

19) RÉPUBLIQUE FRANÇAISE
INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE
PARIS

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

2 930 700

21) N° d'enregistrement national : 08 52891

51) Int Cl⁸ : H 04 L 29/12 (2006.01), H 04 L 12/56

12)

DEMANDE DE BREVET D'INVENTION

A1

22) Date de dépôt : 29.04.08.

30) Priorité :

43) Date de mise à la disposition du public de la demande : 30.10.09 Bulletin 09/44.

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) : AIRBUS FRANCE Société par actions simplifiée — FR.

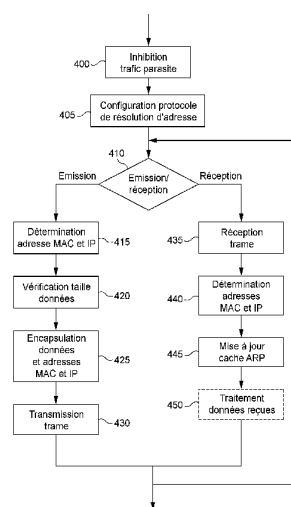
72) Inventeur(s) : DESSERTENNE FRANCK, CANDIA FABRICE et DEPAILLER FABIEN.

73) Titulaire(s) :

74) Mandataire(s) : SANTARELLI.

54) PROCÉDES ET DISPOSITIFS POUR L'ÉCHANGE TEMPS RÉEL DE DONNÉES DANS UN RÉSEAU DE COMMUNICATION COMMUTE.

57) L'invention vise un système pour transmettre en temps réel une donnée applicative d'un premier élément communicant à au moins un second élément communicant, à travers un réseau de communication commuté configuré en mode maître/esclave et utilisant un protocole exempt de connexion. Si le protocole de communication dudit premier élément communicant comprend une fonction adaptée à transmettre une information audit second élément, ladite information étant distincte des données applicatives, ladite fonction est, si possible, inhibée (400). De même, si le protocole de communication dudit premier élément communicant est adapté à mettre en oeuvre une fragmentation des données, ladite fragmentation des données est, si possible, inhibée. La mémoire de résolution d'adresse dudit premier élément communicant est initialisée et configurée (405) pour mémoriser les adresses logiques et physiques dudit au moins un second élément communicant.



FR 2 930 700 - A1



5 La présente invention concerne les transferts de données dans des réseaux de communication et plus particulièrement des procédés et des dispositifs pour l'échange de données, notamment de données de simulation, selon le mode maître/esclave, dans un réseau de communication temps réel utilisant des technologies standard telles qu'un réseau Ethernet commuté et le
10 protocole IP.

 De manière générale, l'invention concerne, dans un réseau de communication comprenant au moins un serveur et au moins un client, les principes régissant la communication de données en temps réel, notamment de données de simulation, entre ces éléments du réseau.

15 L'invention s'applique en particulier dans un réseau de simulation de composants de véhicules, par exemple de composants d'aéronefs, utilisée pour assurer le développement et l'intégration des systèmes électroniques et informatiques embarqués.

 La simulation des comportements des équipements est
20 avantageusement réalisée à la vitesse de leur déroulement réel. Ainsi, l'échange des données de simulation ne doit pas être perturbé par des échanges de données propres au réseau de communication pour simuler au mieux le comportement réel des composants. Une telle simulation est donc basée sur des contraintes temps réel fortes.

25 L'architecture de simulation comprend généralement une pluralité de terminaux aussi appelés nœuds du réseau, chacun de ces terminaux étant apte à réaliser les calculs de simulation ou constituant l'interface électronique avec l'environnement réel, permettant de vérifier le fonctionnement des équipements réels de l'aéronef. Ainsi, cette architecture comprend notamment un terminal de
30 simulation apte à émettre des données selon une séquence synchrone en utilisant le mode requête/réponse.

Plus particulièrement, l'intégration de composants électroniques dans des véhicules fait l'objet de simulations selon lesquelles des dispositifs électroniques d'entrées/sorties, ou cartes d'entrées/sorties, sont utilisés comme interface entre les composants réels du véhicule, tels que, par exemple, des
5 calculateurs, des capteurs et des actionneurs, et un environnement de simulation comprenant généralement un ou plusieurs serveurs ou ordinateurs utilisés pour simuler le comportement du véhicule ou d'une partie de celui-ci. Chaque carte d'entrées/sorties dispose d'un nombre déterminé de voies d'entrée et de voies de sortie.

10 Les mécanismes d'échange de données en temps réel s'appuient généralement sur des solutions matérielles spécifiques, aussi appelées solutions propriétaires. Bien que celles-ci donnent souvent satisfaction pour la simulation, elles comportent plusieurs inconvénients. En particulier, ces solutions étant spécifiques, peu de composants standard du marché sont mis
15 en œuvre et, par conséquent, se posent des problèmes de pérennité, de fournisseurs et de gestion de l'obsolescence du matériel.

Par ailleurs, il existe des réseaux de type « *switch fabric* » basés sur une architecture commutée, c'est-à-dire que les équipements terminaux chargés de l'émission et de la réception des données s'organisent autour des
20 commutateurs chargés du transport de ces données. Le commutateur est chargé d'émettre en parallèle des requêtes en provenance des ordinateurs ou des serveurs à destination des cartes d'entrées/sorties et des réponses en provenance des cartes d'entrées/sorties vers les ordinateurs ou les serveurs. Une même requête et une même réponse doivent pouvoir être adressées par le
25 commutateur à plusieurs destinataires.

L'échange des données entre les différents nœuds d'un tel réseau peut être réalisé sur un port UDP (acronyme de *User Datagram Protocol* en terminologie anglo-saxonne).

30 La topologie d'adressage physique du réseau peut être déterminée, au niveau de chaque nœud du réseau, par l'échange de messages particuliers, typiquement des messages ARP (acronyme d'*Address Resolution Protocol* en terminologie anglo-saxonne).

Cependant, ce type de réseau ne permet généralement pas de garantir le déterminisme des communications, en particulier une latence constante, c'est-à-dire un délai constant entre les instants de transmission et de réception des données, une gigue ou fluctuation minimum, et l'utilisation de la
5 totalité de la bande passante.

La présente invention a pour objet de remédier à au moins un des inconvénients des techniques et processus de l'art antérieur précité. Pour ce faire, l'invention a notamment pour objet un procédé pour transmettre en temps réel au moins une donnée applicative d'un premier élément communicant à au
10 moins un second élément communicant, à travers un réseau de communication commuté configuré en mode maître/esclave et utilisant un protocole exempt de connexion, ce procédé comprenant les étapes suivantes,

- si le protocole de communication dudit premier élément communicant comprend au moins une fonction adaptée à transmettre au moins
15 une information audit second élément, ladite au moins une information étant distincte des données applicatives, et si ladite au moins une fonction peut être inhibée, inhibition de ladite au moins une fonction ;

- si le protocole de communication dudit premier élément communicant est adapté à mettre en œuvre une fragmentation des données et
20 si ladite fragmentation des données peut être inhibée, inhibition de ladite fragmentation des données ; et,

- initialisation et configuration de la mémoire de résolution d'adresse dudit premier élément communicant pour mémoriser les adresses logique et physique dudit au moins un second élément communicant.

25 Le procédé selon l'invention permet ainsi d'utiliser un réseau commuté standard pour échanger des données en temps réel. L'échange de données en temps réel n'est pas perturbé par un trafic parasite.

Par ailleurs, l'utilisation de technologies standard permet de bénéficier des développements existants autour de ces technologies, tels que
30 les outils d'aide à l'analyse et les API de programmation. Elle permet également de profiter de l'expertise de multiples fournisseurs et par conséquent d'être

indépendante de solutions propriétaires. La gestion de l'obsolescence est ainsi facilitée par une offre diversifiée et un marché concurrentiel.

Ladite configuration de ladite mémoire de résolution d'adresse comprend, de préférence, le paramétrage de ladite mémoire de résolution
5 d'adresse pour augmenter sa durée de validité afin de limiter la mise à jour de la mémoire de résolution d'adresse. Si possible, la durée de validité est illimitée.

Selon un mode de réalisation particulier, le procédé comprend en outre les étapes suivantes,

- détermination des adresses logique et physique dudit premier
10 élément communicant ;

- encapsulation desdites adresses logique et physique et de ladite au moins une donnée applicative dans une zone de données utiles d'une trame conforme au protocole de communication dudit réseau de communication commuté ; et,

- transmission de ladite trame audit au moins un second élément
15 communicant.

Le procédé selon l'invention permet ainsi aux éléments communicants connectés au réseau de communication de mettre facilement à jour leur mémoire de résolution d'adresse.

De façon avantageuse, la taille de ladite au moins une donnée applicative est inférieure à une valeur prédéterminée de telle sorte qu'une seule trame conforme au protocole de communication dudit réseau de communication commuté est nécessaire à la transmission de ladite au moins une donnée applicative et desdites adresses logique et physique.

Toujours selon un mode de réalisation particulier, ladite étape d'initialisation de la mémoire de résolution d'adresse dudit premier élément communicant comprend une étape de transmission d'une trame comprenant lesdites adresses logique et physique dudit premier élément communicant dans une zone de données utiles et, en réponse, la réception d'au moins une trame
25 comprenant lesdites adresses logique et physique dudit au moins un second
30 élément communicant dans une zone de données utiles pour permettre la mise

à jour de la mémoire de résolution d'adresse dudit premier élément communicant.

Avantageusement, le procédé comprend en outre une étape de mise à jour périodique de ladite mémoire de résolution d'adresse dudit premier élément communicant à la réception d'une trame reçue dudit au moins un second élément communicant, ladite trame reçue comprenant lesdites adresses logique et physique dudit second élément communicant dans une zone de données utiles. Une telle mise à jour est de préférence effectuée selon la durée de validité de la mémoire de résolution d'adresse pour réduire le nombre de mises à jour nécessaires.

Selon un mode de réalisation particulier, ledit réseau de communication est un réseau de type Ethernet.

De façon avantageuse, un niveau de priorité est associé à la tâche applicative en charge de ladite au moins une donnée applicative, un port de communication étant affecté à ladite tâche applicative, de telle sorte que ledit niveau de priorité est déterminé et géré indépendamment du protocole de transmission dudit réseau de communication pour améliorer l'échange de données en temps réel.

L'invention a également pour objet un programme d'ordinateur comprenant des instructions adaptées à la mise en œuvre de chacune des étapes du procédé décrit précédemment ainsi qu'un dispositif comprenant des moyens adaptés à la mise en œuvre de chacune des étapes de ce procédé.

D'autres avantages, buts et caractéristiques de la présente invention ressortent de la description détaillée qui suit, faite à titre d'exemple non limitatif, au regard des dessins annexés dans lesquels:

- la figure 1 illustre un exemple d'architecture de réseau de communication adapté à simuler des composants d'aéronefs conformément à l'invention ;
- la figure 2 représente un exemple de format de messages adaptés à transmettre des données entre des éléments communicants et à renseigner les mémoires caches de résolution d'adresse de ces éléments ;

- la figure 3 illustre un exemple de séquence de messages transmis lors de la mise en œuvre de l'environnement de simulation représenté sur la figure 1 ;

- la figure 4 illustre certaines étapes d'un exemple d'algorithme pouvant être utilisé, conformément à l'invention, pour transmettre et recevoir des données ; et,

- la figure 5 illustre un exemple d'élément communicant tel que ceux représentés sur la figure 1.

La figure 1 représente un exemple d'architecture de réseau de communication adapté à simuler des composants d'aéronefs conformément à l'invention. Le réseau de communication est ici du type réseau Ethernet commuté, basé sur des technologies standard du marché telles que, par exemple, IEEE 802.3 et IP V4 (IP est le sigle d'*Internet Protocol* en terminologie anglo-saxonne). Le réseau, conforme à l'invention et utilisant le mode maître/esclave, permet d'améliorer le déterminisme du réseau, en particulier la constance de la latence, la gigue et la bande passante réellement utilisée tout en réduisant le transfert de données non directement lié à la simulation.

Comme illustré, le système 100 utilisé pour la simulation de composants d'aéronefs comprend ici un réseau 105, du type Ethernet commuté, comprenant lui-même des commutateurs Ethernet, et des éléments communicants tels que des serveurs et des ordinateurs de type PC (sigle de *Personal Computer* en terminologie anglo-saxonne). L'élément communicant 110 est ici l'élément maître tandis que les éléments communicants 115-1 à 115-3 sont les éléments esclaves.

Les commutateurs Ethernet et les éléments communicants, s'ils ne sont pas configurés de façon particulière, émettent, de façon native, des données sur le réseau, indépendamment des données utilisateur devant être échangées. Ce trafic peut être classé selon les deux catégories suivantes,

- le trafic nécessaire à l'acheminement des données utilisateur tel que le protocole de résolution d'adresse et la gestion de la multi-diffusion, appelée *multicast* en terminologie anglo-saxonne ; et,

- le trafic lié aux protocoles réseaux, normalisés ou propriétaires des constructeurs de commutateurs, pour détecter, par exemple, les boucles sur le réseau.

La seconde catégorie de trafic peut être neutralisée directement sur les commutateurs Ethernet pour rendre disponible la pleine bande passante et, ainsi, minimiser la latence et la gigue sur le réseau. Par exemple, sur certains commutateurs commercialisés par la société CISCO, la désactivation des protocoles générateurs de trafic peut être réalisée à l'aide des commandes suivantes,

10 - "*no spanning-tree vlan 1 – 32*" pour inhiber le protocole appelé STP (sigle de *Spanning Tree Protocol* en terminologie anglo-saxonne), permettant la mise en œuvre d'une topologie réseau sans boucle ;

- "*switchport mode access*" pour inhiber le protocole DTP (sigle de *Dynamic Trunk Protocol* en terminologie anglo-saxonne) permettant de gérer dynamiquement l'activation ou la désactivation de modes particuliers ;

15 - "*no cdp run*" pour inhiber le protocole CDP (sigle de *Cisco Discovery Protocol* en terminologie anglo-saxonne) utilisé pour analyser le voisinage du réseau ; et,

- "*no keepalive*" pour inhiber le protocole appelé *keepalive* en terminologie anglo-saxonne permettant de vérifier l'état d'une interface réseau.

Par ailleurs, l'invention permet de limiter le trafic lié au protocole de résolution d'adresse et d'inhiber les protocoles réseaux.

La solution décrite ici s'appuie sur l'implémentation standard du protocole de résolution d'adresse ARP (RFC 826). Il convient néanmoins de remarquer que l'invention peut être mise en œuvre pour d'autres implémentations.

Selon ce protocole, si l'élément 110, ayant une adresse IP appelée @IP-A et une adresse MAC (acronyme de *Media Access Control* en terminologie anglo-saxonne) appelée @MAC-A, doit envoyer des données à l'élément 115-1, ayant les adresses IP et MAC appelées @IP-B et @MAC-B, respectivement, à travers le réseau 105, il doit connaître les adresses MAC et IP de l'élément 115-1. L'élément 110 analyse donc le contenu de sa mémoire

cache ARP à la recherche de l'adresse IP de l'élément cible, c'est-à-dire de l'élément 115-1, pour trouver l'adresse MAC correspondante.

Il est rappelé ici qu'une mémoire cache ARP, aussi appelée un cache ARP, est un ensemble de couples (adresse logique, adresse physique) contenu dans la mémoire d'un ordinateur utilisant le protocole ARP, c'est-à-dire un espace mémoire dans lequel est enregistrée une table listant des correspondances entre adresses physiques (MAC) et adresses logiques (IP) de nœuds du réseau appartenant au même réseau logique.

Si l'adresse IP de l'élément cible ne figure pas dans la mémoire cache ARP de l'élément 110, ce dernier doit commencer par envoyer une requête selon un mode de diffusion générale, appelé *broadcast* en terminologie anglo-saxonne, pour permettre à l'élément cible d'indiquer son adresse MAC à l'élément 110. Cette requête est de la forme suivante : « *Quelle est l'adresse MAC correspondant à l'adresse IP @IP_B ? Répondez à @MACA* ». Les données à envoyer sont mémorisées et retenues tant que l'adresse MAC du destinataire est inconnue.

S'agissant d'une requête de type *broadcast*, tous les éléments communicants connectés au support physique la reçoivent. En analysant le contenu de la requête, chaque élément peut déterminer si la requête lui est destinée ou non.

L'élément communicant qui possède cette adresse IP est le seul à répondre. Il transmet à l'élément ayant transmis la requête une réponse ARP du type « *Je suis @IP_B, mon adresse MAC est @MAC_B* ». Pour émettre cette réponse au bon élément, il crée une entrée dans sa mémoire cache ARP à partir des données contenues dans la requête ARP reçue.

L'élément 110 à l'origine de la requête ARP reçoit alors la réponse, met à jour sa mémoire cache ARP avec l'adresse MAC reçue. Il peut ensuite envoyer les données mémorisées. Les commutateurs Ethernet mettent à jour leur table de commutation de la même façon, au passage des trames, afin de commuter les futurs échanges sur les bons ports et ainsi éviter des diffusions générales de messages.

La mise à jour des mémoires caches ARP est généralement à durée de validité limitée. Ces opérations sont donc à renouveler régulièrement, lorsque cette durée a expiré.

Alors que de tels échanges ne perturbent pas le réseau de communication en phase d'initialisation des équipements, ils deviennent indésirables dans des phases de simulation temps réel car ces échanges ont une influence directe sur la latence et la gigue du réseau et des couches réseaux des éléments communicants. Il convient de remarquer qu'il n'est généralement pas possible de désactiver le mécanisme ARP de gestion des adresses dans les éléments communicants et les commutateurs.

Pour éviter de tels échanges de données, les procédés et les dispositifs selon l'invention comprennent un mécanisme selon lequel les adresses MAC et IP d'un élément communicant transmettant un message sont transmis dans les données du message. Ainsi, les procédés et les dispositifs selon l'invention permettent notamment de,

- renseigner la mémoire cache ARP de façon statique selon un protocole spécifique décrit ci-dessous ;
- provoquer une mise à jour de la mémoire cache des commutateurs Ethernet, avant les échanges temps réel selon ce même protocole spécifique ; et,
- indiquer, si possible, au mécanisme ARP des éléments communicants que la mémoire cache est de validité permanente.

Ainsi, afin que les mémoires caches de résolution d'adresse soient renseignées avant l'envoi de données utilisateur temps réel, un protocole applicatif est mis en place sur la base du format de message illustré sur la figure 2.

Les adresses MAC et IP de l'émetteur sont ajoutées directement dans les données utiles (appelées *payload* en terminologie anglo-saxonne) des trames transmises conformes, ici, au protocole UDP. Ainsi, chaque élément communicant peut renseigner de façon statique sa mémoire cache ARP et, par conséquent, inhiber l'émission de messages ARP.

Comme illustré sur la figure 2, les données sont, de préférence, transmises selon le protocole UDP. Chaque trame transmise 200 comprend ainsi un en-tête Ethernet et IP 205, un en-tête UDP 210 et un champ 215 de données utiles UDP. Les données utiles UDP 215 comprennent elles-mêmes
5 les adresses MAC 220 et IP 225 de l'émetteur des données ainsi que les données utiles 230 proprement dites, c'est-à-dire les données métiers.

Afin de provoquer la mise à jour des caches ARP, au moins un message est avantageusement toujours émis selon le mode *broadcast* par l'élément maître à destination des éléments esclaves avant le début des
10 échanges temps réel. Sur réception de ce message, chaque élément esclave met à jour sa mémoire cache ARP et répond à l'élément maître par un message similaire.

Les commutateurs Ethernet ainsi que l'élément maître mettent alors à jour, à leur tour, leur table de commutation.

15 Si la durée de vie des mémoires caches ARP est limitée, charge à l'élément communicant de mettre à jour sa mémoire cache avant l'expiration de la durée de vie. Ceci garantit qu'aucun message ARP n'est émis sur le réseau pendant les échanges temps réel.

La figure 3 illustre un exemple de séquence de messages transmis
20 lors de la mise en œuvre de l'environnement de simulation représenté sur la figure 1.

Préalablement à l'échange temps réel des données, un message 300, conforme au format illustré sur la figure 2, est envoyé par l'élément maître 110 selon le mode *broadcast*. Ce message contient, en particulier, les adresses
25 MAC et IP de l'élément maître, notées MACM et IPM, respectivement, dans le champ des données utiles UDP.

L'envoi de ce message, relayé par les commutateurs du réseau commuté 105, aux éléments esclaves 115-1 à 115-3, permet à ces derniers de renseigner leur mémoire cache ARP avec le couple d'adresses MAC et IP de
30 l'élément maître.

Chaque élément esclave émet alors une réponse (305-1 à 305-3) à destination de l'élément maître, toujours selon le format illustré sur la figure 2. A

la réception de ces messages, l'élément maître renseigne à son tour sa mémoire cache ARP avec les adresses MAC et IP de chaque élément esclave.

Par exemple, l'élément esclave 115-1, ayant les adresses MAC et IP MAC1 et IP1, transmet le message 305-1 selon le mode *unicast*, c'est-à-dire
5 selon un mode de transmission individuelle ou point à point, à l'élément maître 110. Ce message, comprenant les adresses MAC1 et IP1 dans le champ des données utiles UDP, permet à l'élément maître 110 de mettre à jour sa mémoire cache ARP avec ce couple de valeurs.

De même, les éléments esclaves 115-2 et 115-3, ayant les adresses
10 MAC et IP MAC2 et IP2 et MAC3 et IP3, respectivement, transmettent chacun un message (305-2 et 305-3) selon le mode *unicast* à l'élément maître 110. Ces messages permettent à l'élément maître 110 de mettre à jour sa mémoire cache ARP avec ces couples de valeurs.

Les commutateurs Ethernet du réseau 105 mettent également à jour
15 leurs tables de commutation lorsqu'ils transfèrent ces messages.

Il convient de remarquer ici que si un élément communicant ne peut pas être paramétré pour augmenter la durée de validité de sa mémoire cache de telle sorte que cette durée soit supérieure à celle de la simulation, l'élément communicant doit mettre à jour son cache selon une période inférieure à celle
20 correspondant à la durée de sa validité.

Suite au message de l'élément maître et aux réponses des éléments esclaves, les échanges temps réel de données peuvent débuter. Les messages échangés durant cette phase de simulation respectent également le format de message illustré sur la figure 2 pour permettre aux éléments communicants de
25 mettre à jour leur cache si une durée de validité permanente n'est pas paramétrable.

Ainsi, lors des échanges temps réel de données, aucun message de résolution d'adresse n'est transmis par le réseau de communication.

Afin de minimiser le temps utilisé par la couche IP de chaque
30 élément communicant pour encoder et décoder les paquets de niveau réseau et minimiser la bande passante sur le réseau, les mécanismes de fragmentation IP ne sont, de préférence, pas mis en oeuvre.

Cependant, ce mécanisme ne pouvant être que rarement désactivé au niveau des éléments communicants, la taille des messages émis est, de préférence, inférieure à la taille maximale d'un paquet unitaire, aussi appelée MTU (sigle de *Maximum Transfer Unit* en terminologie anglo-saxonne).

5 Ainsi, selon un mode de réalisation particulier, la taille des données transmises par chaque message est telle que la taille de la trame UDP encapsulant ces données et les adresses MAC et IP de l'émetteur est inférieure à la taille d'un paquet IP, c'est-à-dire à 1500 octets.

Toujours pour minimiser la latence et la gigue sur le réseau dit temps
10 réel, le système selon l'invention met avantageusement en œuvre un protocole de niveau transport exempt de connexion pour éviter l'émission, pendant les échanges temps réel de données, de messages indésirables liés à la connexion, tels que, par exemple, les messages destinés à vérifier les interfaces réseaux, c'est-à-dire les messages de type *keepalive*, ou les
15 messages d'acquiescement.

De plus, la mise en œuvre d'un tel protocole permet de favoriser la simplicité des implémentations, basée sur la répétition de l'information et non sur l'acquiescement de chaque message reçu. Un mécanisme d'acquiescement peut nécessiter une mise en œuvre plus complexe et, par conséquent, moins
20 performante et non adaptée à optimiser le déterminisme des communications.

Les couches de programmation POSIX (acronyme de « *Portable Operating System Interface* » en terminologie anglo-saxonne) du système d'exploitation des éléments communicants sont avantageusement utilisées pour mettre en œuvre l'invention afin de bénéficier d'un niveau de programmation
25 normalisé.

Enfin, l'invention utilise, de préférence, des mécanismes de gestion de priorité mis en œuvre au niveau des processus logiciels afin que le processus d'échange en temps réel ne soit pas préempté. Ce mécanisme peut également être déployé pour ségréguer différents types de flux tels que des flux
30 temps réel de simulation et des flux de supervision ou de diagnostic. De façon avantageuse, le processus logiciel en charge des échanges temps réel bénéficie de la priorité la plus élevée.

Ainsi, une priorité est attribuée par tâche de traitement et non par donnée ni par port. En d'autres termes, une tâche est ici associée à un port de communication dédié et à un seul, ce qui revient à considérer, indirectement, qu'un port, par exemple un port associé à une tâche de simulation, est prioritaire sur un autre, par exemple un port associé à une tâche de diagnostic.

Le système conforme à l'invention repose ainsi notamment sur les caractéristiques suivantes :

- une configuration spécifique des commutateurs Ethernet et des éléments communicants pour inhiber tout trafic parasite et configurer les protocoles de résolution d'adresse afin de minimiser les échanges de données réseau ;
- un protocole spécifique incluant les informations d'adressage et permettant de s'affranchir de l'utilisation d'un protocole de résolution d'adresse ;
- des échanges de paquets exempts de fragmentation IP ;
- l'utilisation d'un protocole de transport exempt de connexion pour inhiber les échanges protocolaires de niveau transport ; et,
- des règles programmatiques permettant de gérer plusieurs niveaux de priorité pour ségréguer, par exemple, les échanges temps réel de simulation des échanges de diagnostic.

Bien que les performances observées soient dépendantes du réseau de communication mis en place et des performances des éléments communicants, l'invention permet d'atteindre une latence de l'ordre de la milliseconde (d'une application de l'élément maître à une application de l'élément esclave) et une gigue de l'ordre de 500 micro secondes dans un réseau commuté Ethernet utilisant des éléments de communication standard.

La figure 4 illustre certaines étapes d'un exemple d'algorithme pouvant être utilisé dans un élément communicant, conformément à l'invention, pour transmettre et recevoir des données.

L'algorithme comprend ici une phase d'initialisation (étapes 400 et 405) et une phase opérationnelle (étape 410 à 450).

Durant la phase d'initialisation, le trafic parasite est inhibé (étape 400). Cette étape est liée à l'élément communicant lui-même et à ses capacités de configuration.

5 Par ailleurs, le protocole de résolution d'adresse est configuré (étape 405) selon le mécanisme d'initialisation décrit précédemment pour mettre à jour la mémoire ARP de l'élément communicant et pour modifier sa période de validité ou mettre en œuvre un mécanisme de mise à jour dont la période est inférieure à celle de la validité de la mémoire.

10 Les étapes 400 et 405 peuvent être exécutées dans un ordre ou parallèlement.

Si des données doivent être transmises (étape 410), les adresses MAC et IP de l'élément communicant, c'est-à-dire de la source, sont déterminées (étape 415). Simultanément, avant ou après, un test est de préférence effectué sur la taille des données (étape 420). Si la taille des
15 données est supérieure à une valeur prédéterminée correspondant, de préférence, à la MTU du réseau, les données sont ignorées. Comme indiqué précédemment, la taille maximale des données est déterminées selon la taille des paquets échangés pour éviter de provoquer une fragmentation sur le réseau.

20 La comparaison de la taille des données à transmettre est, de préférence, effectuée au niveau applicatif.

Les données à transmettre sont ensuite encapsulées avec les adresses MAC et IP de l'élément communicant (étape 425) avant d'être transmises sous forme de trame (étape 430).

25 Si une trame est reçue par l'élément communicant (étapes 410 et 435), la trame est désencapsulée et les adresses MAC et IP de l'élément communicant ayant transmis ces données, contenues dans les données utiles, sont extraites (étape 440) et utilisées pour mettre à jour, si nécessaire, la mémoire ARP de l'élément communicant (étape 445). La mise à jour de la
30 mémoire ARP doit être effectuée une seule fois si sa durée de vie est illimitée ou périodiquement, en fonction de sa durée de validité.

Les données transmises sont alors traitées selon un mécanisme standard (étape 450).

Un élément communicant tel que l'élément maître 110 ou l'un des éléments esclaves 115-1 à 115-3 adapté à mettre en œuvre l'invention ou une partie de l'invention est illustré sur la figure 5. L'élément communicant 500 est
5 par exemple une station de travail, un serveur ou un micro-ordinateur.

L'élément communicant 500 comporte ici un bus de communication 505 auquel sont reliés :

- 10 - une unité centrale de traitement ou microprocesseur 510 (CPU, sigle de *Central Processing Unit* en terminologie anglo-saxonne) ;
- une mémoire morte 515 (ROM, acronyme de *Read Only Memory* en terminologie anglo-saxonne) pouvant comporter les programmes "Prog", "Prog1" et "Prog2" ;
- 15 - une mémoire vive ou mémoire cache 520 (RAM, acronyme de *Random Access Memory* en terminologie anglo-saxonne) comportant des registres adaptés à enregistrer des variables et paramètres créés et modifiés au cours de l'exécution des programmes précités ; et,
- une interface de communication 550 adaptée à transmettre et à recevoir des données.

20 Optionnellement, l'élément communicant 500 peut également disposer :

- 25 - d'un écran 525 permettant de visualiser des données et/ou de servir d'interface graphique avec l'utilisateur qui pourra interagir avec les programmes précités, à l'aide d'un clavier et d'une souris 530 ou d'un autre dispositif de pointage, un écran tactile ou une télécommande ;
- d'un disque dur 535 pouvant comporter les programmes "Prog", "Prog1" et "Prog2" précités et des données traitées ou à traiter ; et,
- d'un lecteur de cartes mémoires 540 adapté à recevoir une carte mémoire 545 et à y lire ou à y écrire des données traitées ou à traiter.

30 Le bus de communication permet la communication et l'interopérabilité entre les différentes parties de l'élément communicant 500 ou reliées à lui. La représentation du bus n'est pas limitative et, notamment, l'unité

centrale est susceptible de communiquer des instructions à toute partie de l'élément communicant 500 directement ou par l'intermédiaire d'une autre partie de l'élément communicant 500.

5 Le code exécutable de chaque programme peut être stocké, par exemple, dans le disque dur 535 ou en mémoire morte 515.

Selon une variante, la carte mémoire 545 peut contenir des données ainsi que le code exécutable des programmes précités qui, une fois lu par l'élément communicant 500, est stocké dans le disque dur 535.

10 Selon une autre variante, le code exécutable des programmes pourra être reçu, au moins partiellement, par l'intermédiaire de l'interface 550, pour être stocké de façon identique à celle décrite précédemment.

De manière plus générale, le ou les programmes pourront être chargés dans un des moyens de stockage de l'élément communicant 500 avant d'être exécutés.

15 L'unité centrale 510 va commander et diriger l'exécution des instructions ou portions de code logiciel du ou des programmes, instructions qui sont stockées dans le disque dur 535 ou dans la mémoire morte 515 ou bien dans les autres éléments de stockage précités. Lors de la mise sous tension, le ou les programmes qui sont stockés dans une mémoire non volatile, par
20 exemple le disque dur 535 ou la mémoire morte 515, sont transférés dans la mémoire vive 520 qui contient alors le code exécutable du ou des programmes, ainsi que des registres pour mémoriser les variables et paramètres nécessaires à la mise en œuvre de l'invention.

25 Naturellement, pour satisfaire des besoins spécifiques, une personne compétente dans le domaine de l'invention pourra appliquer des modifications dans la description précédente.

REVENDEICATIONS

- 5 1. Procédé pour transmettre en temps réel au moins une donnée applicative d'un premier élément communicant à au moins un second élément communicant, à travers un réseau de communication commuté configuré en mode maître/esclave et utilisant un protocole exempt de connexion, ce procédé étant caractérisé en ce qu'il comprend les étapes suivantes,
- 10 - si le protocole de communication dudit premier élément communicant comprend au moins une fonction adaptée à transmettre au moins une information audit second élément, ladite au moins une information étant distincte des données applicatives, et si ladite au moins une fonction peut être inhibée, inhibition de ladite au moins une fonction (400) ;
- 15 - si le protocole de communication dudit premier élément communicant est adapté à mettre en œuvre une fragmentation des données et si ladite fragmentation des données peut être inhibée, inhibition de ladite fragmentation des données ; et,
- initialisation et configuration (405) de la mémoire de résolution d'adresse dudit premier élément communicant pour mémoriser les adresses logiques et physiques dudit au moins un second élément communicant.
- 20
2. Procédé selon la revendication précédente selon lequel ladite configuration de ladite mémoire de résolution d'adresse comprend le paramétrage de ladite mémoire de résolution d'adresse pour augmenter sa
- 25 durée de validité.
3. Procédé selon la revendication 1 ou la revendication 2 comprenant en outre les étapes suivantes,
- détermination (415) des adresses logique et physique dudit premier élément communicant ;
- 30 - encapsulation (425) desdites adresses logique et physique et de ladite au moins une donnée applicative dans une zone de données utiles d'une

trame conforme au protocole de communication dudit réseau de communication commuté ; et,

- transmission (430) de ladite trame audit au moins un second élément communicant.

5 4. Procédé selon l'une quelconque des revendications précédentes selon lequel la taille de ladite au moins une donnée applicative est inférieure à une valeur prédéterminée de telle sorte qu'une seule trame conforme au protocole de communication dudit réseau de communication commuté est nécessaire à la transmission de ladite au moins une donnée applicative et
10 desdites adresses logique et physique.

5. Procédé selon l'une quelconque des revendications précédentes selon lequel ladite étape d'initialisation de la mémoire de résolution d'adresse dudit premier élément communicant comprend une étape de transmission d'une trame comprenant lesdites adresses logique et physique dudit premier élément
15 communicant dans une zone de données utiles et, en réponse, la réception d'au moins une trame comprenant lesdites adresses logique et physique dudit au moins un second élément communicant dans une zone de données utiles.

6. Procédé selon l'une quelconque des revendications précédentes comprenant en outre une étape de mise à jour périodique de ladite mémoire de
20 résolution d'adresse dudit premier élément communicant à la réception d'une trame reçue dudit au moins un second élément communicant, ladite trame reçue comprenant lesdites adresses logique et physique dudit second élément communicant dans une zone de données utiles.

7. Procédé selon l'une quelconque des revendications précédentes
25 selon lequel ledit réseau de communication est un réseau de type Ethernet.

8. Procédé selon l'une quelconque des revendications précédentes selon lequel un niveau de priorité est associé à la tâche applicative en charge de ladite au moins une donnée applicative, un port de communication étant affecté à ladite tâche applicative, de telle sorte que ledit niveau de priorité est
30 déterminé et géré indépendamment du protocole de transmission dudit réseau de communication.

9. Programme d'ordinateur comprenant des instructions adaptées à la mise en œuvre de chacune des étapes du procédé selon l'une quelconque des revendications précédentes.

5 10. Dispositif comprenant des moyens adaptés à la mise en œuvre de chacune des étapes du procédé selon l'une quelconque des revendications 1 à 8.

1/4

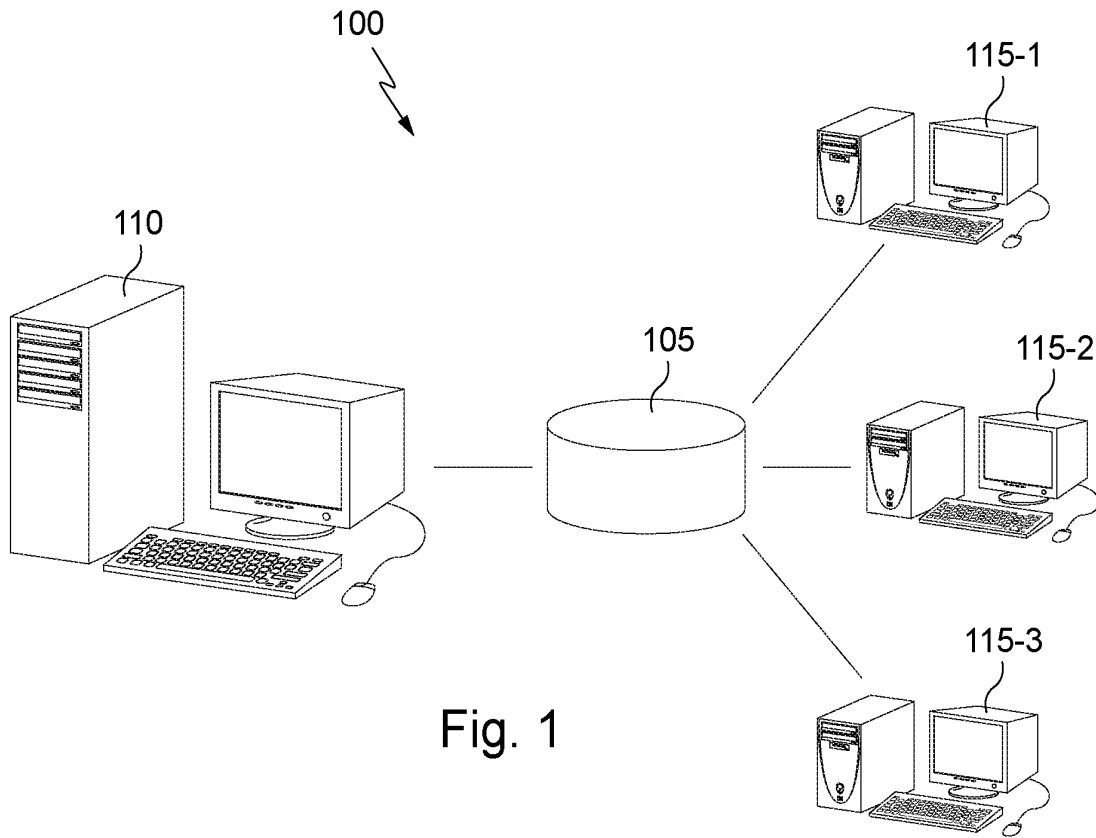


Fig. 1

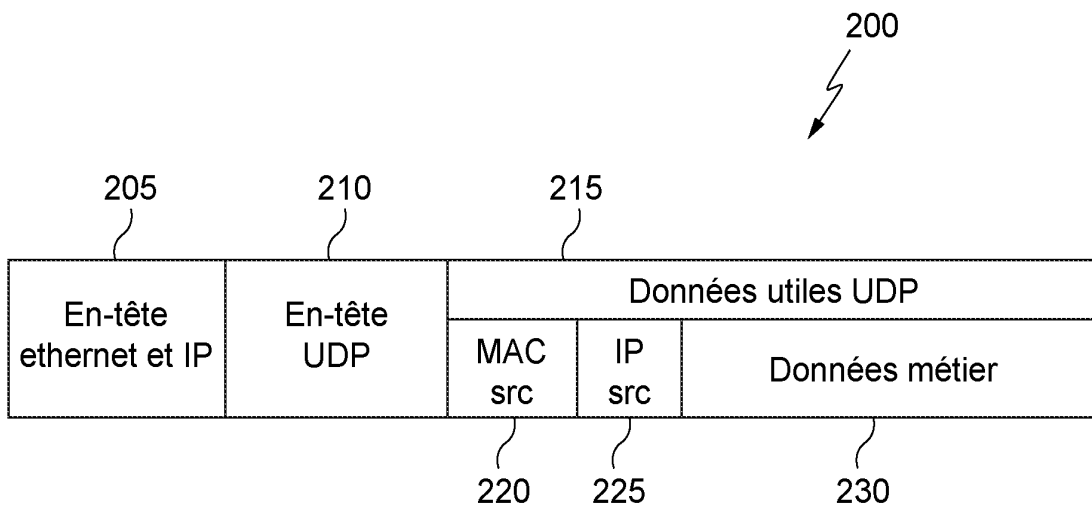


Fig. 2

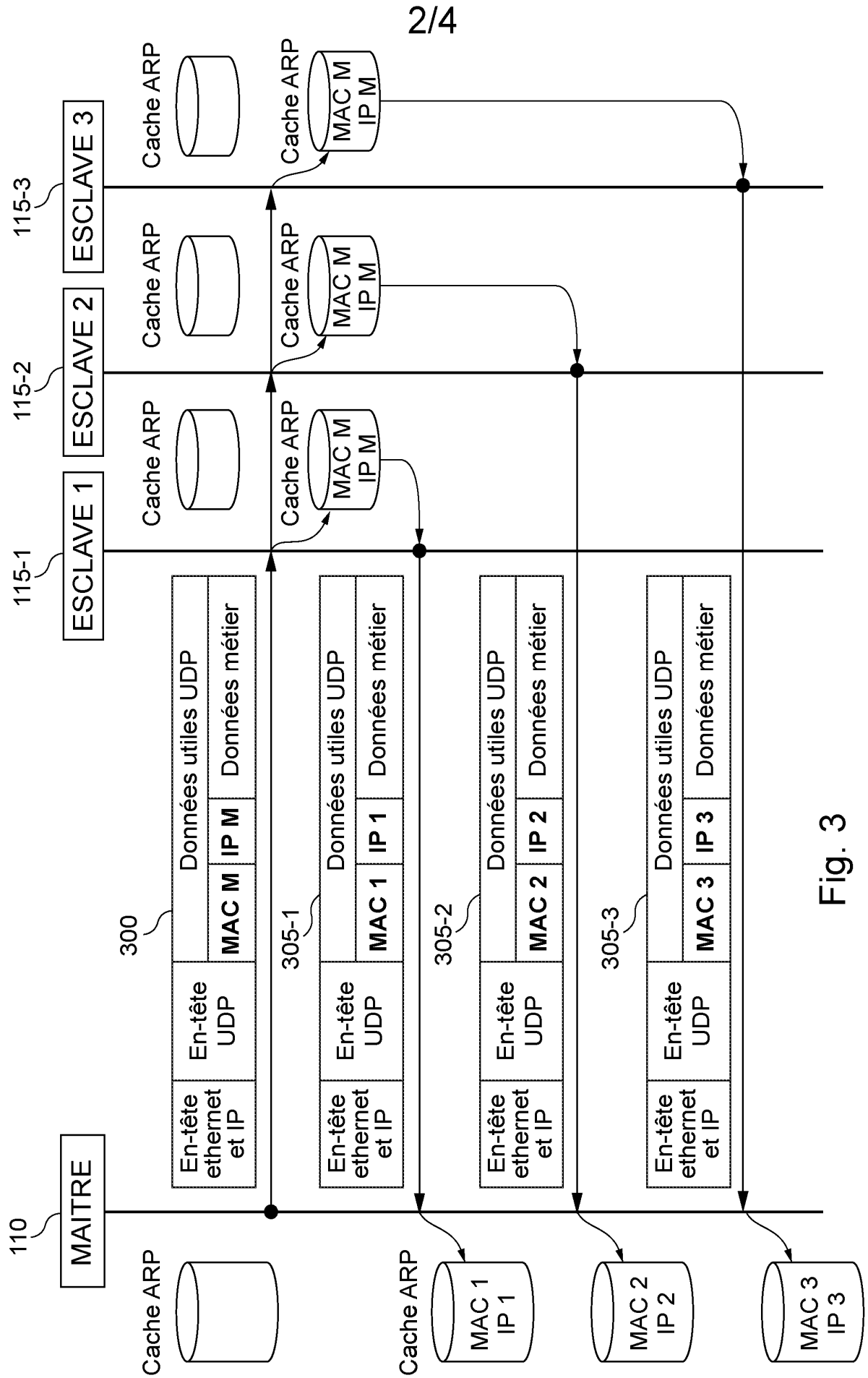


Fig. 3

3/4

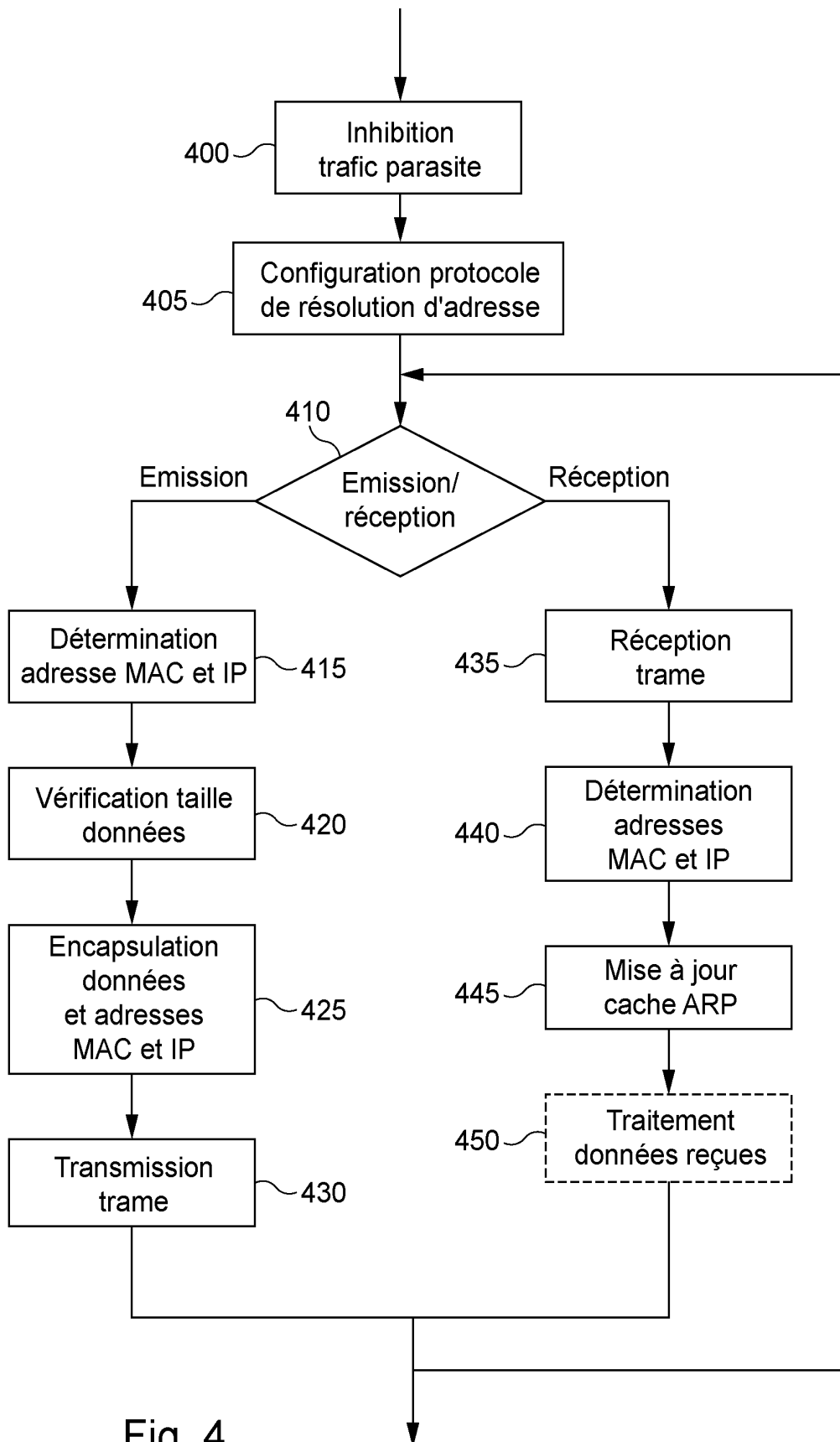


Fig. 4

4/4

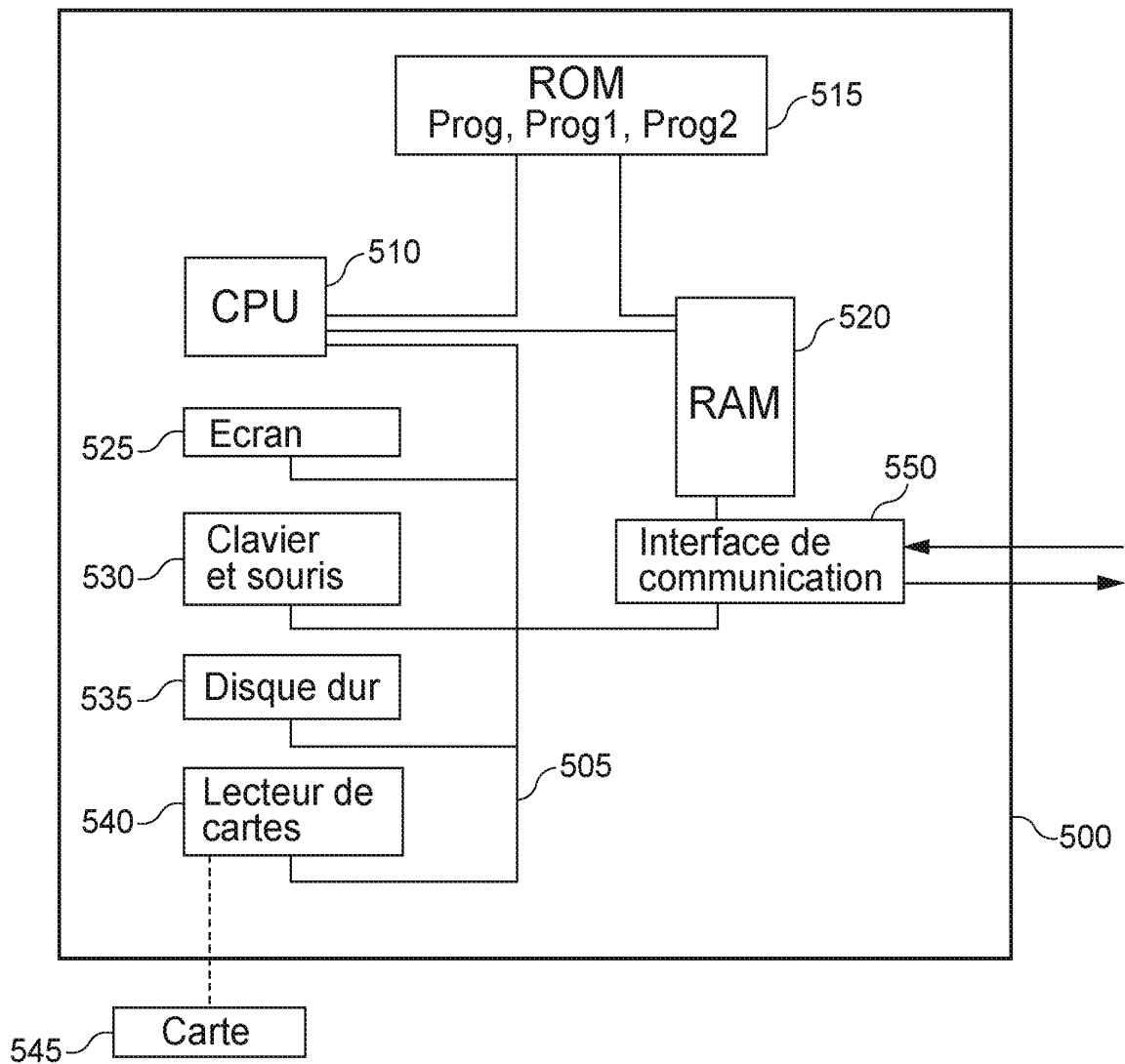


Fig. 5

**RAPPORT DE RECHERCHE
 PRÉLIMINAIRE**

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

N° d'enregistrement
 national

FA 706943
 FR 0852891

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	LARRY HANNETT: "Ethernet in Avionics Networks" ETHERNET ALLIANCE WHITE PAPER, [Online] janvier 2007 (2007-01), pages 0-7, XP002515172 Extrait de l'Internet: URL:http://www.ethernetalliance.org/attach ments/128_ETHERNET_AVIONICS.PDF> [extrait le 2009-02-13] * page 2 - page 6; figure 1 *	1-10	H04L29/12 H04L12/56
A	US 2007/223494 A1 (HYYRYNEN ALTTI [FI] ET AL) 27 septembre 2007 (2007-09-27) * abrégé * * alinéa [0009] - alinéa [0032] *	1-10	
			DOMAINES TECHNIQUES RECHERCHÉS (IPC)
			H04L B64C
		Date d'achèvement de la recherche	Examineur
		16 février 2009	Lebas, Yves
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 0852891 FA 706943**

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 16-02-2009

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
US 2007223494 A1	27-09-2007	WO 2007107624 A2	27-09-2007
