(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE

PARIS

(11) N° de publication : **2 733 065**
(à n'utiliser que pour les commandes de reproduction)

(21) N° d'enregistrement national : **95 10520**

(51) Int Cl⁶ : G 06 F 12/08

(12) DEMANDE DE BREVET D'INVENTION A1

(22) **Date de dépôt** : 08.09.95.

(30) **Priorité** : 11.04.95 KR 9508366.

(43) **Date de la mise à disposition du public de la demande :** 18.10.96 Bulletin 96/42.

(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) :** *LG SEMICON CO LTD* — KR.

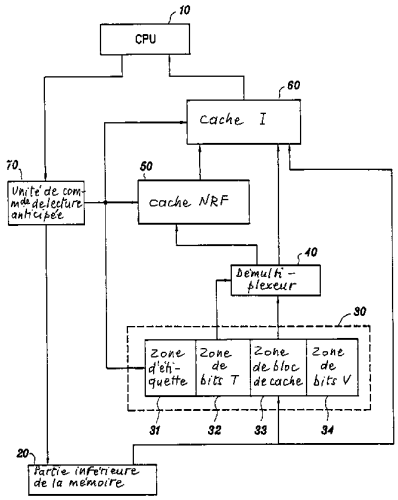
(72) **Inventeur(s) :** HAN TACK DON, PARK GI HO et KIM SHIN DUG.

(73) **Titulaire(s) :**

(74) **Mandataire :** CABINET BEAU DE LOMENIE.

(54) PROCEDE ET CIRCUIT DE LECTURE ANTICIPEE D'INSTRUTIONS/ DONNEES UTILISANT UN CACHE DE LECTURE ANTICIPEE NON CONSULTE.

(57) Un procédé et un circuit pour lecture anticipée d'instructions/données en utilisant un cache de lecture anticipée non consulté (50), pour stocker des blocs d'instructions/données faisant l'objet d'une lecture anticipée conformément à des mécanismes de lecture anticipée mais non consultés par l'unité centrale de traitement (10) dans une mémoire sur puce en tant que cache de lecture anticipée non consultés sans les écarter lorsqu'ils sont remplacés par de nouveaux blocs dans un tampon de lecture anticipée (30) de telle sorte qu'une consultation mémoire directe des blocs d'instructions/données de lecture anticipée non consultés puisse être obtenue lorsqu'ils doivent être consultés à des instants ultérieurs, sans la nécessité de recherche ou de lecture anticipée de ceux-ci dans la partie inférieure de la mémoire (20) à nouveau. Il est possible de diminuer le nombre d'accès manqués au cache et le temps d'attente mémoire et de réduire le trafic mémoire.



FR 2 733 065 - A1

ARRIERE-PLAN DE L'INVENTION

Domaine de l'invention

La présente invention concerne un procédé et un circuit de lecture anticipée d'instructions/données utilisant un cache de lecture anticipée non consulté (NRP) permettant non seulement de réduire le nombre d'accès manqués au cache et de diminuer le temps d'attente d'accès sur la partie inférieure de la mémoire mais permettant également de réduire le trafic mémoire en stockant des blocs lus de façon anticipée dans la partie inférieure de la mémoire mais non consultés par l'unité centrale de traitement (CPU) dans un cache NRP.

Description de l'art antérieur

Bien que les CPU voient leur performance progresser très rapidement, la progression de la performance des mémoires n'est pas aussi rapide que celle des CPU. Un tel écart de performance entre la CPU et la mémoire devient plus intense au fil du temps. Une hiérarchie de mémoires à construction efficiente constitue un facteur important qui affecte la performance globale du système d'ordinateur.

Afin d'obtenir un système de mémoire efficient, la plupart des systèmes d'ordinateur modernes utilisent une mémoire cache. Fondamentalement, une mémoire cache est conçue pour utiliser le groupement des consultations assuré lorsque l'ordinateur exécute un programme. Pendant l'exécution d'un programme dans un ordinateur, l'unité centrale de traitement (CPU) de l'ordinateur consulte la partie inférieure de la mémoire. Dans une certaine période temporelle limitée, ces consultations sont concentrées dans une certaine partie de la partie inférieure

de la mémoire. Si la CPU consulte un élément, il y a un risque que les éléments présentant des adresses adjacentes soient également consultés. Ceci est appelé "groupement spatial". Dans le cas de programmes généraux, plusieurs boucles sont exécutées, lesquelles nécessitent la plus grande part de la totalité du temps d'exécution des programmes. Dans une seule boucle, les mêmes instructions sont exécutées de façon répétée. En particulier, on observe souvent le cas selon lequel les instructions consultées présentement sont consultées à nouveau peu de temps après. Ceci est appelé "groupement temporel". Le principe de la mémoire cache consiste à utiliser ces groupements de consultations, c'est-à-dire à stocker les blocs utilisés fréquemment de la partie inférieure de la mémoire dans une mémoire présentant une vitesse de consultation très élevée interposée entre la partie inférieure de la mémoire et la CPU. Puisque la plupart des programmes d'application présentent les deux caractéristiques mentionnées ci-avant, la mémoire cache peut traiter 90% ou plus de la totalité des consultations dans la partie inférieure de la mémoire au moyen de la CPU même lorsqu'elle présente une capacité mémoire faible de quelques kilo-octets.

Lors de l'utilisation de la mémoire cache, si un bloc d'instruction que la CPU doit consulter existe dans la mémoire cache, la CPU se reporte aisément à ce bloc. Ce statut est appelé "présence de donnée recherchée dans cache". Par ailleurs, le statut selon lequel la mémoire cache ne comporte pas de bloc d'instruction à consulter par la CPU est appelé "accès manqué au cache". Un taux de présence est utilisé en tant que mesure de la performance de la mémoire cache. Le taux de présence est exprimé par l'équation suivante :

$$\text{Taux de présence} = \frac{\text{nombre de présences de données recherchées dans cache}}{\text{nombre total de consultations mémoire}}$$

Dans le même temps, les accès manqués au cache sont classifiés en trois types, à savoir l'accès manqué au cache inévitable, l'accès manqué par conflit et l'accès manqué par capacité. L'accès manqué inévitable est un accès manqué au cache se produisant lorsqu'un bloc est consulté initialement. L'accès manqué par conflit est un accès manqué au cache se produisant lorsqu'un bloc cartographié ou mappé à l'emplacement de mémoire cache correspondant mais remplacé ensuite par un autre bloc est consulté. Par ailleurs, l'accès manqué par capacité est un accès manqué au cache se produisant lorsque le jeu de travail, c'est-à-dire un jeu de pages fréquemment consulté par la CPU lors de l'exécution d'un programme d'application, présente une capacité plus importante que la mémoire cache.

Ces types d'accès manqués au cache présentent des taux de survenue différents par rapport à la totalité des accès manqués au cache lorsque la capacité de la mémoire cache croît. Bien que le nombre d'accès manqués inévitables soit approximativement constant indépendamment d'une variation de la capacité de la mémoire cache, les nombres d'accès manqués par conflit et d'accès manqués par capacité tendent à diminuer lorsque la capacité de la mémoire cache croît. Puisque la capacité de mémoire cache sur puce présente une tendance à la croissance du fait du développement d'une intégration à très grande échelle (VLSI), la proportion des accès manqués inévitables par rapport à la totalité des accès manqués au cache tend également à croître. Moyennant ces circonstances, des procédés permettant de réduire efficacement le nombre d'accès manqués inévitables deviennent plus importants du fait que la plupart des systèmes d'ordinateur modernes utilisent une mémoire cache sur puce à capacité importante.

Parmi les procédés permettant de réduire le nombre d'accès manqués inévitables, le plus simple consiste à augmenter la taille d'un bloc de cache. Ceci est dû au fait que davantage d'instructions sont amenées dans la mémoire cache

lors d'un accès manqué au cache et qu'en outre, les accès manqués au cache provoqués par un accès séquentiel de consultations mémoire peuvent être réduits avec un bloc de cache plus grand. Cependant, cette approche met en jeu une augmentation du cycle de la CPU consommé pour rechercher un bloc depuis la partie inférieure de la mémoire jusqu'à la mémoire cache ainsi qu'une augmentation du trafic mémoire. En outre, une pollution de cache peut se produire du fait de la taille importante du bloc de cache car la totalité du bloc doit être remplacée lors d'un accès manqué au cache même lorsque seulement une partie du bloc est consultée. Cette pollution de cache conduit à une dégradation de la performance. Du fait que la taille d'un bloc de cache doit être sélectionnée sur la base de caractéristiques d'architecture telles que le temps d'attente mémoire et que le débit de transfert, elle ne peut pas être simplement augmentée dans le but d'une lecture anticipée d'instruction.

Afin de surmonter les inconvénients mentionnés ci-avant, diverses techniques de lecture anticipée ont été proposées. Une lecture anticipée consiste à rechercher un bloc de mémoire dont on s'attend à ce qu'il soit consulté depuis la partie inférieure de la mémoire jusqu'à la partie supérieure de la mémoire avant une consultation du bloc mémoire par la CPU. La lecture anticipée séquentielle est la forme la plus simple de lecture anticipée. Lors d'une lecture anticipée séquentielle, le bloc suivant qui fait suite au bloc présentement consulté par la CPU fait l'objet d'une lecture anticipée. Cette technique de lecture anticipée séquentielle permet d'assurer une augmentation importante de la performance pour un programme d'application mettant en jeu une consultation mémoire moyennant une séquentialité importante. De façon générale, la consultation d'instructions présente une séquentialité importante par comparaison à la consultation de données. En vertu d'une telle caractéristique, la lecture anticipée séquentielle permet d'assurer une performance relativement supérieure. Par ailleurs, la lecture anticipée

séquentielle nécessite moins de composants câblés en dur supplémentaires pour atteindre son objectif de lecture anticipée. Lorsque la consultation d'instructions ne suit pas des flux d'exécution séquentiels, cependant, aucune amélioration de performance n'est attendue au moyen de la lecture anticipée séquentielle. En d'autres termes, la lecture anticipée séquentielle a pour inconvénient que le gain de performance obtenu au moyen de la lecture anticipée séquentielle n'est pas très élevé lorsque la consultation mémoire est exécutée suivant des flux d'exécution non séquentiels, comme dans le cas d'instructions de branchement conditionnel ou d'instructions de branchement inconditionnel.

Une lecture anticipée cible consiste à déterminer un bloc à lecture anticipée en utilisant une information d'historique concernant des flux de commande précédents stockée dans une table de prédiction. Cette lecture anticipée cible est basée sur le fait que des instructions de branchement, même s'il s'agit de branchements conditionnels, tendent à suivre les flux de commande précédents à nouveau dans la plupart des cas. Si un bloc qui fait suite à un bloc particulier a été consulté lors d'une exécution précédente, le bloc qui fait suite fera l'objet d'une lecture anticipée. Par ailleurs, si un bloc ne faisant pas suite à un bloc particulier a été consulté lors d'une exécution précédente, le bloc ne faisant pas suite fera l'objet d'une lecture anticipée. Par exemple, lorsqu'un bloc B a été consulté après le bloc A lors d'une exécution précédente, le bloc B fera l'objet d'une lecture anticipée suite à la consultation du bloc A à des instants ultérieurs. Une lecture anticipée cible présente une précision de lecture anticipée plus élevée qu'une lecture anticipée séquentielle du fait qu'elle utilise la caractéristique d'instructions de branchement. Cependant, il est difficile de s'attendre à une amélioration notable de la performance au moyen de la lecture anticipée cible du fait que des instructions de branchement ne suivent pas toujours les flux de commande précédents tout particulièrement dans les cas où la consultation

mémoire sur des instructions de branchement suit des flux de commande séquentiels et non séquentiels de façon alternée.

Une lecture anticipée hybride est un procédé de lecture anticipée permettant de réaliser une lecture anticipée à la fois des blocs séquentiels et non séquentiels afin de surmonter ces inconvénients. Mais la lecture anticipée hybride peut être utilisée dans les super-ordinateurs qui présentent moins de limitation de largeur de bande mémoire. Dans le cas de microprocesseurs, la lecture anticipée hybride n'a pas été considérée du fait de la largeur de bande limitée. Seulement une forme modifiée de la lecture anticipée hybride telle qu'un procédé de lecture anticipée pour exécuter une lecture anticipée sur la base de la priorité entre des blocs faisant l'objet d'une lecture anticipée peut être réalisée dans des systèmes basés sur microprocesseur. La priorité entre les blocs faisant l'objet d'une lecture anticipée peut être déterminée en utilisant une information concernant les flux de commande précédents. Cette lecture anticipée hybride est un mécanisme modifié par rapport aux mécanismes de lecture anticipée hybrides existants. Ce mécanisme sera décrit de manière davantage détaillée.

Lorsque la consultation mémoire a été exécutée pour suivre des flux de commande séquentiels lors de l'exécution précédente, le bloc qui fait suite au bloc présentement consulté (le bloc qui fait suite présente une adresse de bloc correspondant à "l'adresse du bloc courant + 1") est déterminé en tant que premier bloc des blocs candidats à la lecture anticipée. Par ailleurs, le bloc cible du bloc courant est déterminé en tant que second bloc des blocs candidats à la lecture anticipée. Par ailleurs, lorsque la consultation mémoire a été exécutée pour suivre des flux de commande non séquentiels lors de l'exécution précédente, le bloc cible du bloc courant est déterminé en tant que premier bloc des blocs candidats à la lecture anticipée. Par ailleurs, le bloc qui fait suite au bloc présentement consulté est déterminé en tant que second bloc des blocs candidats à la lecture anticipée. Après détermination des

premier et second blocs candidats à la lecture anticipée, une opération de lecture anticipée pour les blocs candidats déterminés est exécutée. Si le premier bloc candidat n'est pas résident dans la mémoire sur puce, il fait l'objet d'une lecture anticipée depuis la partie inférieure de la mémoire jusque dans la mémoire sur puce. Après la fin de cette lecture anticipée, l'opération de lecture anticipée pour les blocs candidats est terminée. Lorsque le premier bloc candidat est résident dans la mémoire sur puce, le second bloc candidat fait l'objet d'une lecture anticipée depuis la partie inférieure de la mémoire jusqu'à la mémoire sur puce s'il n'est pas résident dans la mémoire sur puce. Par ailleurs, lorsqu'à la fois les premier et second blocs candidats sont résidents dans la mémoire sur puce, aucune opération de lecture anticipée n'est exécutée. Par conséquent, le mécanisme de lecture anticipée hybride modifié permet de réaliser une lecture anticipée à la fois du bloc suivant et du bloc cible dans la plupart des cas. Cependant, ce schéma ne permet pas d'obtenir une amélioration importante de la performance. Bien que l'amélioration de la performance obtenue au moyen de la technique de lecture anticipée hybride existante ne soit pas si faible, la technique a pour inconvénient qu'elle peut être mise en oeuvre seulement dans des systèmes d'ordinateur qui présentent une restriction moindre sur la largeur de bande mémoire tels que des super-ordinateurs.

RESUME DE L'INVENTION

Par conséquent, un objet de l'invention consiste à proposer un procédé et un circuit pour une lecture anticipée d'instructions/données utilisant un cache de lecture anticipée non consulté, permettant de réaliser une lecture anticipée à la fois du bloc suivant et du bloc cible dans la plupart des cas même dans un microprocesseur présentant une largeur de bande mémoire limitée en stockant des blocs soumis à une lecture anticipée dans la partie inférieure de la mémoire mais non consultés par la CPU dans un cache NRP, ce qui permet non seulement de réduire le nombre d'accès manqués au cache et de

diminuer le temps d'attente d'accès à la partie inférieure de la mémoire mais aussi de réduire le trafic mémoire.

Selon un aspect, la présente invention propose un procédé de lecture anticipée d'instructions/données dans lequel des blocs d'instructions/données qui font l'objet d'une lecture anticipée conformément à un mécanisme de lecture anticipée d'instructions/données mais non consultés par une unité centrale de traitement sont stockés dans une mémoire sur puce sans être écartés suite à leur remplacement par de nouveaux blocs dans un tampon de lecture anticipée de telle sorte qu'ils puissent être utilisés pour une consultation mémoire à des instants ultérieurs.

Selon un autre aspect, la présente invention propose un circuit de lecture anticipée d'instructions/données comprenant : une unité centrale de traitement pour émettre en sortie divers signaux de commande requis pour l'exécution d'un programme nécessitant une consultation mémoire ; une partie inférieure de la mémoire pour stocker des blocs d'instructions/données requis pour l'exécution du programme par l'unité centrale de traitement ; une unité de commande de lecture anticipée pour commander une lecture anticipée de blocs d'instructions/données qui doivent faire l'objet d'une lecture anticipée au moyen de l'unité centrale de traitement ; un tampon de lecture anticipée pour stocker temporairement des blocs d'instructions/données faisant l'objet d'une lecture anticipée dans la partie inférieure de la mémoire et pour émettre en sortie des signaux de commande respectivement sur la base de si oui ou non les blocs d'instructions/données stockés ont été consultés par l'unité centrale de traitement ; un démultiplexeur pour démultiplexer des blocs d'instructions/données émis en sortie depuis le tampon de lecture anticipée respectivement selon ceux consultés par l'unité centrale de traitement et selon ceux non consultés par l'unité centrale de traitement sur la base des signaux de commande émis en sortie depuis le tampon de lecture anticipée ; une unité de stockage de bloc de lecture anticipée non consulté pour stocker les blocs d'instructions/données non

consultés par l'unité centrale de traitement qui sont émis en sortie depuis le démultiplexeur ; et un cache d'instruction pour stocker les blocs d'instructions/données émis en sortie depuis la partie inférieure de la mémoire, les blocs d'instructions/données émis en sortie depuis le démultiplexeur et des blocs d'instructions/données émis en sortie depuis l'unité de stockage de bloc de lecture anticipée non consulté.

Bien que la présente invention explique une lecture anticipée d'instructions, une lecture anticipée de données peut également être possible dans l'esprit et le cadre de la présente invention.

BREVE DESCRIPTION DES DESSINS

D'autres aspects et objets de l'invention apparaîtront à la lumière de la description qui suit des modes de réalisation par report aux dessins annexés parmi lesquels :

la figure 1 est un schéma fonctionnel d'un circuit de lecture anticipée d'instruction qui utilise un cache de lecture anticipée non consulté selon la présente invention ; et

la figure 2 est un schéma fonctionnel d'un circuit permettant de déterminer des blocs candidats à la lecture anticipée au moyen d'une unité de commande de lecture anticipée constituant une partie du circuit représenté sur la figure 1, le schéma fonctionnel pouvant être modifié aisément même dans le cas d'autres mécanismes de lecture anticipée que le mécanisme de lecture anticipée de la présente invention.

DESCRIPTION DETAILLEE DES MODES DE REALISATION PARTICULIERS

Par report à la figure 1, on peut voir un circuit de lecture anticipée d'instruction qui utilise un cache NRP selon la présente invention.

Comme représenté sur la figure 1, le circuit de lecture anticipée d'instruction inclut une CPU 10 pour exécuter diverses opérations de commande pour un ordinateur auquel le circuit de lecture anticipée d'instruction est appliqué, une partie inférieure de la mémoire 20 pour stocker des blocs d'instruction

à consulter par la CPU 10 et un tampon de lecture anticipée 30 pour stocker temporairement des blocs faisant l'objet d'une lecture anticipée dans la partie inférieure de la mémoire 20. Au tampon de lecture anticipée 30, un démultiplexeur 40 est couplé, lequel permet de démultiplexer des blocs émis en sortie depuis le tampon de lecture anticipée 30 respectivement sur la base de si oui ou non ils ont été consultés par la CPU 10. Le circuit de lecture anticipée d'instruction inclut en outre un cache NRP 50 pour stocker ceux des blocs non consultés par la CPU 10 qui sont émis en sortie depuis le démultiplexeur 40, un cache d'instruction (cache I) 60 pour stocker des blocs d'instruction consultés par la CPU 10 et une unité de commande de lecture anticipée 70 pour commander une opération de lecture anticipée sous la commande de la CPU 10.

Le tampon de lecture anticipée 30, le cache NRP 50 et le cache d'instruction 60 correspondent respectivement à trois emplacements de stockage sur puce différents d'une mémoire sur puce.

La partie inférieure de la mémoire 20 joue le rôle de mémoire principale dans le cas où l'ordinateur présente une hiérarchie de mémoires constituée par un système de mémoire cache à un seul niveau et en tant que mémoire cache de niveau inférieur ou que mémoire principale dans le cas où l'ordinateur présente une hiérarchie de mémoires constituée par un système de mémoire cache multi-niveaux. Bien que le circuit de lecture anticipée d'instruction soit représenté comme étant constitué par la CPU 10, par le tampon de lecture anticipée 30, par le cache NRP 50, par le cache d'instruction 60 et par l'unité de commande de lecture anticipée 70, tous ces éléments étant séparés les uns des autres, ces éléments constitutifs peuvent être intégrés ensemble sur une unique puce de processeur.

Le tampon de lecture anticipée 30 inclut une zone de blocs de cache 33 conçue pour stocker des blocs faisant l'objet d'une lecture anticipée dans la partie inférieure de la mémoire 20, une zone d'étiquette 31 conçue pour stocker des étiquettes dont

chacune est indicative de l'adresse de chacun correspondant des blocs stockés dans la zone de blocs de cache 33, une zone de bits T 32 conçue pour stocker des bits T dont chacun est indicatif de si oui ou non chacun correspondant des blocs stockés dans la zone de blocs de cache 33 a été consulté par la CPU 10, une zone de bits V 34 conçue pour stocker des bits V dont chacun est indicatif de si oui ou non le contenu de chacun correspondant des blocs stockés dans la zone de blocs de cache 33 est effectif.

L'unité de commande de lecture anticipée 70 permet de déterminer un bloc candidat à la lecture anticipée et elle recherche si oui ou non le bloc candidat déterminé existe dans la mémoire sur puce associée. L'unité de commande de lecture anticipée 70 exécute également une opération pour émettre en sortie une instruction de demande de lecture anticipée sur la partie inférieure de la mémoire 20 lorsque le bloc candidat déterminé n'existe pas dans la mémoire sur puce. En tant que partie de circuit pour exécuter l'opération pour déterminer des blocs candidats à la lecture anticipée, l'unité de commande de lecture anticipée 70 inclut une table de prédiction 71, un multiplexeur 75 et une unité de commande de table de prédiction 76, comme représenté sur la figure 2. Chaque entrée de la table de prédiction 71 inclut une unité d'adresse de bloc courant 72 conçue pour stocker l'adresse du bloc présentement consulté par la CPU 10, une unité d'adresse de bloc cible 73 conçue pour stocker l'adresse du bloc ne faisant pas suite consulté à droite après le bloc courant lors de l'exécution précédente et une zone d'information d'historique 74 conçue pour stocker l'information concernant si oui ou non plusieurs flux de commande, à savoir les consultations mémoire préalablement exécutées, sont séquentiels. Le multiplexeur 75 permet de sélectionner soit l'adresse qui résulte de l'addition de l'adresse émise en sortie depuis l'unité d'adresse de bloc courant 72 de la table de prédiction 71 et de l'unité soit l'adresse émise en sortie depuis l'unité d'adresse de bloc cible 73 sur la base de l'information stockée dans la zone d'information d'historique 74. L'unité de

commande de table de prédiction 76 permet d'accéder à la table de prédiction 71 et de la mettre à jour.

Le circuit de lecture anticipée d'instruction mentionné ci-avant qui utilise un cache NRP et le procédé mis en oeuvre qui utilise le circuit selon la présente invention peuvent être appliqués à diverses techniques de lecture anticipée. Tout d'abord, la description qui suit est menée en conjonction avec le cas où la présente invention est appliquée au mécanisme de lecture anticipée hybride pour la lecture anticipée à la fois du bloc qui fait suite et du bloc cible.

Lorsqu'un programme est exécuté, des blocs d'instruction requis pour l'exécution du programme sont consultés par la CPU 10. Une fois qu'une recherche d'un bloc A est terminée, un nouveau bloc est consulté par la CPU 10. Le bloc nouvellement consulté peut être le bloc faisant suite ou le bloc ne faisant pas suite conformément au programme. Dans la description qui suit, le bloc faisant suite et le bloc ne faisant pas suite sont décrits respectivement en tant que bloc B et que bloc C.

Dans le cas où le bloc faisant suite B a été consulté à droite après le bloc A, la mise à jour de la table de prédiction 71 est exécutée. Pour la mise à jour de la table de prédiction 71, l'unité de commande de table de prédiction 76 cherche si oui ou non une information concernant le bloc A existe dans la table de prédiction 71. Lorsqu'aucune information concernant le bloc A n'existe dans la table de prédiction 71, la table de prédiction 71 n'est pas mise à jour. Ceci est dû au fait que le bloc suivant B qui fait suite au bloc A est déterminé par défaut en tant que bloc à soumettre à une lecture anticipée si une entrée correspondante pour le bloc A n'existe pas dans la table de prédiction 71. Par ailleurs, si une information concernant le bloc A existe dans la table de prédiction 71, la zone d'information d'historique 74 est établie comme étant séquentielle.

Après la fin de la mise à jour de la table de prédiction 71, l'unité de commande de table de prédiction 76 détermine des blocs candidats à la lecture anticipée. Cette détermination est

réalisée en utilisant l'information stockée dans la table de prédiction 71 concernant le bloc B présentement consulté par la CPU 10. Si le flux de commande consulté par la CPU 10 à droite après le bloc B lors de l'exécution précédente est séquentiel, le bloc qui fait suite au bloc B (le bloc qui fait suite présente l'adresse qui résulte de l'addition de l'adresse courante du bloc courant B et de l'unité) est déterminé en tant que premier bloc candidat de lecture anticipée. Le bloc cible du bloc courant B, c'est-à-dire le bloc correspondant à l'adresse de bloc stockée dans l'unité d'adresse de bloc cible 73 de l'entrée de table de prédiction pour le bloc B, est déterminé en tant que second bloc candidat de lecture anticipée. Si l'entrée de table de prédiction correspondante pour le bloc B n'existe pas dans la table de prédiction 71, il n'y a aucune information pour le bloc cible du bloc B de telle sorte que le bloc qui fait suite au bloc B est déterminé en tant que seul bloc candidat de lecture anticipée. Par ailleurs, lorsque le flux de commande consulté par la CPU 10 à droite après le bloc B lors de l'exécution précédente est non séquentiel, le bloc cible du bloc courant B, c'est-à-dire le bloc correspondant à l'adresse de bloc stockée dans l'unité d'adresse de bloc cible 73 de l'entrée de table de prédiction correspondant au bloc B sur la base de la sélection du multiplexeur 75, est déterminé en tant que premier bloc candidat de lecture anticipée. Dans ce cas, le bloc qui fait suite au bloc courant B est déterminé en tant que second bloc candidat de lecture anticipée.

Dans le cas où le bloc ne faisant pas suite C a été consulté par la CPU 10 à droite après le bloc A, l'unité de commande de table de prédiction 76 recherche si oui ou non l'information concernant le bloc A existe dans la table de prédiction 71 pour la mise à jour de la table de prédiction 71. Lorsqu'aucune information concernant le bloc A n'existe dans la table de prédiction 71, l'unité de commande de table de prédiction 76 alloue une entrée pour le bloc A dans la table de prédiction 71. Ensuite, l'adresse du bloc A est stockée dans

l'unité d'adresse de bloc courant 72 de l'entrée de table de prédiction. Par ailleurs, l'adresse du bloc C est stockée dans une entrée de table de prédiction de l'unité d'adresse de bloc cible 73 de l'entrée de table de prédiction. Pour finir, la zone d'information d'historique 74 de l'entrée de table de prédiction est établie comme étant non séquentielle. Cependant, lorsqu'une information concernant le bloc A existe dans la table de prédiction 71, l'adresse du bloc C est stockée dans l'unité d'adresse de bloc cible 73 de l'entrée de table de prédiction correspondante. Dans ce cas, la zone d'information d'historique 74 de l'entrée de table de prédiction correspondante est établie comme étant non séquentielle.

Après la fin de la mise à jour de la table de prédiction 71, l'unité de commande de table de prédiction 76 détermine des blocs candidats à la lecture anticipée. Cette détermination est réalisée en utilisant l'information stockée dans la table de prédiction 71 concernant le bloc C du fait que le bloc présentement consulté par la CPU 10 est le bloc C. La détermination de blocs candidats est réalisée sur la base de l'information concernant le bloc C stockée dans la zone d'information d'historique 74 de l'entrée de table de prédiction correspondante, c'est-à-dire si le flux de commande consulté par la CPU 10 à droite après le bloc C est séquentiel ou non séquentiel, d'une manière similaire à ce qui est exécuté pour le bloc B.

Après la fin de la mise à jour de la table de prédiction 71, l'opération de lecture anticipée est exécutée. La lecture anticipée est exécutée tout d'abord pour le premier bloc candidat de lecture anticipée. Pour cette lecture anticipée, une recherche est réalisée pour savoir si oui ou non le premier bloc candidat de lecture anticipée existe au niveau de l'un des emplacements de la mémoire sur puce, c'est-à-dire soit le tampon de lecture anticipée 30, le cache NRP 50 et le cache d'instruction 60. Si le premier bloc candidat de lecture anticipée n'existe pas en un quelconque emplacement de la mémoire sur puce, l'unité de

commande de lecture anticipée 70 lit de façon anticipée le premier bloc candidat de lecture anticipée dans la partie inférieure de la mémoire 10 puis stocke le bloc faisant l'objet de la lecture anticipée dans le tampon de lecture anticipée 30. Par conséquent, l'opération de lecture anticipée est terminée. Cependant, si le premier bloc candidat de lecture anticipée existe dans la mémoire sur puce, l'unité de commande de lecture anticipée 70 tente d'exécuter une lecture anticipée pour le second bloc candidat de lecture anticipée de la même manière que mentionné ci-avant. Lorsqu'à la fois les premier et second blocs candidats de lecture anticipée existent dans la mémoire sur puce, aucune lecture anticipée n'est exécutée.

Lorsque le nombre de blocs faisant l'objet d'une lecture anticipée dans le tampon de lecture anticipée 30 excède la capacité du tampon de lecture anticipée 30, les blocs anciens stockés dans le tampon de lecture anticipée 30 sont remplacés par des blocs faisant nouvellement d'une lecture anticipée dans la partie inférieure de la mémoire 20 conformément au mécanisme de remplacement de bloc du tampon de lecture anticipée 30 lui-même. Dans le même temps, la zone de bits T 32 du tampon de lecture anticipée 30 représente le bit d'état présentant une valeur d'établissement ou une valeur de remise à l'état initial. Chaque bit T présente la valeur d'établissement lorsqu'un bloc correspondant n'a pas été consulté par la CPU 10 et présente la valeur de remise à l'état initial lorsque le bloc correspondant a été consulté par la CPU 10. Par conséquent, un bloc consulté par la CPU 10 est simplement écarté suite à l'exécution du remplacement de bloc dans le tampon de lecture anticipée 30. Cependant, un bloc non consulté par la CPU 10 est déplacé jusque dans le cache NRP 50 du fait que la valeur de bit T correspondante provenant de la zone de bits T 32 est appliquée sur le démultiplexeur 40 en tant que signal de commande. Dans des schémas classiques, ces blocs faisant l'objet d'une lecture anticipée non consultés par la CPU sont écartés. Puisque les blocs faisant l'objet d'une lecture anticipés non consultés par la

CPU sont stockés dans le cache NRP 50 sans être écartés conformément à la présente invention, il est possible d'obtenir une consultation directe des blocs NRP lorsqu'ils doivent être consultés par la CPU 10 à des instants ultérieurs sans une quelconque nécessité de les chercher dans la partie inférieure de la mémoire 20 à nouveau.

Le procédé de lecture anticipée d'instruction qui utilise le cache NRP selon la présente invention permet d'obtenir une amélioration supérieure de la performance lorsqu'il est appliqué à un schéma basé sur une combinaison du mécanisme de lecture anticipée hybride et du mécanisme de lecture anticipée par projection. La lecture anticipée par projection est un mécanisme de lecture anticipée qui lit de façon anticipée le d-ième bloc dont on peut s'attendre à ce qu'il soit consulté après le bloc courant. Le d est appelé degré de lecture anticipée par projection. Par exemple, si la CPU 10 a consulté des blocs D, E, F suite à une exécution précédente, le bloc F au lieu du bloc E fera l'objet d'une lecture anticipée suite à la consultation du bloc D à des instants ultérieurs lors de la lecture anticipée par projection lorsque le degré de lecture anticipée par projection vaut deux.

Lorsque la CPU 10 a consulté un bloc faisant suite après le bloc courant, par exemple le bloc G lors de l'exécution précédente conformément au procédé de lecture anticipée par projection, le bloc I (l'adresse du bloc courant G + 2) sera le bloc à lecture anticipée suite à une consultation du bloc G à des instants ultérieurs. Cependant, un accès manqué au cache peut se produire lors de la consultation mémoire séquentielle si le bloc H (l'adresse du bloc courant G + 1) n'existe pas dans la mémoire sur puce. Afin d'empêcher un tel accès manqué au cache, le bloc H est déterminé en tant que premier bloc candidat de lecture anticipée même si le bloc à soumettre à la lecture anticipée originellement est le bloc I. Dans ce cas, le bloc I à soumettre à la lecture anticipée originellement est déterminé en tant que second bloc candidat de lecture anticipée. Le bloc cible du bloc

courant G est déterminé en tant que troisième bloc candidat de lecture anticipée. Par ailleurs, le bloc à droite avant le bloc cible est déterminé en tant que quatrième bloc candidat de lecture anticipée. Le bloc à droite avant le bloc cible présente une adresse qui résulte de la soustraction d'une unité à l'adresse du bloc cible (l'adresse du bloc cible - 1).

Lorsque la CPU 10 consulte un bloc ne faisant pas suite après le bloc courant lors de l'exécution précédente conformément au procédé de lecture anticipée par projection, le bloc cible du bloc courant est déterminé en tant que premier bloc candidat de lecture anticipée. En tant que second bloc candidat de lecture anticipée, le bloc à droite avant le bloc cible (l'adresse du bloc cible - 1) est déterminé. Ceci est dû au fait que le cas non séquentiel présente une probabilité plus faible que le bloc à droite avant le bloc cible soit consulté avant la consultation du bloc cible, par comparaison avec le cas séquentiel expliqué ci-avant. Par ailleurs, le bloc à droite après le bloc courant (l'adresse du bloc courant + 1) est déterminé en tant que troisième bloc candidat de lecture anticipée. Par ailleurs, le second bloc après le bloc courant (l'adresse du bloc courant + 2) est déterminé en tant que quatrième bloc candidat de lecture anticipée.

Après la fin de la détermination des blocs candidats de lecture anticipée, la lecture anticipée est exécutée. Tout d'abord, la lecture anticipée portant sur le premier bloc candidat de lecture anticipée est tentée. Si le premier bloc candidat de lecture anticipée n'existe pas dans la mémoire sur puce, le bloc candidat fait l'objet d'une lecture anticipée dans la partie inférieure de la mémoire. Par conséquent, la lecture anticipée est terminée. Cependant, si le premier bloc candidat de lecture anticipée existe déjà dans la mémoire sur puce, la lecture anticipée portant sur le second bloc candidat de lecture anticipée est ensuite tentée. Si le second bloc candidat de lecture anticipée n'existe pas dans la mémoire sur puce, le bloc candidat fait l'objet d'une lecture anticipée dans la partie

inférieure de la mémoire et ainsi, la lecture anticipée est terminée. Lorsque le second bloc candidat de lecture anticipée existe déjà dans la mémoire sur puce, une lecture anticipée portant sur le troisième bloc candidat de lecture anticipée est tentée. Si le troisième bloc candidat de lecture anticipée n'existe pas dans la mémoire sur puce, le bloc candidat fait l'objet d'une lecture anticipée dans la partie inférieure de la mémoire et ainsi, la lecture anticipée est terminée. Lorsque le troisième bloc candidat de lecture anticipée existe déjà dans la mémoire sur puce, la lecture anticipée portant sur le quatrième bloc candidat de lecture anticipée est tentée. Si le quatrième bloc candidat de lecture anticipée n'existe pas dans la mémoire sur puce, le bloc candidat fait l'objet d'une lecture anticipée dans la partie inférieure de la mémoire et ainsi, la lecture anticipée est terminée. Par ailleurs, si le quatrième bloc candidat de lecture anticipée existe déjà dans la mémoire sur puce, la lecture anticipée portant sur le quatrième bloc candidat de lecture anticipée n'est par initiée. En d'autres termes, aucune requête de lecture anticipée réelle n'est initiée lorsque tous les blocs candidats de lecture anticipée sont déjà résidents dans la mémoire sur puce.

Lorsque le nombre de blocs faisant l'objet d'une lecture anticipée dans le tampon de lecture anticipée 30 excède la capacité du tampon de lecture anticipée 30 selon le schéma de lecture anticipée par projection auquel la présente invention est appliquée, les blocs anciens stockés dans le tampon de lecture anticipée 30 sont remplacés par des blocs faisant nouvellement l'objet d'une lecture anticipée dans la partie inférieure de la mémoire 20 conformément au mécanisme de remplacement de bloc du tampon de lecture anticipée 30 lui-même. Dans le même temps, la zone de bits T 32 du tampon de lecture anticipée 30 est le bit d'état présentant une valeur d'établissement ou une valeur de remise à l'état initial. Chaque bit T présente la valeur d'établissement si le bloc correspondant n'a pas été consulté par la CPU 10 et présente la valeur de remise à l'état initial si le

bloc correspondant a été consulté par la CPU 10. Par conséquent, les blocs consultés par la CPU 10 sont simplement écartés suite à l'exécution du remplacement de bloc dans le tampon de lecture anticipée 30. Cependant, les blocs non consultés par la CPU 10 sont déplacés jusque dans le cache NRP 50 du fait que la valeur du bit T correspondant provenant de la zone de bits T 32 est appliquée sur le démultiplexeur 40 en tant que signal de commande. Selon des schémas classiques, ces blocs faisant l'objet d'une lecture anticipée non consultés par la CPU sont écartés comme mentionné ci-avant. Puisque les blocs faisant l'objet d'une lecture anticipée non consultés par la CPU sont stockés dans le cache NRP 50 sans être écartés conformément à la présente invention, il est possible d'obtenir une consultation directe des blocs NRP lorsqu'ils doivent être consultés par la CPU 10 à des instants ultérieurs sans une quelconque nécessité de les rechercher dans la partie inférieure de la mémoire 20 à nouveau.

Comme il ressort de la description qui précède, la présente invention propose un procédé et un circuit permettant une lecture anticipée des instructions en utilisant un cache NRP, conçu pour stocker des blocs d'instruction faisant l'objet d'une lecture anticipée conformément à une certaine variété de mécanismes de lecture anticipée existants mais non consultés par la CPU dans une mémoire sur puce en tant que cache NRP sans les écarter lorsqu'ils sont remplacés par de nouveaux dans un tampon de lecture anticipée de telle sorte qu'une consultation mémoire directe des bloc NRP peut être obtenue lorsqu'ils doivent être consultés à des instants ultérieurs, sans une quelconque nécessité de les rechercher dans la partie inférieure de la mémoire à nouveau. Par conséquent, il est possible non seulement de diminuer le nombre d'accès manqués au cache et le temps d'attente mémoire dû à la recherche d'instructions dans la partie inférieure de la mémoire pour la consultation des instructions, mais il est également possible de réduire le trafic mémoire.

Bien que des modes de réalisation particuliers de l'invention aient été décrits à des fins d'illustration, l'homme de l'art appréciera que diverses modifications, ajouts et substitutions soient possibles sans que l'on s'écarte ni du cadre ni de l'esprit de l'invention telle que définie présentement.

REVENDICATIONS

1. Procédé de lecture anticipée d'instruction caractérisé en ce que des blocs d'instruction qui font l'objet d'une lecture anticipée conformément à un mécanisme de lecture anticipée d'instruction mais non consultés par une unité centrale de traitement (10) sont stockés dans une mémoire sur puce sans être écartés suite à leur remplacement par de nouveaux blocs dans un tampon de lecture anticipée (30) de telle sorte qu'ils puissent être utilisés pour une consultation mémoire à des instants ultérieurs.

2. Procédé de lecture anticipée d'instruction selon la revendication 1, caractérisé en ce que le mécanisme de lecture anticipée d'instruction est un mécanisme de lecture anticipée hybride.

3. Procédé de lecture anticipée d'instruction selon la revendication 1, caractérisé en ce que le mécanisme de lecture anticipée d'instruction est un mécanisme de lecture anticipée par projection.

4. Procédé de lecture anticipée d'instruction selon la revendication 1, caractérisé en ce que le mécanisme de lecture anticipée d'instruction est une combinaison d'une pluralité de mécanismes de lecture anticipée d'instruction.

5. Procédé de lecture anticipée d'instruction selon la revendication 1, caractérisé en ce que la mémoire sur puce est une mémoire cache (50).

6. Procédé de lecture anticipée de données, caractérisé en ce que des blocs de données faisant l'objet d'une lecture anticipée conformément à un mécanisme de lecture anticipée de données mais non consultés par une unité centrale de traitement sont stockés dans une mémoire sur puce sans être écartés suite à leur remplacement par de nouveaux blocs dans un tampon de lecture anticipée de telle sorte qu'ils puissent être utilisés pour une consultation mémoire à des instants ultérieurs.

7. Procédé de lecture anticipée de données selon la revendication 6, caractérisé en ce que la mémoire sur puce est une mémoire cache.

8. Circuit de lecture anticipée d'instruction, caractérisé en ce qu'il comprend :

une unité centrale de traitement (10) pour émettre en sortie divers signaux de commande requis pour l'exécution d'un programme nécessitant une consultation mémoire ;

une partie inférieure de la mémoire (20) pour stocker des blocs d'instruction requis pour l'exécution du programme par l'unité centrale de traitement ;

une unité de commande de lecture anticipée (70) pour commander une lecture anticipée de blocs d'instruction qui doivent faire l'objet d'une lecture anticipée au moyen de l'unité centrale de traitement ;

un tampon de lecture anticipée (30) pour stocker temporairement des blocs d'instruction faisant l'objet d'une lecture anticipée dans la partie inférieure de la mémoire et pour émettre en sortie des signaux de commande respectivement sur la base de si oui ou non les blocs d'instruction stockés ont été consultés par l'unité centrale de traitement ;

un démultiplexeur (40) pour démultiplexer des blocs d'instruction émis en sortie depuis le tampon de lecture anticipée respectivement selon ceux consultés par l'unité centrale de traitement et selon ceux non consultés par l'unité centrale de traitement sur la base des signaux de commande émis en sortie depuis le tampon de lecture anticipée ;

une unité de stockage de bloc de lecture anticipée non consulté pour stocker les blocs d'instruction non consultés par l'unité centrale de traitement qui sont émis en sortie depuis le démultiplexeur ; et

un cache d'instruction (60) pour stocker les blocs d'instruction émis en sortie depuis la partie inférieure de la mémoire, les blocs d'instruction émis en sortie depuis le

démultiplexeur et des blocs d'instruction émis en sortie depuis l'unité de stockage de bloc de lecture anticipée non consulté.

9. Circuit de lecture anticipée d'instruction selon la revendication 8, caractérisé en ce que l'unité de stockage de bloc de lecture anticipée non consulté comprend une mémoire cache (50).

10. Circuit de lecture anticipée d'instruction selon la revendication 8, caractérisé en ce que le tampon de lecture anticipée comprend :

une zone de bloc de cache (33) conçue pour stocker chacun des blocs d'instruction faisant l'objet d'une lecture anticipée dans la partie inférieure de la mémoire ;

une zone de bits V (34) conçue pour indiquer si oui ou non le bloc d'instruction stocké dans la zone de bloc de cache est effectif ;

une zone d'étiquette (31) conçue pour indiquer l'adresse du bloc d'instruction stocké dans la zone de bloc de cache ; et

une zone de bits T (32) conçue pour indiquer si oui ou non le bloc d'instruction stocké dans la zone de bloc de cache a été consulté par l'unité centrale de traitement.

11. Circuit de lecture anticipée d'instruction, caractérisé en ce qu'il comprend :

une unité centrale de traitement pour émettre en sortie divers signaux de commande requis pour l'exécution d'un programme nécessitant une consultation mémoire ;

une partie inférieure de la mémoire pour stocker des blocs de données requis pour l'exécution du programme par l'unité centrale de traitement ;

une unité de commande de lecture anticipée pour commander une lecture anticipée de blocs de données qui doivent faire l'objet d'une lecture anticipée par l'unité centrale de traitement ;

un tampon de lecture anticipée pour stocker temporairement des blocs de données faisant l'objet d'une lecture anticipée dans la partie inférieure de la mémoire et pour

émettre en sortie des signaux de commande respectivement sur la base de si oui ou non les blocs de données stockés ont été consultés par l'unité centrale de traitement ;

un démultiplexeur pour démultiplexer des blocs de données émis en sortie depuis le tampon de lecture anticipée respectivement selon ceux consultés par l'unité centrale de traitement et selon ceux non consultés par l'unité centrale de traitement sur la base des signaux de commande émis en sortie depuis le tampon de lecture anticipée ;

une unité de stockage de bloc de lecture anticipée non consulté pour stocker les blocs de données émis en sortie depuis le démultiplexeur non consultés par l'unité centrale de traitement ; et

un cache de données pour stocker les blocs de données émis en sortie depuis la partie inférieure de la mémoire, les blocs de données émis en sortie depuis le démultiplexeur et des blocs de données émis en sortie depuis l'unité de stockage de bloc de lecture anticipée non consulté.

12. Circuit de lecture anticipée de données selon la revendication 11, caractérisé en ce que l'unité de stockage de bloc de lecture anticipée consulté comprend une mémoire cache.

13. Circuit de lecture anticipée de données selon la revendication 11, caractérisé en ce que le tampon de lecture anticipée comprend :

une zone de bloc de cache conçue pour stocker chacun des blocs de données faisant l'objet d'une lecture anticipée dans la partie inférieure de la mémoire ;

une zone de bits V conçue pour indiquer si oui ou non le bloc de données stocké dans la zone de bloc de cache est effectif ;

une zone d'étiquette conçue pour indiquer l'adresse du bloc de données stocké dans la zone de bloc de cache ; et

une zone de bits T conçue pour indiquer si oui ou non le bloc de données stocké dans la zone de bloc de cache a été consulté par l'unité centrale de traitement.

FIG. 1

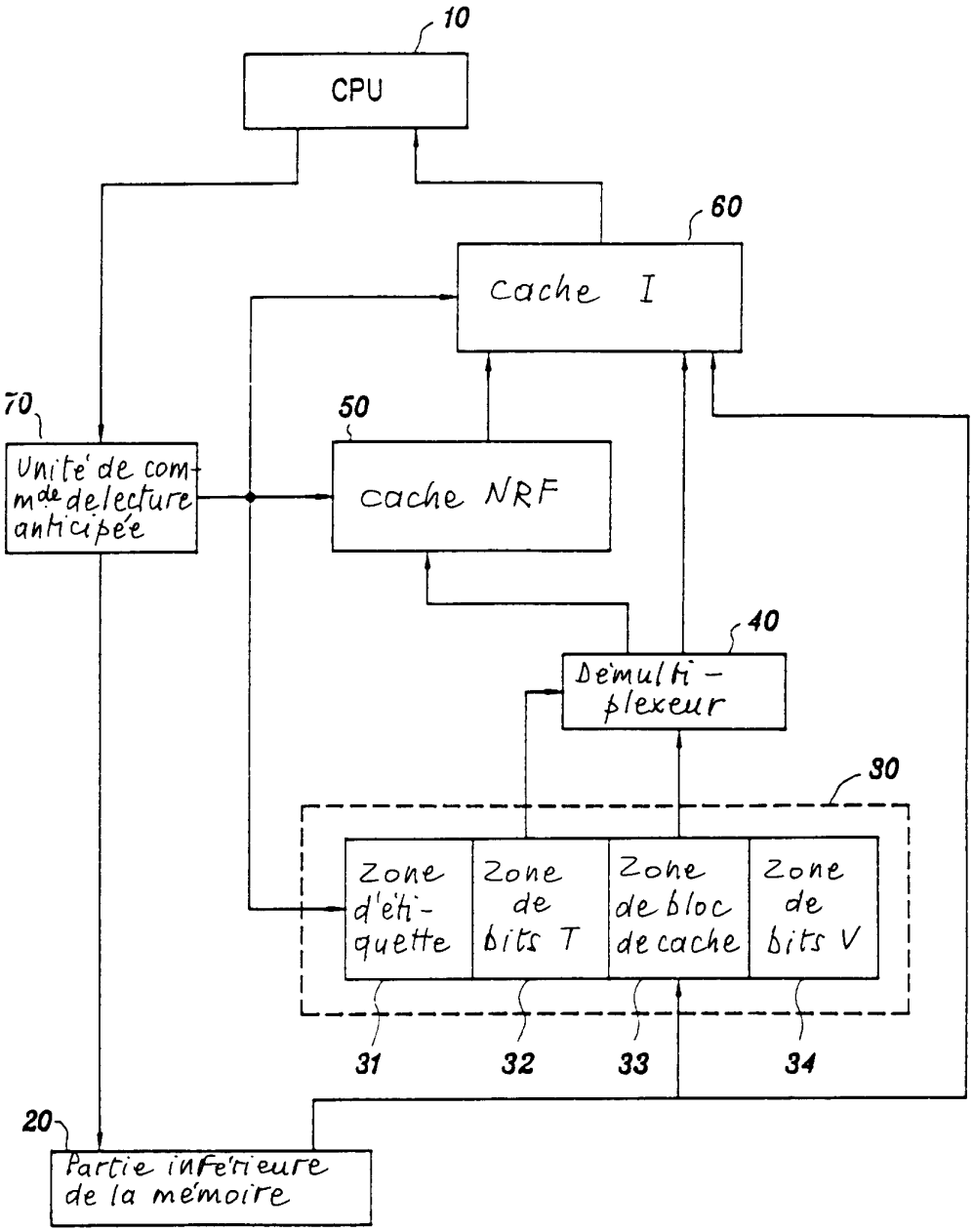
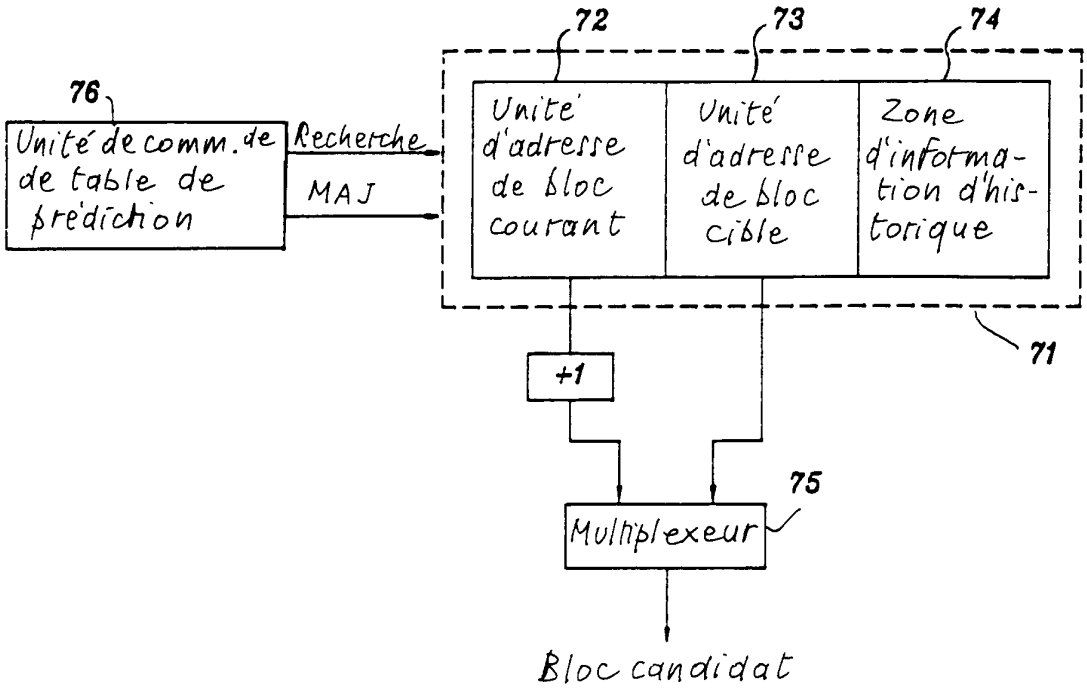

10
CPU
60
Cache I
70
Unité de com- mde de lecture anticipée
50
cache NRF
40
Démulti- plexeur
30
Zone d'éti- quette
Zone de bits T
Zone de bloc de cache
Zone de bits V
31
32
33
34
20
Partie inférieure de la mémoire

FIG. 2


72
73
74
76
Unité de comm. de table de prédiction
Recherche
MAJ
Unité d'adresse de bloc courant
Unité d'adresse de bloc cible
Zone d'information d'historique
71
+1
75
Multiplexeur
Bloc candidat