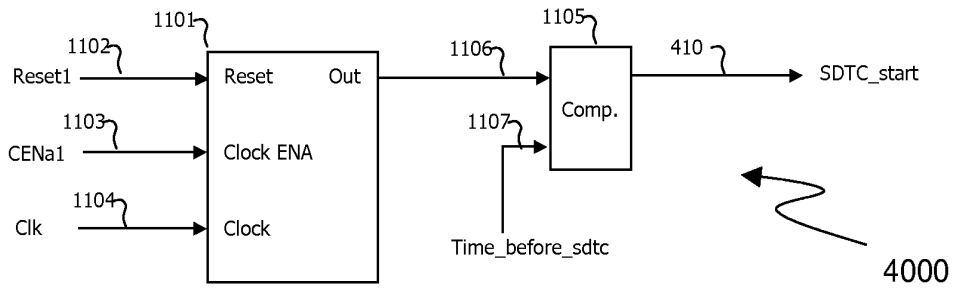


Figure 11a

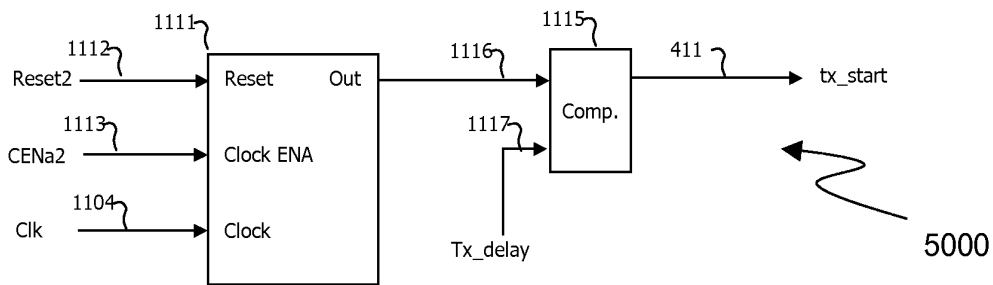


Figure 11b

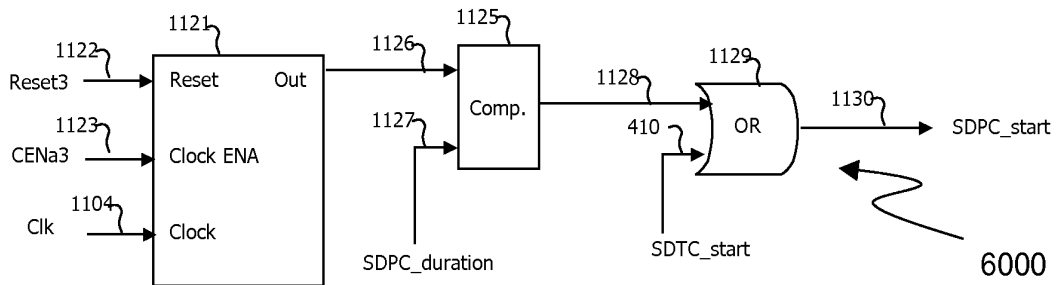


Figure 11c

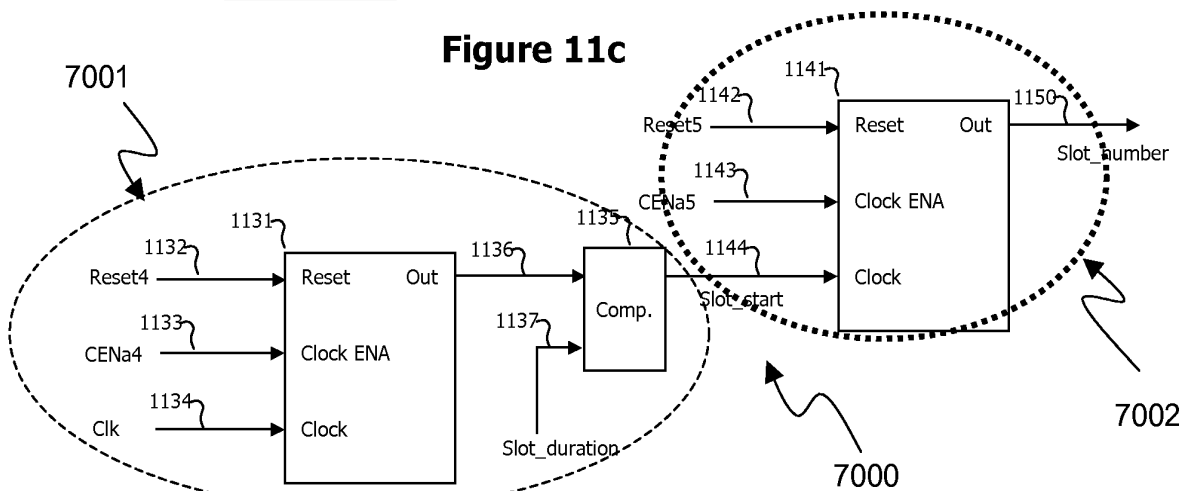


Figure 11d



**RAPPORT DE RECHERCHE
PRÉLIMINAIRE**

N° d'enregistrement
national

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

FA 705937
FR 0850184

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		
X	GB 1 235 711 A (SIERRA RESEARCH CORP [US]) 16 juin 1971 (1971-06-16) * page 1, colonne de gauche, ligne 1 - page 2, colonne de gauche, ligne 18 * * page 3, colonne de gauche, ligne 31 - page 4, colonne de droite, ligne 73 * * page 5, colonne de gauche, ligne 49 - page 8, colonne de droite, ligne 72 * * figures 1-5 *	1,2, 12-14	H04L12/56
A	WO 2004/075446 A (ZARLINK SEMICONDUCTOR INC [CA]; REPKO WILLEM L [NL]; VAN DER VALK ROBE) 2 septembre 2004 (2004-09-02) * alinéa [0009] * * alinéa [0023] * * alinéa [0028] *	1-22	
A	WO 02/065707 A (BLUESOCKET INC [US]) 22 août 2002 (2002-08-22) * page 17, ligne 9 - page 19, ligne 2 *	1-22	
			DOMAINES TECHNIQUES RECHERCHÉS (IPC)
			H04J H04B H04L
		Date d'achèvement de la recherche	Examineur
		10 octobre 2008	Larcinese, Annamaria
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 0850184 FA 705937**

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 10-10-2008

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
GB 1235711 A	16-06-1971	DE 1798357 A1	20-01-1972
		FR 1586030 A	06-02-1970
		NL 6813613 A	08-04-1969
		SE 336011 B	21-06-1971
		US 3440652 A	22-04-1969

WO 2004075446 A	02-09-2004	EP 1595349 A1	16-11-2005
		JP 2006523046 T	05-10-2006
		KR 20050108357 A	16-11-2005

WO 02065707 A	22-08-2002	AUCUN	
