

12

DEMANDE DE BREVET D'INVENTION

A1

22 Date de dépôt : 02.10.00.

30 Priorité : 18.02.00 US 09507546.

43 Date de mise à la disposition du public de la demande : 24.08.01 Bulletin 01/34.

56 Liste des documents cités dans le rapport de recherche préliminaire : *Ce dernier n'a pas été établi à la date de publication de la demande.*

60 Références à d'autres documents nationaux apparentés :

71 Demandeur(s) : HEWLETT PACKARD COMPANY — US et INTEL CORPORATION — US.

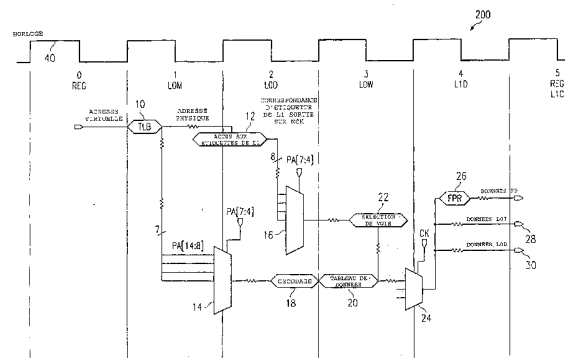
72 Inventeur(s) : RIEDLINGER REID JAMES, MULLA DEAN A et GRUTKOWSKI TOM.

73 Titulaire(s) :

74 Mandataire(s) : REGIMBEAU.

54 SYSTEME ET PROCEDE UTILISANT L'ACCES SPECULATIF A L'ANTEMEMOIRE POUR DES PERFORMANCES AMELIOREES.

57 Un système et procédé qui prévoient une structure d'antémémoire permettant d'accéder en avance aux données de la structure d'antémémoire (20) sont présentés. En réponse à la réception d'une demande d'accès en mémoire, le modèle d'antémémoire (200) commence à accéder aux données du niveau d'antémémoire avant qu'une détermination n'ait été effectuée pour savoir si une vraie coïncidence a été réalisée pour ce niveau d'antémémoire. Le modèle d'antémémoire (200) présenté permet donc d'accéder de façon spéculative aux données en antémémoire avant qu'une détermination n'est été effectuée pour savoir si une adresse en mémoire est vraiment présente dans l'antémémoire. Un mode de réalisation préféré met en oeuvre un chemin parallèle en commençant l'accès aux données en antémémoire tandis qu'une détermination est effectuée pour savoir si une vraie coïncidence d'antémémoire a été réalisée. Ainsi, les données en antémémoire sont rappelées en avance de la structure d'antémémoire et sont disponibles en temps voulu.



La présente invention concerne en général les accès à l'antémémoire dans un processeur, et en particulier un système et procédé pour accéder de façon spéculative aux données dans une antémémoire avant de vérifier une coïncidence d'étiquette pour cette antémémoire.

5 Les systèmes informatiques peuvent utiliser une hiérarchie de mémoire à plusieurs niveaux, avec une mémoire relativement rapide, coûteuse mais de capacité limitée au plus haut niveau de la hiérarchie et en continuant avec une mémoire relativement plus lente, moins coûteuse mais de plus grande capacité au plus bas niveau de la hiérarchie. La hiérarchie peut inclure une petite
10 mémoire rapide appelée antémémoire, soit intégrée physiquement dans un processeur, soit installée physiquement à proximité du processeur dans un souci de rapidité. Le système informatique peut utiliser des antémémoires d'instructions et des antémémoires de données séparées. De plus, le système informatique peut utiliser plusieurs niveaux d'antémémoires. L'utilisation d'une
15 antémémoire est généralement transparente pour un programme informatique au niveau des instructions et peut ainsi être ajoutée à une architecture informatique sans modifier le jeu d'instructions ou nécessiter la modification des programmes existants.

Les processeurs informatiques incluent typiquement une antémémoire
20 pour stocker des données. Lors de l'exécution d'une instruction qui nécessite un accès à la mémoire (par exemple, lecture ou écriture en mémoire), un processeur accède typiquement à l'antémémoire pour essayer de satisfaire l'instruction. Bien sûr, il est souhaitable de mettre en œuvre l'antémémoire d'une manière qui permet au processeur d'accéder à l'antémémoire de
25 manière efficace. C'est-à-dire, il est souhaitable de mettre en œuvre l'antémémoire d'une manière telle que le processeur soit capable d'accéder à l'antémémoire (c'est-à-dire, lecture ou écriture en antémémoire) rapidement de sorte que le processeur puisse être capable d'exécuter les instructions rapidement.

Les modèles d'antémémoire de l'art antérieur pour les processeurs informatiques nécessitent typiquement que des "données de commande" soient
30 disponibles avant le début d'un accès aux données en antémémoire. Ces "données de commande" indiquent si une adresse souhaitée (c'est-à-dire, une adresse nécessaire à une demande d'accès en mémoire) est contenue dans l'antémémoire. Par conséquent, les antémémoires de l'art antérieur sont

typiquement mises en œuvre en série, où après que l'antémémoire a reçu une demande d'accès en mémoire, des données de commande sont obtenues pour la demande, et ensuite, si les données de commande indiquent que l'adresse souhaitée est contenue dans l'antémémoire, le tableau de données de l'antémémoire fait l'objet d'un accès pour satisfaire à la demande d'accès en mémoire.

Ainsi, les modèles d'antémémoire de l'art antérieur génèrent typiquement des données de commande indiquant si une vraie "coïncidence" d'antémémoire a été réalisée pour un niveau d'antémémoire, et uniquement lorsqu'une vraie coïncidence d'antémémoire a été réalisée, les données en antémémoire font-elles réellement l'objet d'un accès pour satisfaire à la demande d'accès en mémoire. Une vraie "coïncidence" d'antémémoire a lieu lorsqu'un processeur demande un item d'une antémémoire et l'item est effectivement présent dans l'antémémoire. Un "accès manqué" à l'antémémoire survient lorsqu'un processeur demande un item d'une antémémoire et l'item n'est pas présent dans l'antémémoire. Les données de commande indiquant si une "vraie" coïncidence d'antémémoire a été réalisée pour un niveau d'antémémoire comprennent typiquement un signal de correspondance d'étiquette. Le signal de correspondance d'étiquette indique si une correspondance a été effectuée pour une adresse demandée dans les étiquettes d'un niveau d'antémémoire. Cependant, un tel signal de correspondance d'étiquette à lui tout seul n'indique pas si une vraie coïncidence d'antémémoire a été réalisée.

À titre d'exemple, dans un système multiprocesseur, une correspondance d'étiquette peut être réalisée pour un niveau d'antémémoire, mais la ligne d'antémémoire particulière pour laquelle la correspondance a été réalisée peut être erronée. Par exemple, la ligne d'antémémoire particulière peut être erronée parce qu'un autre processeur a inspecté cette ligne d'antémémoire particulière. Par conséquent, dans des systèmes multiprocesseurs, un signal MESI est aussi utilisé typiquement pour indiquer si une ligne en antémémoire est "Modifiée et Exclusive, Partagée ou Erronée". Donc, les données de commande qui indiquent si une vraie coïncidence d'antémémoire a été réalisée pour un niveau d'antémémoire comprennent typiquement un signal MESI, ainsi que le signal de correspondance d'étiquette. Uniquement si une correspondance d'étiquette est trouvée pour un niveau d'antémémoire et que le protocole MESI indique que cette correspondance d'étiquette est valable, les

données de commande indiquent-elles qu'une vraie coïncidence d'antémémoire a été réalisée ? En raison de ce qui précède, dans les modèles d'antémémoire de l'art antérieur, une détermination est d'abord effectuée pour savoir si une correspondance d'étiquette est trouvée pour un niveau d'antémémoire, et ensuite une détermination est effectuée pour savoir si le protocole MESI indique qu'une correspondance d'étiquette est valable. Ensuite, si une détermination a été effectuée indiquant qu'une vraie coïncidence d'étiquette a été réalisée, l'accès aux données en antémémoire réelles demandées commence.

Typiquement, dans les modèles d'antémémoire à plusieurs niveaux, on accède d'abord au premier niveau d'antémémoire (c'est-à-dire, L0) pour déterminer si une vraie coïncidence d'antémémoire pour une demande d'accès en mémoire est réalisée. Si une vraie coïncidence d'antémémoire n'est pas réalisée pour l'antémémoire de premier niveau, alors une détermination est effectuée pour le deuxième niveau d'antémémoire (c'est-à-dire, L1), et ainsi de suite, jusqu'à ce que la demande d'accès à la mémoire soit satisfaite par un niveau d'antémémoire. Si l'adresse demandée n'est trouvée dans aucun des niveaux d'antémémoire, alors le processeur envoie une demande à la mémoire principale du système pour essayer de satisfaire à la demande. Dans de nombreux modèles de processeur, le temps nécessaire pour accéder à un item pour une vraie coïncidence d'antémémoire est l'un des principaux facteurs de limitation de la fréquence d'horloge du processeur si le concepteur recherche un temps d'accès à l'antémémoire en un seul cycle. Dans d'autres modèles, le temps d'accès à l'antémémoire peut prendre plusieurs cycles, mais les performances d'un processeur peuvent être améliorées dans la plupart des cas lorsque le temps d'accès à l'antémémoire en cycles est réduit. Donc, l'optimisation du temps d'accès pour les coïncidences d'antémémoire est essentielle pour les performances du système informatique.

En se référant à la Figure 1, elle montre un exemple d'un modèle d'antémémoire typique de l'art antérieur. Typiquement, lorsqu'une instruction doit accéder à une adresse particulière, une adresse virtuelle est fournie par le processeur au système d'antémémoire. Comme il est bien connu dans l'art, une telle adresse virtuelle contient typiquement un champ d'indexation et un champ de numéro de page virtuel. L'adresse virtuelle est entrée dans un tampon de traduction à consultation séparée ("TLB") 10. Le TLB 10 est un

composant courant des architectures d'antémémoire modernes qui est bien connu dans l'art. Le TLB 10 assure une traduction de l'adresse virtuelle reçue en une adresse physique. Dans un système informatique, l'espace d'adressage virtuel est typiquement nettement plus grand que l'espace d'adressage physique. L'espace d'adressage physique est l'adresse en mémoire physique, réelle
5 d'un système informatique, qui comprend l'antémémoire, la mémoire principale, un disque dur, et tout autre élément auquel l'ordinateur peut accéder pour rappeler des données. Ainsi, pour qu'un système informatique soit capable d'accéder à l'ensemble de l'espace d'adressage physique, un mappage
10 physique complet entre adresses virtuelles et adresses physiques est typiquement prévu.

Une fois l'adresse virtuelle reçue traduite en une adresse physique par le TLB 10, le champ d'indexation de cette adresse physique est entré dans la ou les étiquettes 12 du niveau d'antémémoire, qui peut/peuvent être dupli-
15 quée(s) N fois pour N "voies" d'associativité. Tel qu'utilisé ici, le terme "voie" se réfère à une partition de l'antémémoire. Par exemple, l'antémémoire d'un système peut être partagée en un nombre quelconque de voies. Les antémémoires sont normalement partagées en quatre voies. L'index d'adresse physique est également entré dans le(s) tableau(x) 16 du niveau d'anté-
20 mémoire, qui peut/peuvent également être dupliqué(s) N fois pour N voies d'associativité.

À partir de l'étiquette/ des étiquettes 12 du niveau d'antémémoire, un signal de correspondance d'étiquette de voie est généré pour chaque voie. Le signal de correspondance d'étiquette de voie indique si une correspondance
25 pour l'adresse physique a été effectuée dans le(s) étiquette(s) du niveau d'antémémoire 12. Comme évoqué ci-dessus, dans les systèmes multiprocesseurs, un protocole MESI est utilisé typiquement pour indiquer si une ligne en antémémoire est modifiée et exclusive, partagée ou erronée. Par conséquent, dans de tels systèmes multiprocesseurs, le protocole MESI est combiné avec le
30 signal de correspondance d'étiquette de voie pour indiquer si une "vraie" coïncidence d'étiquette a été réalisée pour un niveau d'antémémoire. Ainsi, dans les systèmes multiprocesseurs, une vraie coïncidence d'étiquette est réalisée lorsque à la fois une correspondance d'étiquette est trouvée pour la ou les étiquettes 12 et que le protocole MESI indique que cette correspondance

d'étiquette est une correspondance valable. Par conséquent, sur la Figure 1, les circuits MESI 14 sont utilisés pour calculer un signal de "vraie" coïncidence d'étiquette pour déterminer si une vraie coïncidence d'étiquette a été réalisée pour ce niveau d'antémémoire. Dès qu'il a été déterminé par le MESI 14 qu'une "vraie" coïncidence d'étiquette a été réalisée pour ce niveau d'antémémoire, alors on accède au(x) tableau(x) de données 16 de ce niveau d'antémémoire, qui peut/ peuvent aussi être dupliqué(s) N fois pour N voies d'associativité, pour satisfaire à la demande d'accès en mémoire reçue. Plus spécifiquement, le signal de vraie coïncidence d'étiquette peut être utilisé pour commander un multiplexeur ("MUX") 18 pour sélectionner la voie de tableau de données appropriée pour sortir des données qui satisfont à la demande d'accès en mémoire reçue. Les données sélectionnées à partir du ou des tableaux de données 16 sont délivrées au noyau 20 de la puce, qui est l'unité d'exécution particulière (par exemple, une unité d'exécution à base d'entiers ou une unité d'exécution en virgule flottante) qui a émis la demande d'accès en mémoire pour l'antémémoire.

En raison de ce qui précède, les antémémoires de l'art antérieur sont typiquement mises en œuvre en série, où l'adresse physique est déterminée d'abord, ensuite il est déterminé si une correspondance d'étiquette est réalisée pour l'adresse physique demandée à l'intérieur d'un niveau d'antémémoire particulier, ensuite il est déterminé si une "vraie" coïncidence d'antémémoire est réalisée à l'intérieur du niveau d'antémémoire particulier et enfin on accède au(x) tableau(x) de données pour le niveau d'antémémoire particulier si une "vraie" coïncidence d'antémémoire a été réalisée. Ainsi, même si les antémémoires de l'art antérieur déterminent l'adresse physique relativement tôt, les données de l'antémémoire ne font pas l'objet d'un accès jusqu'à ce qu'il ait été déterminé si une "vraie" coïncidence d'antémémoire a été réalisée pour l'antémémoire. Un tel accès série aux données en antémémoire est peu avantageux en ce qu'il est lent. Une telle mise en œuvre d'antémémoire série ajoute en général de un à deux cycles d'horloge de latence d'accès aux données parce que les étiquettes d'antémémoire et le MESI doivent d'abord terminer la détermination pour savoir si une coïncidence valable a été réalisée pour l'antémémoire avant de commencer l'accès aux données. Ainsi, un tel accès série aux données en antémémoire nécessite un délai d'une durée longue

indésirable pour accéder aux données en antémémoire. Donc, les modèles d'antémémoire série de l'art antérieur augmentent la latence liée au rappel des données de l'antémémoire, ce qui ralentit l'unité d'exécution à l'intérieur du noyau d'une puce. C'est-à-dire, pendant que l'unité d'exécution attend les données en provenance de l'antémémoire, elle est bloquée, ce qui aboutit à des performances inférieures nettes pour le processeur d'un système.

En raison de ce qui précède, il existe le souhait d'un modèle d'antémémoire qui permet d'accéder aux données en antémémoire en temps voulu. C'est-à-dire, il existe le souhait d'un modèle d'antémémoire qui diminue la latence liée au rappel des données de l'antémémoire qui existe dans les modèles d'antémémoire de l'art antérieur. Il existe un souhait supplémentaire d'un modèle d'antémémoire qui permet d'accéder aux données en antémémoire en temps voulu, tout en vérifiant encore qu'une vraie coïncidence d'étiquette a été réalisée pour l'antémémoire afin d'assurer que les données appropriées font l'objet d'un accès pour satisfaire à une demande d'accès en mémoire. Par conséquent, il existe le souhait d'un modèle d'antémémoire qui permet d'accéder rapidement aux données en antémémoire, réduisant ainsi le nombre de blocages nécessaires dans les unités d'exécution qui demandent un accès en mémoire et améliorant les performances globales du système.

Ceux-ci ainsi que d'autres objets, caractéristiques et avantages techniques sont réalisés par un système et procédé prévoyant un modèle d'antémémoire qui, en réponse à la réception d'une demande d'accès en mémoire, commence à accéder aux données d'un niveau d'antémémoire avant qu'il n'ait été déterminé si une vraie coïncidence a été réalisée pour le niveau d'antémémoire. C'est-à-dire, un système et procédé sont prévus qui permettent d'accéder de manière spéculative aux données en antémémoire avant qu'il ne soit déterminé si une adresse en mémoire nécessaire pour satisfaire à une demande d'accès en mémoire est vraiment présente dans l'antémémoire.

Dans un mode de réalisation préféré, une structure d'antémémoire est prévue qui reçoit des demandes d'accès en mémoire en provenance d'au moins un processeur d'un système informatique. En réponse à la réception d'une telle demande d'accès en mémoire, la structure d'antémémoire commence à accéder à son/ses tableau(x) de données pour essayer de satis-

faire à la demande reçue, sans déterminer d'abord si une adresse en mémoire nécessaire pour satisfaire à la demande d'accès en mémoire reçue est vraiment présente dans la structure d'antémémoire. Dans le mode de réalisation préféré entre tous, une telle structure d'antémémoire est un niveau d'une antémémoire à plusieurs niveaux mise en œuvre pour un système informatique.

Dans un mode de réalisation préféré, l'antémémoire est mise en œuvre de sorte qu'une détermination est effectuée pour savoir si une adresse en mémoire nécessaire pour satisfaire à une demande d'accès en mémoire reçue est vraiment présente dans la structure d'antémémoire. C'est-à-dire, un mode de réalisation préféré détermine si une vraie coïncidence d'antémémoire est réalisée dans la structure d'antémémoire pour une demande d'accès en mémoire reçue. Toutefois, une telle détermination n'est pas effectuée avant le début de l'accès aux données en antémémoire. Au lieu de cela, dans un mode de réalisation préféré, une détermination pour savoir si une vraie coïncidence d'antémémoire est réalisée est effectuée dans la structure d'antémémoire en parallèle avec l'accès aux données de la structure d'antémémoire pour une demande d'accès en mémoire reçue. C'est-à-dire, un mode de réalisation préféré détermine si une correspondance d'étiquette est réalisée pour les étiquettes de la structure d'antémémoire et si un protocole MESI vérifie qu'une correspondance d'étiquette réalisée est une correspondance valable en parallèle avec l'accès au(x) tableau(x) de données de la structure d'antémémoire. Donc, plutôt que le chemin série des modèles d'antémémoire de l'art antérieur dans lesquels une correspondance d'étiquette est d'abord déterminée, ensuite une correspondance est vérifiée par un protocole MESI, ensuite les données en antémémoire font l'objet d'un accès pour satisfaire à une demande d'accès en mémoire reçue, un mode de réalisation préféré met en œuvre un chemin parallèle en commençant l'accès aux données en antémémoire pendant qu'une détermination est effectuée pour savoir si une vraie coïncidence d'antémémoire a été réalisée. Ainsi, les données en antémémoire sont rappelées en avance de la structure d'antémémoire et sont disponibles en temps voulu pour être utilisées par une unité d'exécution interrogatrice, dès qu'il a été déterminé qu'une vraie coïncidence d'antémémoire a été réalisée pour la structure d'antémémoire.

Dans un mode de réalisation préféré, l'accès aux données commence avant de déterminer si une vraie coïncidence d'antémémoire est réalisée uniquement pour des demandes d'accès en mémoire qui sont des demandes de lecture de données. Ainsi, pour les demandes "d'écriture", un mode de réalisation préféré détermine d'abord si une vraie coïncidence d'antémémoire est réalisée avant de commencer l'écriture demandée.

Il convient d'apprécier qu'un avantage technique d'un aspect de la présente invention est qu'une structure d'antémémoire est mise en œuvre pour permettre un accès plus rapide aux données de la structure d'antémémoire en commençant l'accès aux données avant qu'il ne soit déterminé si une vraie coïncidence d'antémémoire est réalisée pour la structure d'antémémoire. Par conséquent, le modèle d'antémémoire permet d'accéder aux données en antémémoire en temps voulu. Un autre avantage technique d'un aspect de la présente invention est qu'une structure d'antémémoire est mise en œuvre qui permet d'accéder aux données en antémémoire en temps voulu, tout en vérifiant toujours qu'une vraie coïncidence d'étiquette a été réalisée pour l'antémémoire afin d'assurer que les données appropriées font l'objet d'un accès pour satisfaire à une demande d'accès en mémoire reçue. Cependant, un autre avantage technique d'un aspect de la présente invention est qu'une structure d'antémémoire est mise en œuvre de sorte que les données de la structure d'antémémoire font l'objet d'un accès en parallèle avec la détermination de savoir si une vraie coïncidence d'antémémoire est réalisée pour la structure d'antémémoire, diminuant ainsi la latence liée au rappel des données de l'antémémoire qui existe dans les modèles d'antémémoire de l'art antérieur. C'est-à-dire, une structure d'antémémoire est mise en œuvre qui permet d'accéder rapidement aux données en antémémoire, réduisant ainsi le nombre de blocages nécessaires dans les unités d'exécution qui demandent un accès en mémoire et améliorant les performances globales du système.

Ce qui précède a exposé dans les grandes lignes les caractéristiques et avantages techniques de la présente invention de façon à ce que la description détaillée de l'invention qui suit soit mieux comprise. Des caractéristiques et avantages supplémentaires de l'invention seront décrits ci-après qui constituent l'objet des revendications de l'invention. Ceux compétents dans l'art devraient apprécier que la conception et le mode de réalisation spécifique présentés

peuvent être facilement utilisés comme une base pour modifier ou concevoir d'autres structures pour réaliser les mêmes buts de la présente invention. L'homme du métier devrait également réaliser que de telles constructions équivalentes ne s'éloignent pas de l'esprit et du domaine de l'invention tels qu'exposés dans les revendications ci-jointes.

Pour une compréhension plus complète de la présente invention, et de ses avantages, nous nous référons maintenant à la description suivante prise en conjonction avec le dessin ci-joint, où :

- la Figure 1 montre un modèle d'antémémoire typique de l'art antérieur, qui détermine si une "vraie" coïncidence d'étiquette est réalisée pour un niveau d'antémémoire et accède ensuite en série aux données en antémémoire ;
- la Figure 2 montre un mode de réalisation préféré pour un modèle d'antémémoire de la présente invention ;
- la Figure 3 montre un TLB mis en œuvre dans un mode de réalisation préféré ;
- la Figure 4 montre les circuits mis en œuvre pour un accès aux étiquettes d'antémémoire d'un mode de réalisation préféré ; et
- la Figure 5 montre les circuits mis en œuvre dans un mode de réalisation préféré pour indiquer si une vraie coïncidence d'antémémoire est réalisée pour le niveau d'antémémoire.

En se référant à la Figure 2, elle montre un mode de réalisation préféré de la présente invention. Comme montré, l'exécution d'un mode de réalisation préféré d'un modèle d'antémémoire 200 est représentée en conjonction avec un signal d'horloge 40 pour représenter le flux d'exécution à travers divers étages de pipeline. En général, un "étage de pipeline" est un cycle d'horloge individuel qui est utilisé pour l'occurrence de certain(s) événement(s) d'un accès en mémoire. Étant donné que l'utilisation des étages de pipeline est bien connue dans l'art, elle ne sera pas évoquée plus en détail dans ce document.

La Figure 2 représente un mode de réalisation préféré mis en œuvre pour le niveau L1 de l'antémémoire. Cependant, il convient de comprendre qu'un tel modèle peut être mis en œuvre pour un niveau d'antémémoire quelconque et n'est pas prévu pour se limiter uniquement à L1. À l'étage de pipeline REG O,

le niveau L1 de l'antémémoire 200 reçoit une adresse virtuelle pour une demande d'accès en mémoire d'un niveau d'antémémoire L0, qui est entrée dans un tampon de traduction à consultation séparée ("TLB") 10 au début de l'étage de pipeline LOM. Au cours de la première phase d'horloge (la phase haute de l'horloge) de LOM, le TLB 10 traduit l'adresse virtuelle reçue en une adresse physique. Comme évoqué ci-dessus, le TLB 10 est un composant courant des architectures d'antémémoire modernes qui est bien connu dans l'art. En se référant brièvement à la Figure 3, une mise en œuvre préférée pour le TLB 10 est représentée. Comme le montre la Figure 3, dans un mode de réalisation préféré, une adresse virtuelle sur 64 bits (VA[63:0]) est reçue par le TLB 10, et une adresse physique sur 45 bits (PA[44:0]) est sortie par le TLB 10. Toutefois, dans d'autres modes de réalisation, un nombre quelconque de bits peut être utilisé pour l'adresse virtuelle et l'adresse physique. Comme représenté sur la Figure 3, le TLB 10 reçoit une adresse virtuelle (VA[63:0]) et traduit cette adresse virtuelle en une adresse physique (PA[44:0]).

Dans la plupart des architectures d'antémémoire, les bits d'adresse inférieurs de l'adresse virtuelle et de l'adresse physique correspondent. Comme le montre la Figure 3, dans un mode de réalisation préféré, les douze bits inférieurs de l'adresse virtuelle (VA[11:0]) correspondent aux douze bits inférieurs de l'adresse physique (PA[11:0]). Toutefois, dans d'autres modes de réalisation, un nombre quelconque de bits de l'adresse virtuelle et de l'adresse physique peut correspondre. Étant donné que les douze bits inférieurs de l'adresse virtuelle et de l'adresse physique correspondent dans un mode de réalisation préféré, le TLB 10 traduit les bits non correspondants de l'adresse virtuelle (VA[63:12]) en l'adresse physique appropriée PA[44:12]. C'est-à-dire, le TLB 10 effectue une consultation pour déterminer le mappage pour l'adresse virtuelle reçue. En général, il existe un seul mappage dans le TLB 10 pour l'adresse virtuelle reçue. Étant donné que PA[11:0] correspond à VA[11:0] et le TLB 10 traduit VA[63:12] en PA[44:12], toute l'adresse physique PA[44:0] est déterminée et délivrée par le TLB 10.

En se référant à nouveau à la Figure 2, au cours de la deuxième phase d'horloge (la phase basse de l'horloge) de l'étage de pipeline LOM, la ou les étiquettes du niveau d'antémémoire font l'objet d'un accès, ce qui est représenté comme l'accès 12 aux étiquettes de L1 (accès aux étiquettes du

niveau d'antémémoire un). L'accès 12 aux étiquettes de L1 sert à accéder à l'étiquette/ aux étiquettes du niveau d'antémémoire et à déterminer une correspondance d'étiquette de voie pour la demande d'accès en mémoire. En se référant brièvement à la Figure 4, un mode de réalisation préféré pour l'accès 12 aux étiquettes de L1 est montré plus en détail. Dans un mode de réalisation préféré, les bits d'adresse physique sont utilisés pour l'accès aux étiquettes dès que ces bits d'adresse physique sont connus. Comme évoqué ci-dessus, dans un mode de réalisation préféré, les douze bits inférieurs de l'adresse physique (PA[11:0]) sont les mêmes que les douze bits inférieurs de l'adresse virtuelle (VA[11:0]). Ainsi, ces bits ne nécessitent aucune traduction par le TLB. Donc, les douze bits inférieurs de l'adresse physique (PA[11:0]) sont connus très tôt. C'est-à-dire, les douze bits inférieurs de l'adresse physique (PA[11:0]) sont connus un cycle entier plus tôt que l'ensemble de l'adresse physique (PA[44:0]) parce que les douze bits inférieurs sont connus dès que l'adresse virtuelle est reçue. Un mode de réalisation préféré utilise ces données connues en commençant le chemin d'accès 12 aux étiquettes de L1 en avance.

Le TLB 10 nécessite typiquement environ un cycle pour terminer. Au cours de la deuxième moitié de ce cycle (la phase basse de l'horloge), l'accès à l'étiquette/aux étiquettes de l'antémémoire commence, dans un mode de réalisation préféré. Comme le montre la Figure 4, les bits d'adresse physique PA[11:7] sont entrés dans les circuits de décodage 402. Toutefois, dans d'autres modes de réalisation, l'un quelconque des bits d'adresse physique connus peut être utilisé pour ce décodage. Les circuits de décodage 402 décodent les bits d'adresse physique PA[11:7] pour déterminer l'adresse physique nécessaire pour la demande d'accès en mémoire reçue, et ensuite les lignes de mots sont amorcées. Dès que les lignes de mots sont amorcées, la ou les étiquettes 404 sont extraites. Dans un mode de réalisation préféré, les étiquettes 404 sont dupliquées 8 fois pour 8 voies d'associativité. Ainsi, dans un mode de réalisation préféré, lorsque les lignes de mots sont amorcées, huit étiquettes différentes sont extraites des étiquettes 404. Bien sûr, dans d'autres modes de réalisation, un nombre quelconque de voies peut être mis en œuvre pour l'antémémoire.

Comme le montre la Figure 4, le décodage de PA[11:7] et l'extraction des étiquettes 404 sont effectués à l'étage de pipeline LOM. Ainsi, tandis que

le TLB 10 est toujours en train de terminer, l'accès aux étiquettes 404 d'antémémoire de L1 commence, dans un mode de réalisation préféré. Par conséquent, dans un mode de réalisation préféré, l'accès au TLB et l'accès aux étiquettes d'antémémoire sont effectués en parallèle. Il sera rappelé que les
5 modèles d'antémémoire de l'art antérieur accèdent typiquement en série au TLB et aux étiquettes d'antémémoire. C'est-à-dire, les modèles d'antémémoire de l'art antérieur accèdent typiquement d'abord au TLB et commencent ensuite à accéder aux étiquettes d'antémémoire uniquement après achèvement
10 du TLB. Ainsi, un mode de réalisation préféré réduit la latence liée à la détermination pour savoir si une correspondance d'étiquette est réalisée en accédant en avance à l'étiquette/ aux étiquettes du niveau d'antémémoire.

À l'étage de pipeline LOM, une étiquette est extraite pour chaque voie. Donc, dans un mode de réalisation préféré, huit étiquettes (étiquette de voie 0 à étiquette de voie 7) sont extraites à l'étage de pipeline LOM. Dans un mode
15 de réalisation préféré, les étiquettes sont conservées jusqu'à l'étage de pipeline LOD. À l'étage de pipeline LOD, le TLB 10 a terminé sa traduction et l'adresse physique souhaitée PA[44:0] est connue. C'est-à-dire, à l'étage de pipeline LOD, le TLB 10 a terminé son mappage de l'adresse virtuelle pour déterminer les bits de l'adresse physique qui étaient inconnus initialement (PA[44:12]).
20 Dans un mode de réalisation préféré, les bits [14:12] de l'adresse physique (c'est-à-dire, PA[14:12]) sont utilisés à l'étage de pipeline LOD pour sélectionner l'une des huit étiquettes extraites du tableau d'étiquettes 404 à l'étage de pipeline LOM. Dans d'autres modes de réalisation, l'un quelconque des bits de l'adresse physique qui résulte du mappage du TLB 10 (par exemple l'un
25 quelconque des bits PA[44:12]) peut être utilisé pour sélectionner l'une des étiquettes extraites à l'étage de pipeline LOM. Dans un mode de réalisation préféré, les huit étiquettes extraites du tableau d'étiquettes 404 sont entrées dans un MUX 406, et les bits PA[14:12] sont utilisés pour commander le MUX 406. Ainsi, les bits PA[14:12] sont utilisés pour sélectionner l'étiquette
30 de voie appropriée à sortir par le MUX 406.

Ainsi, à l'étage de pipeline LOD, un signal de correspondance d'étiquette de voie est généré par le MUX 406. Il convient de comprendre qu'il ne s'agit dans ce cas pas d'un "vrai" signal de coïncidence d'étiquette parce que les informations MESI n'ont pas été combinées avec le signal de correspondance à

ce moment-là. Au lieu de cela, l'accès aux étiquettes 12 génère strictement un signal de correspondance d'étiquette de voie. En se référant à nouveau à la Figure 2, elle montre qu'au cours de la première phase (la phase haute de l'horloge) de LOD, l'accès 12 aux étiquettes de L1 termine le chemin de correspondance d'étiquette, comme évoqué ci-dessus en se référant à la Figure 4. En parallèle avec l'achèvement de l'accès 12 aux étiquettes de L1, la première phase (la phase haute de l'horloge) de LOD est utilisée pour faire sortir les données d'adresse physique du TLB 10 vers le(s) tableau(x) de données en antémémoire. Ainsi, dans un mode de réalisation préféré, on commence à accéder au(x) tableau(x) de données du niveau d'antémémoire pour une adresse avant qu'il ne soit déterminé si une "vraie" coïncidence d'antémémoire est réalisée pour le niveau d'antémémoire. Il sera rappelé que les modèles d'antémémoire de l'art antérieur déterminent typiquement si une vraie coïncidence d'antémémoire est réalisée pour un niveau d'antémémoire et commencent ensuite à accéder en série au(x) tableau(x) de données du niveau d'antémémoire. C'est-à-dire, les modèles d'antémémoire de l'art antérieur déterminent typiquement d'abord si une vraie coïncidence d'antémémoire est réalisée pour un niveau d'antémémoire, et commencent un accès au(x) tableau(x) de données du niveau d'antémémoire uniquement après qu'il a été déterminé qu'une vraie coïncidence d'antémémoire a été réalisée. Ainsi, un mode de réalisation préféré réduit la latence liée à l'accès aux données du niveau d'antémémoire en accédant en avance de façon spéculative aux données.

Plus spécifiquement, dans un mode de réalisation préféré, le TLB 10 sort une adresse physique au cours de la deuxième phase (la phase basse de l'horloge) de LOM. Dans un mode de réalisation préféré, les bits [14:8] de l'adresse physique sont entrés dans un MUX 14. Toutefois, dans d'autres modes de réalisation, l'un quelconque des bits de l'adresse physique peut être utilisé. Dans le mode de réalisation préféré entre tous, une antémémoire de 256 kilo-octets est mise en œuvre, qui est répartie sur 16 bancs de 128 lignes par banc. Ainsi, dans le mode de réalisation préféré entre tous, les bits [14:8] de l'adresse physique peuvent être décodés pour identifier l'une quelconques des 128 lignes d'un banc. De plus, dans le mode de réalisation préféré entre tous, les bits [7:4] de l'adresse physique sont décodés pour sélectionner à quel

banc un accès doit être émis. Étant donné que ces bits de l'adresse physique sont connus en avance (par exemple, ils sont connus lorsque l'adresse virtuelle est reçue), le banc qui doit faire l'objet d'un accès peut être sélectionné en avance. Ainsi, un mode de réalisation préféré permet aux circuits de commande d'être configurés en avance, augmentant ainsi l'efficacité de l'accès en mémoire. Bien sûr, dans d'autres modes de réalisation, une taille d'antémémoire quelconque peut être mise en œuvre. De plus, dans d'autres modes de réalisation, un nombre quelconque de bancs (c'est-à-dire, de 0 banc à N bancs) peut être mis en œuvre pour l'antémémoire. En général, il est souhaitable de disposer du plus grand nombre possible de bancs mis en œuvre pour l'antémémoire.

Comme le montre la Figure 2, les PA[14:8] du résultat de TLB 10 sont entrés dans un MUX dynamique 14 au cours de la deuxième phase (la phase basse de l'horloge) de LOM. Ensuite, au cours de la première phase (la phase haute de l'horloge) de l'étage de pipeline LOD, le MUX 14 sort les PA[14:8], qui sont entrés dans les circuits de décodage 18. Par conséquent, dans un mode de réalisation préféré, les circuits de décodage 18 commencent leur décodage pour le(s) tableau(x) de données 20 en parallèle avec la détermination de la correspondance d'étiquette pendant l'accès 12 aux étiquettes de L1 au cours de la première phase (la phase haute de l'horloge) de LOD. Au fur et à mesure que les circuits de décodage 18 décodent les PA[14:8], les données de correspondance d'étiquette de l'accès 12 aux étiquettes de L1 se terminent à la fin de la première phase (la phase haute de l'horloge) de LOD. Ces données de correspondance d'étiquette de l'accès 12 aux étiquettes de L1 sont entrées dans un MUX dynamique 16, qui délivre ces données de correspondance d'étiquette au cours de la deuxième phase de LOD (la phase basse de l'horloge). Donc, le MUX 16 est utilisé dans un mode de réalisation préféré pour sélectionner le banc approprié pour une correspondance d'étiquette de voie. Plus spécifiquement, les PA[7:4] commandent le fonctionnement du MUX 16 pour acheminer la correspondance d'étiquette de voie vers le banc approprié. Le MUX 14 d'un mode de réalisation préféré est commandé par les PA[7:4] pour acheminer les adresses physiques vers le banc approprié. Au cours de la première phase (la phase haute de l'horloge) de LOW, les circuits de décodage 18 ont terminé et l'accès au tableau de

données 20 commence pour l'adresse décodée. C'est-à-dire, au cours de la première phase de LOW, le décodage d'adresse physique est disponible à partir des circuits de décodage 18, et les lignes de mots sont amorcées, entraînant l'extraction de huit voies de données du tableau de données 20 selon cette
5 adresse physique décodée. Par conséquent, au cours de la première phase de LOW, on accède de façon spéculative aux tableaux de données avant qu'une détermination n'ait été réalisée pour savoir si une vraie coïncidence d'étiquette a été réalisée pour l'antémémoire de L1.

Ensuite, plus tard au cours de la première phase de l'étage de pipeline
10 LOW, la sélection de voie est entrée par les circuits de sélection de voie 22 dans le(s) tableau(x) de données 20. Ces données de sélection de voie sont reçues ultérieurement parce qu'elle subissent l'accès d'un demi-cycle supplémentaire à travers les étiquettes pendant l'accès 12 aux étiquettes de L1. Donc, dans un mode de réalisation préféré, les données de sélection de voie ne
15 sont pas reçues par le tableau de données 20 jusqu'à la fin de la première phase (la phase haute de l'horloge) de l'étage de pipeline LOW. Une seule des huit voies de l'antémémoire sera à l'état haut, et les circuits, tels qu'un MUX (non représenté), sont utilisés pour sélectionner la ligne de données extraite du tableau de données en antémémoire 20 pour la voie qui est à l'état haut.
20 C'est-à-dire, l'une des huit lignes de données extraites du tableau de données en antémémoire 20 est sélectionnée sur la base du signal de sélection de voie provenant des circuits de sélection de voie 22, et ces données sélectionnées sont envoyées à l'unité d'exécution interrogatrice. Par exemple, un MUX dynamique 24 peut être mis en œuvre pour fournir les données sélectionnées
25 pendant la première phase (la phase haute de l'horloge) de l'étage de pipeline L1D à l'unité d'exécution interrogatrice. Par exemple, le MUX 24 peut sortir les données sélectionnées pour l'unité en virgule flottante interrogatrice 26, une unité d'instructions LOI 28, ou l'unité d'exécution à base d'entiers LOD 30.

Dans un mode de réalisation préféré, une détermination est effectuée
30 pour savoir si une vraie coïncidence d'antémémoire est réalisée pour le niveau d'antémémoire. Ainsi, même si l'accès au(x) tableau(x) de données du niveau d'antémémoire commence avant qu'une détermination ne soit effectuée pour savoir si une vraie coïncidence d'antémémoire est réalisée pour le niveau d'antémémoire, cette détermination est effectuée ultérieurement. En se réfère-

rant maintenant à la Figure 5, elle montre un mode de réalisation préféré pour déterminer si une vraie coïncidence d'antémémoire est réalisée pour un niveau d'antémémoire. Comme montré, l'accès LIT 12 est terminé au cours de la première phase (la phase haute de l'horloge) de l'étage de pipeline LOD. Dans un mode de réalisation préféré, les données de correspondance d'étiquette de voie sorties par l'accès LIT 12 sont combinées avec le signal MESI 50 au cours de la deuxième phase (la phase basse de l'horloge) de LOD. Ainsi, dans un mode de réalisation préféré, un signal de vraie coïncidence/accès manqué est généré au cours de la deuxième phase de LOD en combinant les informations MESI avec la correspondance d'étiquette. Le signal de vraie coïncidence/accès manqué est entré dans les tableaux de données de L1 20 à l'étage de pipeline LOW. De plus, un signal de "code opération" valable est entré dans les tableaux de données de L1 20, qui indique si l'instruction reçue était une opération valable. Dans un mode de réalisation préféré, le signal de code opération valable est combiné avec le signal de vraie coïncidence/accès manqué pour produire un signal d'accès valable, qui est ensuite envoyé à l'unité d'exécution interrogatrice par l'intermédiaire du MUX 24 au cours de la première phase de LID. Par exemple, le MUX 24 peut envoyer le signal d'accès valable à une unité en virgule flottante 26, une unité d'instructions LOI 28, ou une unité d'exécution à base d'entiers LOD 30. Ce signal d'accès valable indique aux unités d'exécution que les données, dont l'accès a commencé en LOD, sont effectivement valables.

Il convient de comprendre que dans le mode de réalisation préféré entre tous, l'accès spéculatif aux données en antémémoire tel que décrit ici est mis en œuvre uniquement pour un chemin de chargement. C'est-à-dire, dans le mode de réalisation préféré entre tous, l'accès spéculatif aux données en antémémoire est utilisé uniquement pour les demandes d'accès en mémoire reçues souhaitant lire des données à partir de l'antémémoire. Ainsi, pour le mode de réalisation préféré entre tous, une détermination pour savoir si une vraie coïncidence a été réalisée pour le niveau d'antémémoire est effectuée avant l'accès aux données du niveau d'antémémoire pour un chemin de stockage (c'est-à-dire, pour une demande d'accès en mémoire reçue souhaitant écrire des données en antémémoire). Les lectures peuvent être effectuées de façon spéculative avec uniquement un risque de gaspiller de l'énergie en

accédant aux données du niveau d'antémémoire et en déterminant que le niveau d'antémémoire a produit un accès manqué (c'est-à-dire, que l'adresse demandée n'est pas dans le niveau d'antémémoire). Cependant, si une écriture est effectuée de façon spéculative, alors les données de l'antémémoire peuvent
5 être corrompues si un accès manqué se produit. Par conséquent, le mode de réalisation préféré entre tous réduit la latence nécessaire pour accéder aux données en antémémoire afin de satisfaire à une demande de lecture de données reçue.

De plus, dans le mode de réalisation préféré entre tous, plusieurs niveaux
10 d'une antémémoire à plusieurs niveaux peuvent faire l'objet d'un accès en parallèle. Par exemple, dans le mode de réalisation préféré entre tous, l'adresse physique du TLB 10 peut être envoyée à l'antémémoire de L2 pour commencer l'accès au(x) tableau(x) de données en antémémoire de L2 en parallèle avec le(s) tableau(x) de données en antémémoire de L1 20 à l'étage
15 de pipeline LOD. Par exemple, l'antémémoire peut être mise en œuvre de façon à accéder à plusieurs niveaux en parallèle de la manière exposée dans la demande de brevet américain, numéro de série [bordereau n° 10971421], intitulée "METHOD AND SYSTEM FOR EARLY TAG ACCESSES FOR LOWER-LEVEL CACHES IN PARALLEL WITH FIRST-LEVEL CACHE".
20 Ainsi, dans le mode de réalisation préféré entre tous, l'antémémoire peut être mise en œuvre de façon à se couvrir, de sorte que si l'antémémoire de L1 produit un accès manqué, un accès a déjà commencé vers l'antémémoire de L2. Si L1 produit une coïncidence, alors un signal peut être envoyé à l'antémémoire de L2, amenant l'antémémoire de L2 à abandonner son accès
25 aux données.

Dans le mode de réalisation préféré entre tous, la mise en œuvre des bancs décrite dans la demande de brevet américain, numéro de série [bordereau n° 10971178], intitulée "METHOD AND SYSTEM FOR PROVIDING A HIGH BANDWIDTH CACHE THAT ENABLES SIMULTA-
30 NEOUS READS AND WRITES WITHIN THE CACHE", peut être mise en œuvre dans cette architecture pour permettre une utilisation de bande passante accrue. C'est-à-dire, la bande passante accrue exposée dans "METHOD AND SYSTEM FOR PROVIDING A HIGH BANDWIDTH CACHE THAT ENABLES SIMULTANEOUS READS AND WRITES WITHIN

THE CACHE" peut être mise en œuvre avec le modèle à accès spéculatif aux données en antémémoire exposé ici pour augmenter encore les performances du système.

Comme évoqué ci-dessus, dans un mode de réalisation préféré, une antémémoire associative à huit voies est mise en œuvre. Cependant, il convient de comprendre que divers modes de réalisation peuvent être mis en œuvre avec un nombre quelconque de voies, et toute mise en œuvre de ce type est prévue pour tomber dans le domaine de la présente invention. En outre, il convient de comprendre qu'aucune pénalité de temps n'est infligée en effectuant l'accès spéculatif aux données d'un mode de réalisation préféré. C'est-à-dire, dans un mode de réalisation préféré, les données sont extraites en avance de façon spéculative sans qu'aucune pénalité de temps ne soit infligée si un accès manqué se produit. Si un accès manqué se produit sur le chemin d'un mode de réalisation préféré, un accès manqué se serait également produit sur le chemin série de l'art antérieur, mais sur le chemin série, les unités d'exécution ne seraient pas notifiées aussi rapidement d'un tel accès manqué. Il convient de comprendre également qu'une structure d'antémémoire de la présente invention peut être mise en œuvre dans tout type de système informatique disposant d'un processeur, y compris sans y être limité, un ordinateur individuel (PC), un ordinateur portable, et un organisateur (par exemple un ordinateur de poche).

Bien que la présente invention et ses avantages aient été décrits en détail, il convient de comprendre que diverses modifications, substitutions et altérations peuvent y être apportées sans s'éloigner de l'esprit et du domaine de l'invention tels que définis par les revendications ci-jointes. De plus, le domaine de la présente demande ne doit pas être limité aux modes de réalisation particuliers du processus, de la machine, de la fabrication, de la composition de matière, des moyens, des procédés et étapes décrits dans la spécification. Comme une personne normalement compétente dans l'art appréciera facilement d'après l'exposé de la présente invention, les processus, les machines, la fabrication, les compositions de matière, les moyens, les procédés ou étapes existant actuellement ou qui vont être développés ultérieurement, qui effectuent sensiblement la même fonction ou atteignent sensiblement le même résultat que les modes de réalisation correspondants

décrits ici, peuvent être utilisés selon la présente invention. Par conséquent, les revendications ci-jointes sont prévues pour être incluses dans le cadre du domaine de ces processus, ces machines, cette fabrication, ces compositions de matière, ces moyens, ces procédés ou ces étapes.

RE V E N D I C A T I O N S

1. Procédé d'accès en antémémoire, ledit procédé comprenant les étapes consistant à :

- 5 — entrer une demande d'accès en mémoire dans une structure d'antémémoire; et
- commencer un accès aux données de ladite structure d'antémémoire pour ladite demande d'accès en mémoire sans déterminer d'abord si une adresse en mémoire nécessaire pour satisfaire à ladite demande d'accès en mémoire est vraiment présente dans ladite structure d'antémémoire.

10 2. Système informatique comprenant :

- au moins un processeur qui exécute des instructions; et
- 15 — une structure d'antémémoire accessible par ledit processeur pour satisfaire aux demandes d'accès en mémoire, dans lequel ladite structure d'antémémoire est configurée pour commencer un accès aux données de ladite structure d'antémémoire (20) pour une demande d'accès en mémoire reçue sans déterminer d'abord si une adresse en mémoire nécessaire pour satisfaire à ladite demande d'accès en mémoire reçue est vraiment présente dans ladite structure d'antémémoire.

20 3. Structure d'antémémoire qui est accessible pour au moins un processeur informatique pour satisfaire aux demandes d'accès en mémoire pour des instructions exécutées par ledit au moins un processeur informatique, ladite structure d'antémémoire comprenant :

- des moyens (10) pour recevoir une demande d'accès en mémoire provenant d'au moins un processeur; et
- 25 — des moyens (14, 18, 20) pour commencer un accès aux données de ladite structure d'antémémoire pour une demande d'accès en mémoire reçue sans avoir déterminé d'abord si une adresse en mémoire nécessaire pour satisfaire à ladite demande d'accès en mémoire reçue est vraiment présente dans ladite structure d'antémémoire.

4. Procédé selon la revendication 1 ou système informatique selon la revendication 2, caractérisé en ce que ladite structure d'antémémoire est un niveau d'une antémémoire à plusieurs niveaux.

5 5. Procédé selon la revendication 4 comprenant en outre les étapes consistant à commencer un accès de données en parallèle pour plusieurs niveaux de ladite antémémoire à plusieurs niveaux sans avoir déterminé d'abord si une adresse en mémoire nécessaire pour satisfaire à ladite demande d'accès en mémoire est vraiment présente dans l'un quelconque desdits plusieurs niveaux de ladite antémémoire à plusieurs niveaux.

10 6. Procédé selon la revendication 1, système informatique selon la revendication 2 ou structure d'antémémoire selon la revendication 3, caractérisé(e) en ce que ladite demande d'accès en mémoire est une demande de lecture de données.

15 7. Procédé selon la revendication 1, comprenant en outre l'étape consistant à déterminer si l'adresse en mémoire nécessaire pour satisfaire à ladite demande d'accès en mémoire est vraiment présente dans ladite structure d'antémémoire.

20 8. Système informatique selon la revendication 2, caractérisé en ce que ladite structure d'antémémoire comprend en outre un TLB (10) qui reçoit une adresse virtuelle pour ladite demande d'accès en mémoire et sort une adresse physique correspondante.

25 9. Système informatique selon la revendication 2, caractérisé en ce que ladite antémémoire comprend en outre des circuits de correspondance d'étiquette (12) qui déterminent si une coïncidence d'étiquette est réalisée pour ladite structure d'antémémoire pour ladite demande d'accès en mémoire, et des circuits MESI (50) qui déterminent si ladite coïncidence d'étiquette est une vraie coïncidence d'étiquette.

30 10. Système informatique selon la revendication 9, caractérisé en ce que ladite antémémoire est mise en œuvre de telle sorte que lesdits circuits de décodage commencent leur exécution avant que lesdits circuits MESI ne déterminent si ladite coïncidence d'étiquette est une vraie coïncidence d'étiquette.

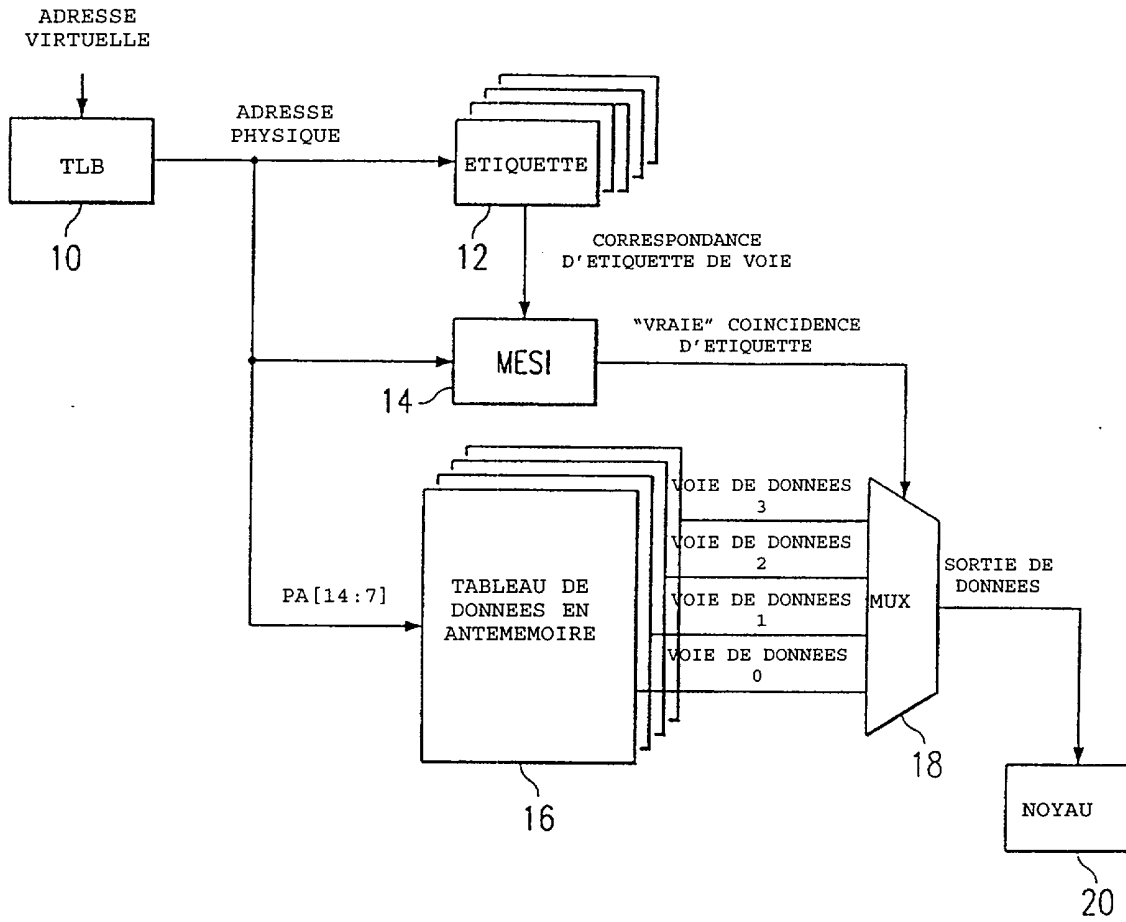


FIGURE 1
(ART ANTERIEUR)

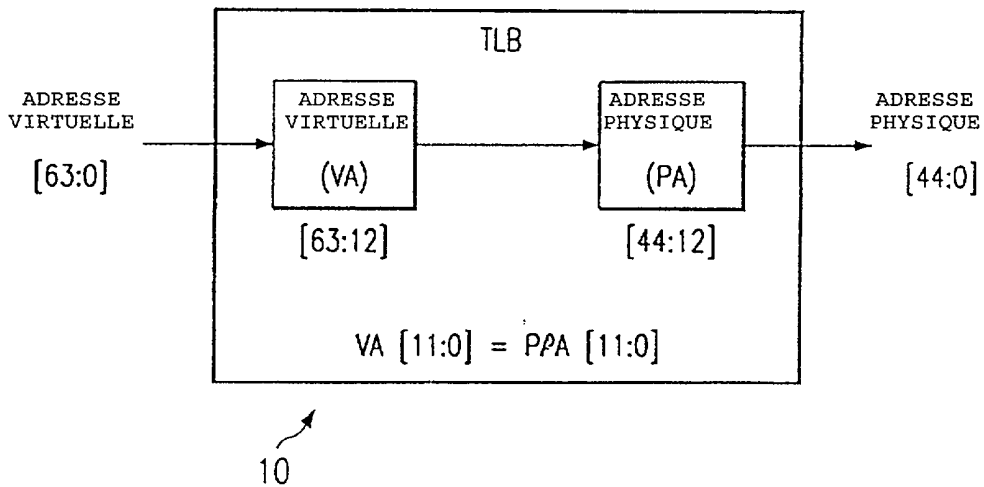


FIGURE 3
(ART ANTERIEUR)

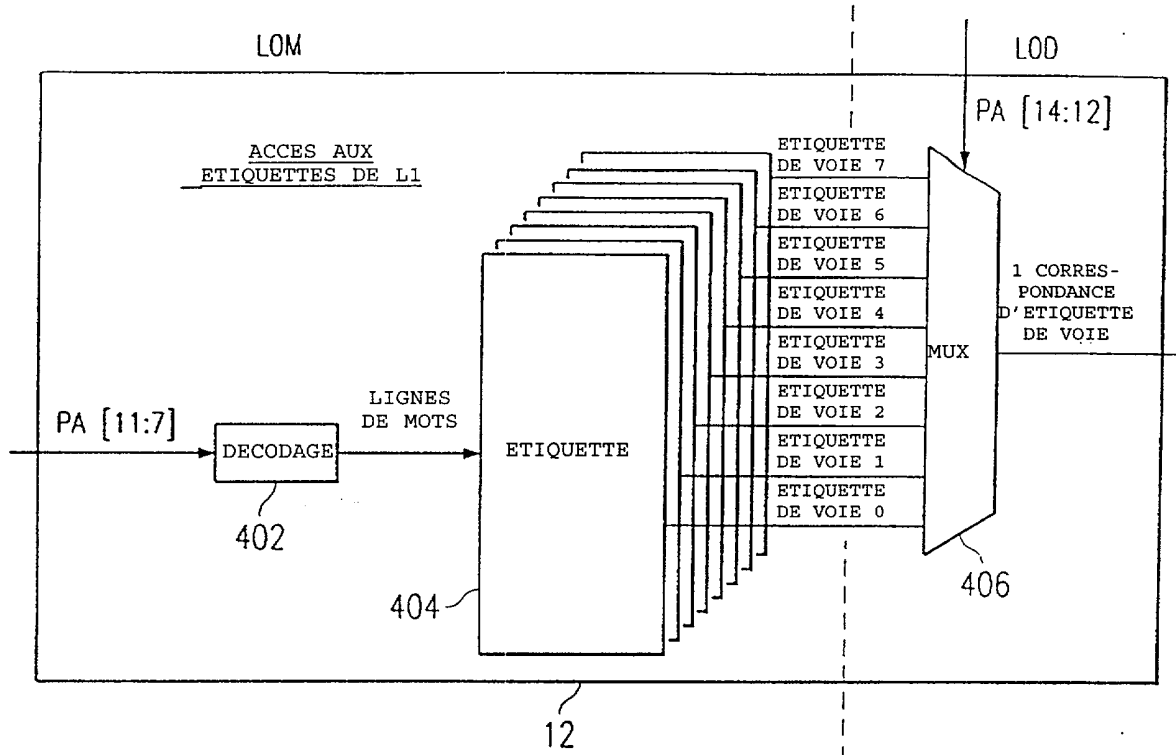


FIGURE 4

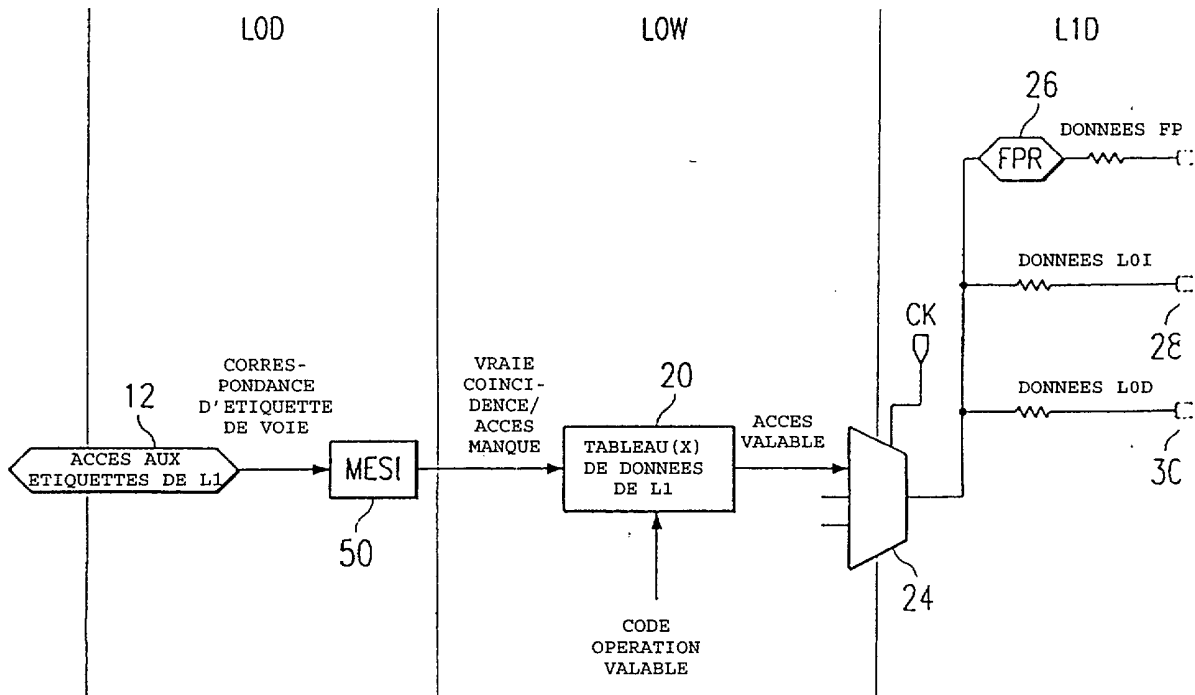


FIGURE 5