

(12) DEMANDE INTERNATIONALE PUBLIÉE EN VERTU DU TRAITÉ DE COOPÉRATION
EN MATIÈRE DE BREVETS (PCT)

(19) Organisation Mondiale de la Propriété
Intellectuelle
Bureau international



(43) Date de la publication internationale
11 juin 2009 (11.06.2009)

PCT

(10) Numéro de publication internationale
WO 2009/071791 A2

(51) Classification internationale des brevets :
G06F 21/02 (2006.01)

WORLD INTERNATIONAL N.V. [BE/BE]; Excelsior-
laan 44-46, B-1930 Zaventem (BE).

(21) Numéro de la demande internationale :
PCT/FR2008/052073

(72) Inventeurs; et

(22) Date de dépôt international :
18 novembre 2008 (18.11.2008)

(75) Inventeurs/Déposants (pour US seulement) : **ROMAIN, Fabrice** [FR/FR]; Les Plantiers, F-83560 Rians (FR). **MO-DAVE, Jean-Louis** [BE/BE]; 56, rue du Petit Ry, B-1340 Ottignies (BE).

(25) Langue de dépôt : français

(26) Langue de publication : français

(74) Mandataire : **CABINET BEAUMONT**; 1, rue Champol-
lion, F-38000 Grenoble (FR).

(30) Données relatives à la priorité :
0759136 19 novembre 2007 (19.11.2007) FR

(71) Déposants (pour tous les États désignés sauf US) :
STMICROELECTRONICS SA [FR/FR]; 29, boulevard
Romain Rolland, F-92120 Montrouge (FR). **PROTON**

(81) États désignés (sauf indication contraire, pour tout titre de
protection nationale disponible) : AE, AG, AL, AM, AO,
AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH,
CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG,

[Suite sur la page suivante]

(54) Title: VERIFICATION OF DATA READ IN MEMORY

(54) Titre : VERIFICATION DE DONNEES LUES EN MEMOIRE

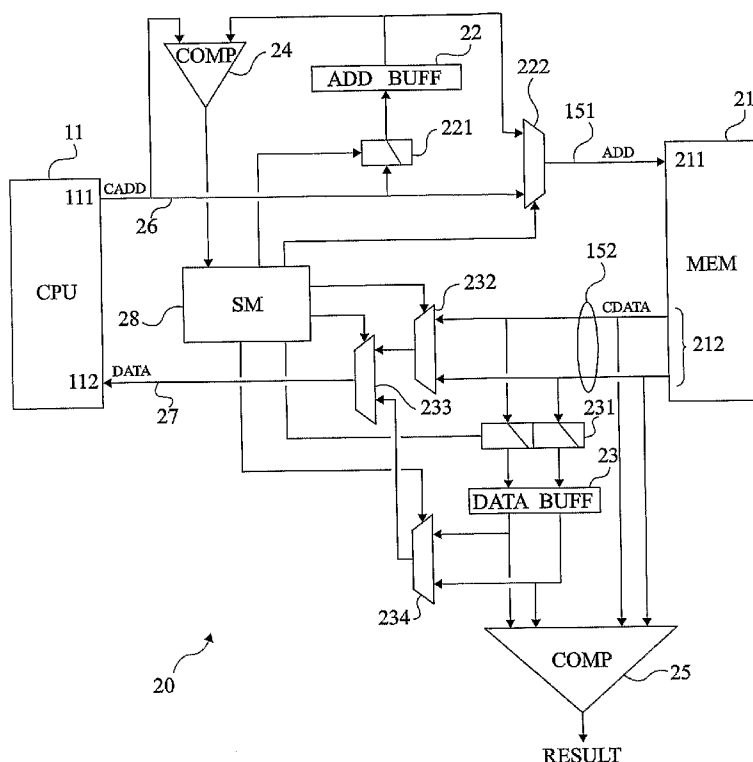


Fig 3

(57) Abstract: The invention relates to a method and a circuit for verifying data transferred between a circuit (21) and a processing unit (11), in which: the data originating from the circuit travels through a first temporary storage element (23) having a size representing an integer multiple of the size of data liable to be presented subsequently on a bus (27) of the processing unit; an address provided by the processing unit (11) destined for the circuit is stored temporarily in a second element (22); and the content of the first element is compared with a current data item (CDATA) originating from the circuit, at least when said data item corresponds to an address of a data item already present in this first element.

(57) Abrégé : L'invention concerne un procédé et un circuit de vérification de données transférées entre un circuit (21) et une unité de traitement (11), dans lequel: les données provenant du circuit transitent par un premier élément de

mémorisation temporaire (23) ayant une taille

[Suite sur la page suivante]



ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), eurasien (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), européen (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

(84) États désignés (sauf indication contraire, pour tout titre de protection régionale disponible) : ARIPO (BW, GH,

Publiée :

— sans rapport de recherche internationale, sera republiée dès réception de ce rapport

VÉRIFICATION DE DONNÉES LUES EN MÉMOIRE

Domaine de l'invention

La présente invention concerne de façon générale les circuits électroniques et, plus particulièrement, les circuits contenant une unité de traitement de données numériques susceptible de traiter des données contenues dans une ou plusieurs
5 mémoires internes ou externes au circuit.

La présente invention s'applique plus particulièrement à la vérification de l'intégrité (de l'absence de modification) d'informations transférées entre une mémoire et une unité de
10 traitement.

Exposé de l'art antérieur

Dans de nombreuses applications, il est souhaitable de s'assurer que des données transférées entre une mémoire et une unité de traitement ne soient pas modifiées soit lors de leur
15 transfert sur les bus de communication, soit lors de leur stockage temporaire entre leur mémoire source et l'unité de traitement. De telles modifications peuvent être accidentelles ou volontaires. Une modification volontaire provient, par exemple, d'une attaque dite par injection de faute qui consiste à introduire
20 une perturbation dans le fonctionnement du circuit électronique (par exemple en intervenant sur son alimentation) de façon à modifier les états des données sur les bus de transfert ou dans

les mémoires. De telles injections de faute peuvent, par exemple, modifier le déroulement d'un programme, modifier des données accédées par ce programme, etc. et peuvent conduire à, par exemple, accepter une authentification erronée, introduire un
5 programme parasite (virus), pirater une clé ou un algorithme de cryptographie, etc.

Pour contrer de telles attaques, on connaît des solutions matérielles consistant à détecter physiquement la perturbation d'origine (lumineuse, thermique, en courant d'alimen-
10 tation, etc.), ou à procéder à des calculs redondants. Ces solutions matérielles sont coûteuses en surface du circuit intégré.

On connaît également des solutions logicielles qui consistent, par exemple, à vérifier que le programme est bien passé par certaines étapes, à lire plusieurs fois successivement
15 des données et les comparer les unes aux autres. Toutes ces solutions ont un coût en performance et notamment en vitesse d'exécution du programme.

Une modification accidentelle du déroulement d'un programme peut provenir d'un environnement bruité (par exemple, un
20 microcontrôleur dans un milieu industriel ou dans un véhicule) et se traduire par des dysfonctionnements non souhaitables.

Résumé de l'invention

La présente invention vise à pallier tout ou partie des inconvénients des solutions connues de contrôle d'intégrité
25 de données lues dans un circuit externe à une unité de traitement, qu'il soit ou non intégré avec cette unité.

Un objet vise plus particulièrement une solution applicable à une mémoire intégrée dans le même circuit que l'unité de traitement.

30 Un autre objet vise plus particulièrement une solution efficace contre des attaques par injection de faute.

Un autre objet est de ne pas dégrader les performances du circuit en raison de la vérification effectuée.

Un autre objet est une solution transparente pour le
35 programme s'exécutant sur l'unité de traitement.

Pour atteindre tout ou partie de ces objets ainsi que d'autres, il est prévu un procédé de vérification de données transférées entre un circuit et une unité de traitement, dans lequel :

5 les données provenant du circuit transitent par un premier élément de mémorisation temporaire ayant une taille multiple de la taille de données susceptibles d'être présentées ensuite sur un bus de l'unité de traitement ;

10 une adresse fournie par l'unité de traitement à destination du circuit est stockée temporairement dans un deuxième élément ; et

15 le contenu du premier élément est comparé avec une donnée courante provenant du circuit, au moins lorsqu'elle correspond à une adresse d'une donnée déjà présente dans ce premier élément.

Selon un mode de mise en oeuvre de la présente invention, toutes les données présentées sur le bus de l'unité de traitement et contenues dans le premier élément sont comparées à la donnée courante extraite de l'élément périphérique à partir de l'adresse stockée dans le deuxième élément.

20 Selon un mode de mise en oeuvre de la présente invention, lorsqu'une adresse courante est présentée par l'unité de traitement, elle est comparée avec l'adresse contenue dans le deuxième élément et, en cas d'identité entre les deux adresses, la donnée contenue dans le premier élément est fournie au bus de l'unité de traitement.

30 Selon un mode de mise en oeuvre de la présente invention, en cas d'absence d'identité entre l'adresse fournie par l'unité de traitement et celle contenue dans le deuxième élément, l'adresse courante est stockée dans le deuxième élément et le contenu du premier élément est remplacé par la donnée fournie par le circuit d'après l'adresse courante, en parallèle à la fourniture d'une partie de cette donnée à l'unité de traitement.

Il est également prévu un procédé de détection d'une injection de faute dans un circuit électronique, dans lequel des données transférées entre une mémoire et une unité de traitement du circuit sont vérifiées par la mise en oeuvre du procédé de
5 vérification, une absence d'identité entre le contenu du premier élément et la donnée courante déclenchant un traitement d'exception.

Il est également prévu un circuit électronique incluant au moins une unité de traitement, et comportant un circuit
10 d'interface entre des bus d'adresses et de données de l'unité de traitement et des bus d'adresses et de données destinés à au moins un circuit périphérique à cette unité, ledit circuit d'interface comportant :

au moins un premier élément de mémorisation temporaire
15 ayant une taille multiple à celle de données transitant sur le bus de données de l'unité de traitement ;

un deuxième élément de mémorisation temporaire de stockage d'une adresse fournie par l'unité de traitement à destination du circuit périphérique ; et

20 des moyens pour la mise en oeuvre du procédé de vérification de données transférées entre le circuit périphérique et l'unité de traitement.

Selon un mode de réalisation de la présente invention, le circuit périphérique est une mémoire.

25 Brève description des dessins

Ces objets, caractéristiques et avantages, ainsi que d'autres seront exposés en détail dans la description suivante de modes de réalisation particuliers faite à titre non-limitatif en relation avec les figures jointes parmi lesquelles :

30 la figure 1 représente une carte à puce du type auquel s'applique à titre d'exemple la présente invention ;

la figure 2 est un schéma-bloc d'un exemple d'architecture de circuit électronique du type auquel s'applique à titre d'exemple la présente invention ;

la figure 3 est un schéma-bloc fonctionnel d'un mode de réalisation d'une interface entre une unité de traitement et une mémoire ; et

la figure 4 est un organigramme illustrant un mode de mise en oeuvre d'un procédé de vérification d'intégrité des données lues.

De mêmes éléments ont été désignés par de mêmes références aux différentes figures.

Description détaillée

Par souci de clarté, seuls les étapes et éléments qui sont utiles à la compréhension de l'invention ont été représentés et seront décrits. En particulier, les fonctions principales des unités de traitement et des programmes traités par ces unités n'ont pas été exposées, l'invention étant compatible avec tout circuit électronique pourvu d'une unité de traitement. De même, la nature des données vérifiées n'a pas été détaillée, l'invention étant là encore compatible avec toute donnée, qu'il s'agisse d'instructions (codes opératoires, opérandes ou arguments) ou de variables. En outre l'invention sera plus particulièrement décrite en relation avec un exemple d'application à la protection de données contre d'éventuelles attaques par injection de fautes dans un circuit intégrant l'unité de traitement et les mémoires dans lesquelles sont lues les données, mais elle s'applique plus généralement à tout dysfonctionnement accidentel. En particulier, pour un fonctionnement d'un microcontrôleur dans un environnement bruité, l'invention trouve également un intérêt si les mémoires sont externes au circuit intégrant l'unité de traitement.

La figure 1 représente de façon très schématique, une carte à puce 1 du type à laquelle s'applique à titre d'exemple la présente invention. Une telle carte est par exemple constituée d'un support 2 en matière plastique dans ou sur lequel est rapportée une puce 10 de circuit électronique susceptible de communiquer avec l'extérieur au moyen de contacts 3 ou au moins d'éléments (non représentés) d'émission/réception sans contact.

Le circuit 10 de la carte contient une unité de traitement capable d'exécuter des programmes stockés dans des mémoires généralement contenues dans le circuit 10 ou dans d'autres circuits portés par la carte.

5 La figure 2 est un schéma-bloc d'un mode de réalisation d'un circuit électronique 10. Ce circuit comporte une unité centrale de traitement 11 (CPU) capable d'exécuter des programmes contenus dans une ou plusieurs mémoires. Dans cet exemple, le circuit 10 comporte une mémoire non volatile non
10 reprogrammable 12 (ROM), une mémoire non volatile reprogrammable 13 (EEPROM) et une mémoire vive 14 (RAM). Un ou plusieurs bus 15 de données, d'adresses et de commandes servent de support de communication entre les différents constituants du circuit 10 et avec une interface 16 (I/O) d'entrée/sortie pour communication
15 avec ou sans contact avec l'extérieur. Le plus souvent, le circuit 10 comporte d'autres fonctions (bloc 17, FCT) dépendant de l'application. Il s'agit, par exemple, de cellules de calcul cryptographique dédiées pour mettre en oeuvre des algorithmes de chiffrement et de déchiffrement.

20 Dans le mode de réalisation représenté en figure 2, le circuit 10 comporte en outre un circuit 20 (CHECK) d'interface entre l'unité centrale et les mémoires pour vérifier l'absence de modification des données entre plusieurs lectures dans la mémoire. Dans l'exemple représenté, le circuit 20 est intercalé
25 entre le bus 15 et l'unité 11 de sorte qu'il vérifie les données lues dans toutes les mémoires. En variante ce traitement peut ne concerner qu'une partie seulement des mémoires.

 La figure 3 est un schéma-bloc fonctionnel d'un mode de réalisation d'une interface 20 de contrôle des données transférées entre une mémoire 21 (MEM) et l'unité de traitement 11
30 (CPU). La mémoire 21 est par exemple l'une des mémoires 12, 13 ou 14 du circuit 10 de la figure 2. Plus généralement, la mémoire 21 peut être constituée par n'importe quel circuit de mémorisation de données, intégré ou non avec l'unité de traitement
35 (par exemple, une mémoire tampon d'un circuit périphérique

intégré à l'unité de traitement, une mémoire de masse externe au circuit intégrant l'unité de traitement, etc.).

Le circuit 20 comporte un élément de mémorisation 22 (ADD BUFF) de stockage temporaire des adresses fournies par l'unité de traitement 11. L'élément 22 prélève, par exemple, sous commande d'un bloc 221 de commutateurs, une adresse sur un bus d'adresses 26 de l'unité de traitement 11. Un sélecteur 222 est intercalé entre ce bus 26 et une entrée 211 de bus d'adresses 151 de la mémoire 21. Le circuit 20 comporte également un élément 23 (DATA BUFF) de stockage temporaire des données extraites de la mémoire 21. Cet élément 23 prélève, par exemple sous commande d'un bloc 231 de commutateurs, les données présentées en sortie 212 de bus de données 152 de la mémoire 21. En variante aux commutateurs 221 et 231, la même fonction de chargement sur commande des adresses et données dans les éléments respectifs 22 et 23 (par exemple des registres) peut être obtenue en masquant ou pas les fronts d'horloge qui commandent la copie des bits dans des bascules formant les éléments 22 et 23.

Selon ce mode de réalisation, l'élément 23 a une taille correspondant à la taille des données dans la mémoire et au double de la taille des données exploitées par l'unité de traitement 11. En d'autres termes, la granularité de lecture de la mémoire 21 (la taille de ses mots) est deux fois plus grande que la taille du bus de données 27 de l'unité de traitement 11. Par exemple, l'élément 23 est un registre 16 bits pour une unité de traitement 8 bits ou un registre 32 bits pour une unité 16 bits, etc. En conséquence, le bus de données 152 entre la mémoire 21 et le circuit 20 est double par rapport au bus de données 27 entre l'unité de traitement 11 et le circuit 20. Un premier sélecteur 232 reçoit le bus 152 et sélectionne une moitié des données. Un deuxième sélecteur 233 reçoit la sortie du sélecteur 232 et la sortie d'un troisième sélecteur 234 séparant le contenu de l'élément 23 en deux. Le sélecteur 233 fournit un mot de données (un demi-mot mémoire) sur le bus 27.

De préférence, les éléments 22 et 23 sont physiquement placés le plus près possible des accès 111 et 112 de l'unité 21 aux bus 26 et 27.

Le circuit 20 comporte en outre deux comparateurs 24 et 25 (COMP), respectivement de l'adresse fournie par l'unité de traitement par rapport à l'adresse contenue dans l'élément 22, et de la donnée présente sur le bus 152 par rapport à celle contenue dans l'élément 23.

Une machine d'états 28 (SM) commande les sélecteurs 222, 232, 233 et 234 en fonction du résultat fourni par le comparateur 24, de même que les blocs 221, 231, donc l'écriture dans les registres 22 et 23. Le comparateur 25 fournit un résultat à destination, par exemple, de l'unité 11 ou d'un autre élément du système pour lui permettre de prendre toute action appropriée en cas de détection d'une modification des données.

Fonctionnellement, la vérification des données par le comparateur 25 est effectuée suite à une lecture dans l'élément 23 en comparant d'un seul coup tout le contenu de l'élément 23 à celui présent sur le bus 152.

On tire profit du fait que, dans la plupart des cas, les mots lus en un seul cycle dans les mémoires sont de taille supérieure aux mots susceptibles d'être introduits par cycle dans l'unité de traitement. Cela permet notamment d'effectuer la vérification des données en arrière-plan dans l'interface mémoire 20.

On tire également profit du fait que, dans la plupart des cas, le chargement des données (programmes, constantes, variables, etc.) entre une mémoire et une unité de traitement s'effectue sur des adresses consécutives. Par conséquent, la deuxième partie de la donnée vérifiée en arrière-plan contient dans la plupart des cas, en deuxième partie, celle que demande effectivement l'unité de traitement au cycle suivant.

La figure 4 illustre un mode de mise en oeuvre du procédé de vérification sous la forme d'un organigramme. Cet

organigramme correspond au traitement effectué par le circuit 20 de la figure 3.

Les registres 22 et 23 sont initialisés (bloc 31, INIT), par exemple, lors d'une activation du circuit 20, ou lors
5 d'une réinitialisation de l'unité de traitement 11. L'initialisation consiste à vider (adresse 0) le registre d'adresse 22 ou à y stocker une adresse arbitraire, et à placer, dans le registre de donnée 23, la donnée contenue dans la mémoire à l'adresse portée par le registre 22.

10 Par la suite, à chaque fois que l'unité de traitement appelle (bloc 32, CADD) une donnée dans la mémoire 21 par fourniture d'une adresse sur son bus 111, on met en oeuvre le processus de vérification ci-après.

On commence (bloc 33, CADD = ADD BUFF ?) par comparer
15 le contenu du registre d'adresse 22 avec l'adresse courante CADD fournie par l'unité de traitement 11. En cas d'identité entre les deux adresses (sortie Y du bloc 33), cela signifie que la donnée correspondante est déjà présente dans le registre de donnée 23. Par exemple, c'est le cas lorsqu'il s'agit du deuxième
20 octet appelé d'un mot de 16 bits lu dans la mémoire alors que l'étape précédente effectuée par l'unité de traitement consistait à lire le premier octet de ce mot de 16 bits. L'octet correspondant P(DATA BUFF) du registre 23 est alors fourni (bloc 34) sur le bus 27 à l'unité de traitement en tant que données
25 DATA.

Dans le même cycle, le contenu du registre de donnée 23 est comparé avec la donnée courante présentée par la mémoire 21 sur le bus 152 (bloc 35, DATA BUFF = MEM(ADD BUFF)?) suite à l'adresse fournie par le registre d'adresse 22. Cette comparaison permet de vérifier que la donnée qui a été fournie à
30 l'unité de traitement 11 et qui était déjà présente dans le registre 23 lors d'un cycle de lecture précédent n'a pas été modifiée par rapport au cycle courant. Il s'agit donc bien d'une vérification d'intégrité de la donnée extraite de la mémoire 21
35 et d'une vérification d'absence de perturbation par injection de

faute. En cas d'incohérence entre les deux données (sortie N du bloc 35), cela déclenche un traitement d'exception FAIL. Un tel traitement consiste, par exemple, en un blocage du circuit électronique 10 ou en toutes autres contremesures adaptées à une
5 injection de faute. En cas d'intégrité validée de la donnée (sortie Y du bloc 35), le processus se poursuit normalement et, du point de vue de l'interface 20, cela signifie se remettre en attente d'une nouvelle adresse CADD (bloc 32) appelée par l'unité de traitement 11.

10 Dans le cas où l'adresse courante n'est pas identique à celle contenue dans le registre 22 (sortie N du bloc 33), cela signifie qu'il faut recharger des données dans le registre 23. Le contenu du registre d'adresse 22 est alors remplacé par l'adresse courante CADD fournie par l'unité de traitement (bloc
15 36, ADD BUFF = CADD). Puis, le contenu du registre de donnée 23 est chargé par la donnée courante fournie par la mémoire 21 (DATA BUFF = CDATA). En parallèle, l'interface mémoire fournit à l'unité de traitement 11 un des deux mots extraits de la mémoire 21 (bloc 38, DATA = P(CDATA)).

20 Pour la mise en oeuvre des modes de réalisation ci-dessus, on utilise un élément de mémorisation temporaire 23 dont la taille correspond à deux fois la taille du bus de données de l'unité de traitement concernée. Tout autre facteur représentant un multiple entier de la taille du bus peut être prévu.

25 Un avantage est que le fonctionnement de l'interface est transparent pour le programme exécuté par l'unité 11 et ne prend pas de temps sur son exécution.

Un autre avantage est que le mécanisme décrit est particulièrement efficace contre les attaques par injection de
30 faute.

Un autre avantage est que le coût en matériel de la mise en oeuvre n'est que de quelques éléments logiques supplémentaires dans l'interface mémoire.

Le fait que la donnée soit fournie à l'unité de
35 traitement malgré la vérification (blocs 34 et 38 en parallèle

avec la vérification 35) n'est en pratique pas gênant. En effet, même si l'erreur n'est détectée que quelques cycles après que la donnée ait été utilisée par l'unité de traitement, l'unité n'aura en pratique pas eue le temps de mettre en danger la
5 sécurité du système.

Des modes de réalisation particuliers de la présente invention ont été décrits. Diverses variantes et modifications apparaîtront à l'homme de l'art. En particulier, la mise en oeuvre pratique de l'invention est à la portée de l'homme du
10 métier à partir des indications fonctionnelles données ci-dessus, notamment pour ce qui est de la synchronisation des échanges entre l'unité de traitement 11 et l'interface 20 et de l'interprétation, par l'unité de traitement 11, des comparaisons effectuées.

15 En outre, bien que l'invention ait été décrite en relation avec une mémoire intégrée au circuit électronique, elle s'applique plus généralement à une vérification de données présentées en entrée d'une unité de traitement, que ces données proviennent d'une mémoire ou de tout autre circuit (par exemple
20 tout périphérique) dans la mesure où ces données sont alors, même dans ce périphérique, généralement contenus dans une mémoire.

Par ailleurs, les adresses fournies par l'unité 11 pourront subir diverses conversions (par exemple, d'adresse
25 virtuelle en adresse physique) sans que cela ne modifie le fonctionnement, pourvu que les deux adresses comparées (comparateur 24 ou bloc 33) soient prises toutes les deux soit avant soit après conversion. De même, les données pourront subir divers traitements entre la mémoire 21 et l'unité 11 (par exemple, un
30 déchiffrement) pourvu que les deux données comparées (comparateur 25 ou bloc 35) soient cohérentes entre elles.

REVENDICATIONS

1. Procédé de vérification de données transférées entre un circuit (21) et une unité de traitement (11), caractérisé en ce que :

les données provenant du circuit transitent par un
5 premier élément de mémorisation temporaire (23) ayant une taille représentant un multiple entier de la taille de données susceptibles d'être présentées ensuite sur un bus (27) de l'unité de traitement ;

une adresse fournie par l'unité de traitement (11) à
10 destination du circuit est stockée temporairement dans un deuxième élément (22) ; et

le contenu du premier élément est comparé avec une donnée courante (CDATA) provenant du circuit, au moins lorsqu'elle correspond à une adresse d'une donnée déjà présente dans ce
15 premier élément.

2. Procédé selon la revendication 1, dans lequel toutes les données présentées sur le bus (27) de l'unité de traitement (11) et contenues dans le premier élément (23) sont comparées à la donnée courante (CDATA) extraite de l'élément
20 périphérique à partir de l'adresse stockée dans le deuxième élément.

3. Procédé selon la revendication 1 ou 2, dans lequel lorsqu'une adresse courante (ADD) est présentée par l'unité de traitement (11), elle est comparée avec l'adresse contenue dans
25 le deuxième élément (22) et, en cas d'identité entre les deux adresses, la donnée contenue dans le premier élément (23) est fournie au bus (27) de l'unité de traitement.

4. Procédé selon la revendication 3, dans lequel en cas d'absence d'identité entre l'adresse fournie par l'unité de
30 traitement (11) et celle contenue dans le deuxième élément (22), l'adresse courante est stockée dans le deuxième élément et le contenu du premier élément (23) est remplacé par la donnée fournie par le circuit (21) d'après l'adresse courante, en

parallèle à la fourniture d'une partie de cette donnée à l'unité de traitement.

5 5. Procédé de détection d'une injection de faute dans un circuit électronique (10), caractérisé en ce que des données transférées entre une mémoire (21) et une unité de traitement (11) du circuit sont vérifiées par la mise en oeuvre du procédé selon l'une quelconque des revendications 1 à 4, une absence d'identité entre le contenu du premier élément (23) et la donnée courante (CDATA) déclenchant un traitement d'exception.

10 6. Circuit électronique (10) incluant au moins une unité de traitement (11), caractérisé en ce qu'il comporte un circuit (20) d'interface entre des bus d'adresses (26) et de données (27) de l'unité de traitement et des bus d'adresses (151) et de données (152) destinés à au moins un circuit (21) périphérique à cette unité, ledit circuit d'interface comportant :

20 au moins un premier élément de mémorisation temporaire (23) ayant une taille représentant un multiple entier de celle de données transitant sur le bus de données de l'unité de traitement ;

un deuxième élément de mémorisation temporaire (22) de stockage d'une adresse fournie par l'unité de traitement à destination du circuit périphérique ; et

25 des moyens pour la mise en oeuvre du procédé selon l'une quelconque des revendications 1 à 4.

7. Circuit selon la revendication 6, dans lequel le circuit périphérique (21) est une mémoire (12, 13, 14).

1/3

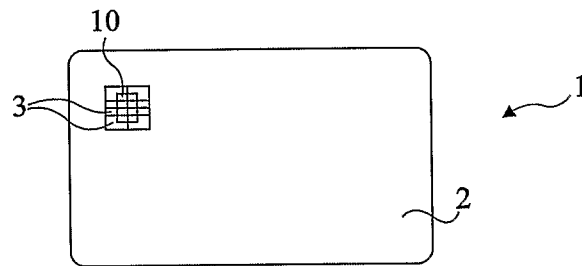


Fig 1

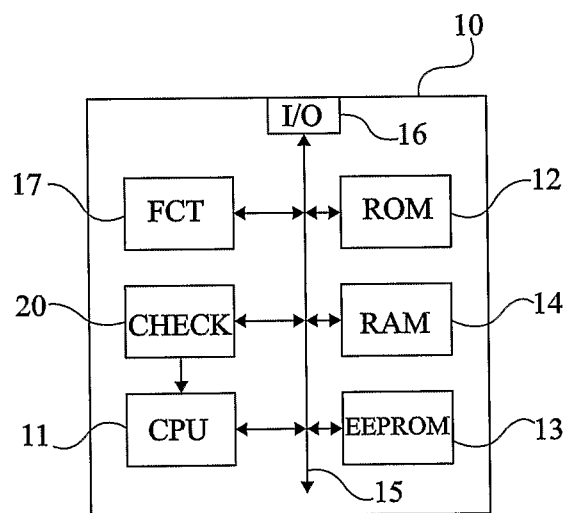


Fig 2

2/3

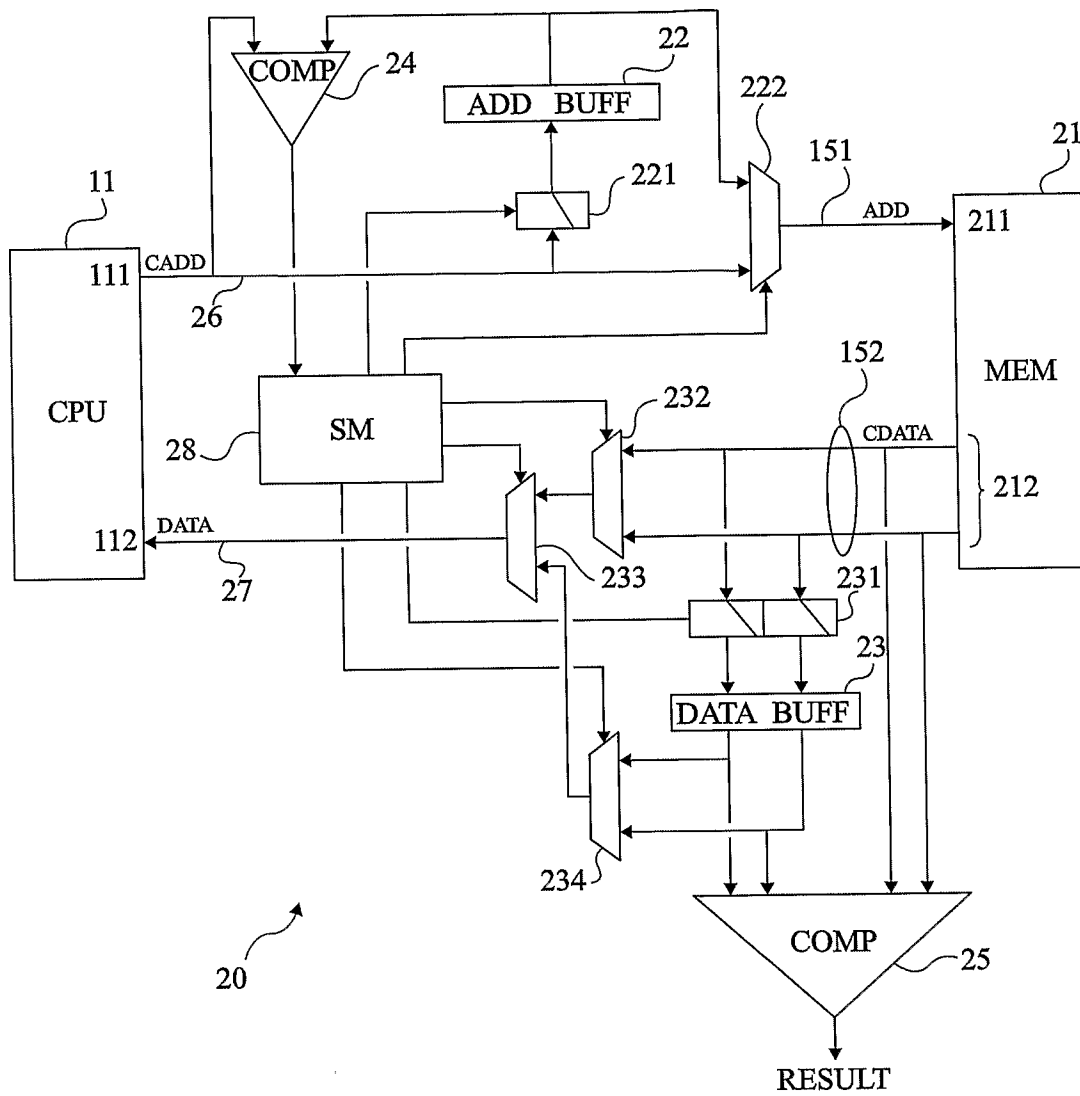


Fig 3

3/3

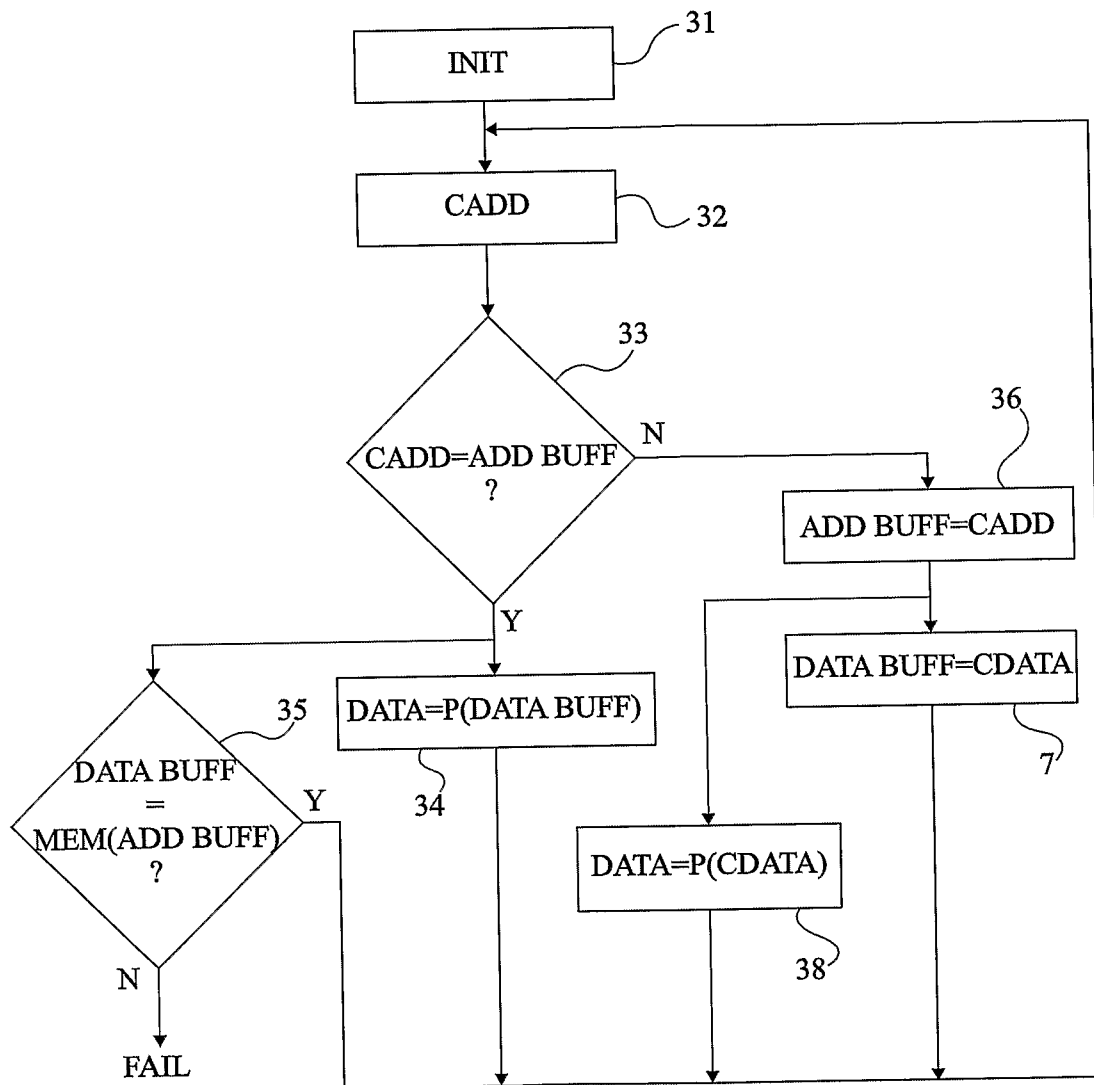


Fig 4