(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE

PARIS

(11) N° de publication : **2 739 706**
(à n'utiliser que pour les commandes de reproduction)

(21) N° d'enregistrement national : **95 12176**

(51) Int Cl[6] : G 06 K 19/073, G 07 F 7/10

(12) DEMANDE DE BREVET D'INVENTION A1

(22) **Date de dépôt** : 09.10.95.

(30) **Priorité** :

(43) **Date de la mise à disposition du public de la demande :** 11.04.97 Bulletin 97/15.

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

(60) **Références à d'autres documents nationaux apparentés :**

(71) **Demandeur(s) :** *INSIDE TECHNOLOGIES SOCIETE ANONYME* — FR.

(72) **Inventeur(s) :** KOWALSKI JACEK et STERN JACQUES.

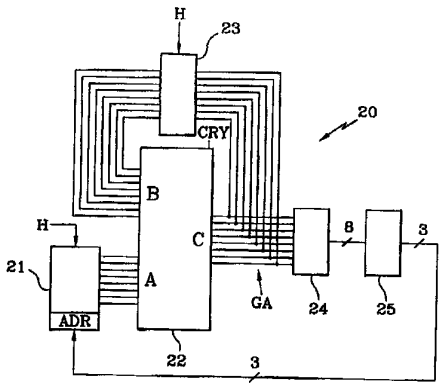
(73) **Titulaire(s) :**

(74) **Mandataire :** CABINET BALLOT SCHMIT.

(54) PERFECTIONNEMENTS AUX CARTES A MEMOIRE.

(57) Procédé pour produire un code d'authentification (CA) à partir d'un code d'entrée (CE), comprenant de façon cyclique les étapes consistant à lire un mot binaire (M_n) dans une mémoire secrète (21) comportant une pluralité de mots binaires. Il est prévu à chaque cycle une opération de combinaison (Σ) des mots lus au cours des cycles précédents, et le résultat de la combinaison est utilisé comme mot générateur (GA) de l'adresse du (M_{n+1}) mot à lire au cycle suivant.

Avantages: production de codes d'authentification à haute résistance contre l'effraction. Application notamment aux procédures d'authentification des cartes à mémoire.



FR 2 739 706 - A1

Perfectionnements aux cartes à mémoire

La présente invention concerne les cartes à mémoire, et plus particulièrement les microcircuits à logique câblée dont sont équipées les cartes à mémoire.

Sous l'appellation générique de "carte à puce", on désigne en fait deux grandes catégories de cartes qui se distinguent au plan de la technologie : d'une part, les cartes à microprocesseur, d'autre part, les cartes dites "à mémoire". A la différence des cartes à microprocesseur, les cartes à mémoire ne disposent que d'un microcircuit à logique câblée qui offre des possibilités beaucoup plus réduites en termes de souplesse d'emploi, de capacité de traitement des données, de programmation, et particulièrement en termes de sécurité et de protection contre la fraude.

En contrepartie, les circuits à logique câblée présentent l'avantage d'être d'une structure plus simple et d'un coût de revient très bas, de sorte que les cartes à mémoire ont connu ces dernières années un important développement dans le cadre d'applications ne nécessitant pas un haut niveau de sécurité. Ainsi, on a vu se généraliser l'emploi des cartes à prépaiement du type carte téléphonique. A l'heure actuelle, de nouvelles applications de type porte-monnaie électronique ou bien encore clef électronique (notamment dans le domaine automobile) sont envisagées à grande échelle.

Il est toutefois nécessaire, pour que ces nombreuses perspectives d'applications puissent se concrétiser, que les cartes à mémoire offrent à l'avenir un niveau de sécurité beaucoup plus élevé qu'aujourd'hui, et qu'elles puissent, sur le plan de la sécurité

d'emploi, rivaliser avec les cartes à microprocesseur qui disposent de mécanismes de sécurité performants implantés sous forme de logiciel dans les mémoires du microprocesseur.

Ainsi, un objectif général de la présente invention est d'améliorer les mécanismes de protection de cartes à mémoire, sans perdre de vue le fait que le coût des circuits à logique câblée augmente très rapidement dès que l'on cherche à réaliser des fonctions de sécurité sophistiquées.

Parmi les perfectionnements que vise la présente invention, certains concernent les procédés d'authentification, comme la reconnaissance du caractère authentique d'une carte par un terminal, ou encore le contrôle au moyen d'un code secret de la légitimité de l'utilisateur, d'autres concernent les moyens permettant de réaliser ces procédés.

Inconvénients des procédés classiques d'authentification

A titre de rappel, on décrira tout d'abord en relation avec la figure 1 la structure et le fonctionnement d'un microcircuit 1 d'une carte à mémoire classique. Le microcircuit 1 est réalisé en logique câblée et comprend essentiellement une mémoire 2 de type série (c'est-à-dire accessible bit par bit), un circuit d'authentification 3 et un séquenceur logique 4 qui gère le fonctionnement des divers éléments à partir d'un signal d'horloge H fourni par un terminal 10 dans lequel la carte est insérée. La mémoire 2 contient, stockés sous forme binaire, un numéro de série de la carte NI (ou numéro d'identification du microcircuit) et des données DA de transaction, par exemple des données représentant la valeur monétaire de la carte ou, dans le cas d'une carte téléphonique, un nombre d'impulsions téléphoniques. Le circuit d'authentification 3 présente une entrée série

3-1 destinée à recevoir un code d'entrée CE et une sortie série 3-2 pour délivrer un code d'authentification CA. Le microcircuit 1 est en outre équipé de plots de contact assurant l'interface électrique avec le terminal 10, dont le nombre est réduit au strict minimum pour des raisons de simplicité. On trouve ainsi un plot d'entrée-sortie I/O pour la communication de données numériques avec le terminal 10, un plot RST d'initialisation du microcircuit, un plot H d'entrée du signal d'horloge, et deux plots d'alimentation électrique Vcc et GND. La sortie de la mémoire 2, ainsi que l'entrée 3-1 et la sortie 3-2 du circuit d'authentification, sont reliées au plot d'entrée-sortie I/O. Il est important de rappeler ici que dans un tel microcircuit 1, les données numériques circulent sous forme série, c'est-à-dire bit par bit en synchronisation avec le signal d'horloge, ce qui permet de simplifier considérablement la structure interne du circuit en ramenant à un seul fil les liaisons entre les divers éléments.

Premier procédé classique : reconnaissance de la carte lors de son introduction dans un terminal

Lorsque la carte est insérée dans le terminal 10, il est indispensable, pour des raisons de sécurité, que le terminal 10 puisse déterminer si la carte est authentique ou frauduleuse. En effet, la carte pourrait avoir été fabriquée illicitement ou rechargée illicitement (pour les cartes à prépaiement rechargeables). Egalement, il pourrait s'agir d'une carte factice reliée par un faisceau de fils à un micro-ordinateur. Le circuit d'authentification 3 va donc intervenir dans une procédure de vérification de l'authenticité de la carte qui va se dérouler d'une manière décrite ci-après. On rappellera préalablement que le terminal 10 est généralement équipé d'un microprocesseur 11 commandé par une mémoire programme 12,

et qu'il connaît les secrets des mécanismes de sécurité implantés dans la carte.

Etape 1 - Lorsque la carte est introduite dans le terminal 10, le terminal 10 génère un code binaire aléatoire ALEXT qu'il applique au circuit d'authentification 3. Le circuit 3 transforme ALEXT en un code binaire d'authentification CA pouvant s'écrire

$$CA = F_{Ks} (ALEXT)$$

F_{Ks} représentant la fonction de transformation, ou fonction d'authentification, réalisée par le circuit 3 à partir d'une clef secrète Ks dont il dispose. Selon une variante connue de cette première étape, les données NI et DA de la mémoire 2 sont chaînées au code aléatoire ALEXT pour former un code d'entrée CE = (NI, DA, ALEXT) appliqué au circuit 3. Dans ce cas, le code d'authentification CA s'écrit

$$CA = F_{Ks} (NI, DA, ALEXT)$$

Etape 2 - Parallèlement à l'étape 1, le terminal 10, qui connaît la clef secrète Ks et la fonction d'authentification F_{ks} (enregistrée sous forme de logiciel dans la mémoire programme 12) calcule, de son côté, un code CA' tel que

$$CA' = F_{Ks} (ALEXT)$$

ou, selon le procédé choisi

$$CA' = F_{Ks} (NI, DA, ALEXT)$$

Etape 3 - Le terminal 10 compare le code CA produit par la carte et le code CA' qu'il a lui-même calculé. Si

les deux codes sont différents, la carte n'est pas authentique et doit être refusée par le terminal.

Dans une variante souvent utilisée du procédé qui vient d'être décrit, le terminal 10 ne connaît pas la clef secrète Ks, mais la détermine à partir du numéro de série NI et au moyen d'une autre clef secrète Kp dont il dispose et d'une fonction de transformation F_{Kp} du type

$$Ks = F_{Kp} (NI)$$

Dans ce cas, l'étape 1 est précédée d'une étape préliminaire où le terminal 10 lit le numéro de série NI dans la mémoire 2 et en déduit Ks.

En définitive, on voit que le mécanisme de protection contre la fraude repose entièrement sur la fonction d'authentification F_{KS}. Une telle fonction ne doit pas pouvoir être décodée par un fraudeur qui pourrait ensuite "tromper" le terminal 10 en lui faisant croire qu'il est en présence d'une carte authentique.

Un inconvénient du procédé qui vient d'être décrit est que la carte peut être "interrogée" à volonté par un fraudeur, qui peut lui envoyer une grande quantité de codes aléatoires ALEXT, observer les codes d'authentification CA émis en retour par le circuit d'authentification 3, et essayer, par recoupements, de décoder la fonction d'authentification F_{Ks} ou trouver son secret Ks.

Ainsi, un objectif de la présente invention est d'éviter qu'une carte à mémoire puisse être interrogée à répétition par une personne non habilitée qui cherche à en découvrir le secret.

Pour atteindre cet objectif, la présente invention propose un procédé dans lequel la carte authentifie le terminal avant que le terminal n'authentifie la carte, et

la carte ne délivre pas son code d'authentification tant qu'elle n'a pas vérifié l'authenticité du terminal.

Plus particulièrement, la présente invention prévoit un microcircuit à logique câblée, notamment pour carte à mémoire, comprenant un circuit d'authentification pour produire un code d'authentification à partir d'un code binaire d'entrée et des moyens pour échanger des données numériques avec un terminal, microcircuit comprenant en outre des moyens pour générer un code aléatoire, des moyens pour envoyer le code aléatoire vers le terminal ainsi qu'à l'entrée du circuit d'authentification, et des moyens pour provoquer un blocage au moins partiel du fonctionnement interne du microcircuit si un code d'authentification externe produit par le terminal sur réception du code aléatoire est différent d'un code d'authentification interne produit par le circuit d'authentification au moins à partir du code aléatoire.

Bien entendu, les moyens de blocage doivent être choisis au moins de manière à empêcher la procédure d'authentification de la carte.

Selon un mode de réalisation particulièrement simple et avantageux de la présente invention, les moyens de blocage comprennent des moyens pour empêcher la transmission de données numériques entre le circuit d'authentification et le terminal.

Par exemple, les moyens de blocage peuvent comprendre un comparateur logique recevant sur ses entrées le code d'authentification externe et le code d'authentification interne, et une porte logique recevant sur sa première entrée la sortie du comparateur et sur sa deuxième entrée la sortie du circuit d'authentification, la sortie de la porte logique étant connectée à un plot d'entrée-sortie permettant l'échange de données numériques entre le microcircuit et le terminal. Ainsi,

si la porte logique se bloque selon le résultat à la sortie du comparateur, aucune donnée numérique émise par le circuit d'authentification ne pourra être reçue par le terminal.

La présente invention concerne également un procédé d'authentification d'un terminal réalisé par un microcircuit quand le microcircuit est inséré dans le terminal, comprenant les opérations suivantes : simultanément, générer et envoyer un code aléatoire vers le terminal et dans un circuit d'authentification du microcircuit, produire un code d'authentification interne à partir du code aléatoire et recevoir simultanément du terminal un code d'authentification externe, comparer le code d'authentification interne et le code d'authentification externe, bloquer la sortie du circuit d'authentification si le code d'authentification externe est différent du code d'authentification interne.

Selon un mode de réalisation, les codes d'authentification interne et externe sont produits à partir du code aléatoire et de données comprises dans une mémoire du microcircuit.

Avantageusement, la mémoire contient dans une zone non accessible en lecture depuis l'extérieur un code secret supposé connu par un utilisateur, le code d'authentification interne est produit par le circuit d'authentification à partir du code aléatoire auquel est ajouté le code secret. On peut ainsi procéder simultanément à l'authentification du terminal et à l'authentification d'un utilisateur du microcircuit.

Deuxième procédé classique : authentification de l'utilisateur par la carte et le terminal

Outre l'authentification de la carte, un deuxième mécanisme classique de sécurité est la vérification de la légitimité du porteur de la carte. Cette procédure qui a pour but de protéger les cartes contre le vol repose sur

l'introduction préalable par l'utilisateur d'un code secret CS' dans le terminal. Comme illustré en traits pointillés sur la figure 1, on équipe le microcircuit 1 déjà décrit d'un circuit 5 de vérification qui va recevoir (sous forme numérique) du terminal 10 le code secret CS' que le terminal a lui-même reçu de l'utilisateur. Le circuit 5 est par exemple un comparateur logique de type série qui possède le code secret CS. Si le code envoyé CS' par le terminal, tel qu'il lui a été donné par l'utilisateur, correspond au code CS inscrit dans le circuit 5, la sortie du circuit 5 va par exemple délivrer sur le plot I/O un bit égal à 1 et le terminal saura qu'il peut commencer la transaction. Si par contre le bit délivré est à 0, le terminal va indiquer à l'utilisateur que le code CS' est erroné.

Cette technique d'authentification de l'utilisateur au moyen d'un circuit à logique câblée présente l'inconvénient d'être complexe et coûteuse à mettre en oeuvre. Un autre inconvénient de ce procédé classique est que le code secret circule clairement sur la ligne de communication carte/terminal, de sorte qu'un "espionnage" électronique de cette ligne permettrait à coup sûr de connaître le code secret.

Ainsi, un autre objectif de la présente invention est de prévoir une solution simple permettant la vérification du code secret CS' de l'utilisateur sans faire appel à des moyens à logique câblée, tout en offrant de meilleures conditions de sécurité que les techniques classiques.

Pour atteindre cet objectif, l'idée de la présente invention est de faire intervenir le circuit d'authentification 3, normalement réservé à la procédure de reconnaissance de la carte par le terminal, dans la procédure de vérification du code secret.

Plus particulièrement, la présente invention prévoit un procédé pour contrôler la validité d'un code secret d'un utilisateur d'un microcircuit lors de l'introduction du microcircuit dans un terminal, comprenant une opération préliminaire au cours de laquelle l'utilisateur communique au terminal son code secret, le microcircuit étant équipé d'une mémoire dans laquelle se trouve enregistré au moins le code secret, le terminal et le microcircuit comportant des moyens d'authentification aptes à produire un code d'authentification à partir d'un code d'entrée, procédé dans lequel le code secret est disposé dans une zone de la mémoire non accessible par le terminal, et comprenant les étapes suivantes : production par le microcircuit d'un code d'authentification à partir d'un code d'entrée comprenant le code secret contenu dans la mémoire, production par le terminal d'un code d'authentification à partir d'un code d'entrée comprenant le code secret fourni par l'utilisateur du microcircuit, comparaison du code d'authentification produit par le microcircuit et du code d'authentification produit par le terminal, l'utilisateur étant habilité à utiliser le microcircuit si les deux codes d'authentification sont identiques.

Selon un mode de réalisation, les codes d'authentification sont produits à partir de codes d'entrée comprenant en outre un code aléatoire.

Le code aléatoire peut être généré par le microcircuit ou par le terminal.

La comparaison du code d'authentification produit par le microcircuit et du code d'authentification produit par le terminal peut être réalisée par le terminal ou par le microcircuit.

Bien entendu, le procédé de la présente invention peut permettre simultanément l'authentification du microcircuit par le terminal ou l'authentification du

terminal par le microcircuit. En effet, si du microcircuit ou du terminal, l'un des deux n'est pas authentique, les deux codes d'authentification ne seront pas identiques.

Dans ce cas, il peut être prévu d'empêcher la transmission de données numériques entre la sortie des moyens d'authentification du microcircuit et le terminal, quand le code d'authentification produit par le microcircuit et le code d'authentification produit par le terminal sont différents.

Par ailleurs, le code d'authentification peut être produit à partir d'un code d'entrée comprenant en outre d'autres données stockées dans la mémoire, par exemple un numéro de série du microcircuit.

Enfin, la zone mémoire où est disposé le code secret peut être rendue accessible en écriture une fois que la validité du code secret de l'utilisateur a été contrôlée, afin que l'utilisateur puisse modifier le code secret.

Inconvénients des circuits classiques d'authentification

On a vu dans ce qui précède que la sécurité offerte par une carte à mémoire repose entièrement sur le circuit d'authentification 3. Egalement, on a vu que la transmission de données numériques s'effectue en série pour simplifier la structure du microcircuit.

Ainsi, un circuit d'authentification, pour être performant, doit présenter les caractéristiques ou avantages suivants :

- une entrée série et une sortie série,
- la possibilité de produire un code d'authentification long, c'est-à-dire d'au moins 16 bits, après avoir introduit le code d'entrée CE,

- une très haute sécurité, c'est-à-dire la quasi impossibilité pour un fraudeur de déceler le fonctionnement interne du circuit d'authentification,
- la production d'un bit du code d'authentification à chaque coup d'horloge,
- la production de deux codes d'authentification CA très différents à partir de deux codes d'entrée CE très ressemblants, par exemple ne se différenciant que par un bit (une même suite de "1" et de "0" ne comportant qu'un bit n'ayant pas la même valeur).

Pour l'homme de l'art, et comme on l'a illustré en figure 2, un circuit d'authentification 3 vu de l'extérieur est une machine logique 6 cadencée par un signal d'horloge H, dans laquelle on injecte, en synchronisation avec le signal d'horloge H, une série de bits formant le code d'entrée CE, et de laquelle on extrait, toujours en synchronisation avec l'horloge, une série de bits formant le code d'authentification CA. Par "machine logique" on entend dans la présente demande de brevet un circuit logique caractérisé à un instant donné par un certain état logique interne, puis à un instant suivant par un autre état (logique) interne, ainsi de suite, capable de fonctionner de façon autonome, c'est-à-dire de passer d'un état interne à un autre état interne sur réception d'un signal d'horloge même quand aucun code d'entrée CE ne lui est appliqué. Le mode de fonctionnement de la machine logique 6 doit être secret et il est généralement basé sur une clef secrète Ks. L'introduction des bits du code d'entrée CE modifie les transitions d'états internes de la machine logique, et le code d'authentification CA extrait en série de la machine logique est représentatif des transitions d'états internes de la machine.

Si l'on souhaite produire un code série CA d'une certaine longueur, par exemple un code de 16 bits, après

que le code d'entrée CE a été introduit, il est nécessaire de disposer d'une machine logique présentant un grand nombre d'états internes et une grande diversité dans les enchaînements de ses états internes. Par exemple, pour produire un code d'authentification CA de 16 bits en série après que le code d'entrée CE a été introduit, il faut disposer d'une machine logique pouvant effectuer d'elle-même environ 65500 transitions différentes entre ses états internes afin d'exploiter toutes les possibilités offertes par les 16 bits du code d'authentification (un code de 16 bit pouvant prendre environ 65500 valeurs).

Dans l'art antérieur, et en particulier dans les brevets français FR 92 13913 et FR 89 09734, on a proposé des circuits d'authentification réalisés à partir d'un même type de machine logique, représenté en figure 3. La machine logique 6 de l'art antérieur comprend une mémoire secrète 7 dont la sortie de type parallèle est ramenée sur l'entrée d'adresse ADR par l'intermédiaire d'un registre tampon 8. La mémoire secrète 7 contient une pluralité de mots binaires M1, M2,....Mn représentant la clef secrète Ks. A chaque coup d'horloge, l'adresse du mot à lire dans la mémoire secrète est déterminée en partie par le mot lu au cycle précédent, et en partie par le bit du code d'entrée CE, qui est appliqué sur un bit de l'entrée d'adresse ADR de la mémoire 7. Le bit du code d'authentification CA est prélevé à la sortie de la mémoire secrète.

L'inconvénient d'une telle machine logique est que, une fois le code d'entrée introduit, les transitions entre les états internes ne dépendent plus que des mots M_i contenus dans la mémoire secrète 7. Ainsi, par exemple, si l'on voulait obtenir 65000 transitions d'états une fois le code CE introduit, on devrait prévoir une mémoire secrète d'une capacité de 65000 mots

binaires, ce qui n'est pas envisageable en pratique pour des raisons de coût. Pour pallier ces inconvénients, on a proposé dans le brevet FR 92 13913 de procéder à plusieurs passages successifs du code d'entrée CE dans la machine logique avant de délivrer le code d'authentification CA. Cette solution présente toutefois l'inconvénient de nécessiter plusieurs coups d'horloge pour l'obtention d'un seul bit du code d'authentification CA, ce qui ralentit considérablement le fonctionnement du circuit d'authentification et la durée de la procédure d'authentification.

Un objectif de l'invention est de prévoir un circuit d'authentification performant qui présente les caractéristiques et avantages énumérés plus haut.

Un autre objectif de l'invention est de prévoir une machine logique et un circuit d'authentification utilisant une mémoire secrète qui présentent un grand nombre d'état internes pour un nombre limité de mots dans la mémoire secrète.

Un objectif plus particulier de l'invention est de prévoir un circuit d'authentification pouvant présenter environ 65000 transitions d'états internes afin de pouvoir produire des codes d'authentification d'au moins 16 bits.

Encore un autre objectif de l'invention est de prévoir une machine logique et un circuit d'authentification qui soient simples à fabriquer et d'un prix de revient réduit.

Ces objectifs sont atteints grâce à un procédé pour produire un code d'authentification, comprenant de façon cyclique les étapes consistant à lire un mot binaire dans une mémoire secrète comportant une pluralité de mots binaires, procédé dans lequel, à chaque cycle, il est prévu une opération de combinaison des mots lus au cours des cycles précédents, le résultat de la combinaison

étant utilisé comme mot générateur de l'adresse du mot à lire au cycle suivant.

Grâce à ce procédé, on peut réaliser une machine logique dans laquelle la transition d'un état interne à un autre ne dépend plus du mot qui vient d'être lu dans la mémoire, mais du cumul des mots qui ont été lus. Ainsi, on peut obtenir, à partir d'un nombre limité de mots binaires stockés dans la mémoire, un grand nombre d'état internes différents et de transitions entre ces états.

Selon un mode de réalisation, l'opération de combinaison inclut le mot qui vient d'être lu dans la mémoire.

Selon un mode de réalisation, l'opération de combinaison consiste à faire l'addition des mots binaires lus dans la mémoire secrète. Dans ce cas, le mot générateur d'adresse est donc la somme des mots lus au cours des cycles précédents.

Avantageusement, il est en outre prévu une première opération de transformation du mot générateur d'adresse, consistant à combiner entre eux, de façon logique, au moins une partie des bits du mot générateur d'adresse. On complique ainsi un peu plus le fonctionnement de la machine logique.

Avantageusement, il est en outre prévu une deuxième opération de transformation du mot générateur d'adresse, consistant à combiner, de façon logique, des bits du mot générateur d'adresse avec des bits internes d'un registre à décalage. On multiplie ainsi les possibilités de transition d'états internes, qui vont être égales au produit des états internes possibles du registre à décalage et des valeurs possibles du mot générateur d'adresse.

Avantageusement, il est en outre prévu une troisième opération de transformation du mot générateur

d'adresse, consistant à réduire le nombre de bits du mot générateur d'adresse dans une quantité correspondant au nombre d'entrées d'adresse de la mémoire secrète.

Avantageusement, au moins un bit du mot générateur d'adresse est combiné de façon logique, à chaque cycle, avec un bit d'un code d'entrée.

Avantageusement, au cours d'une phase d'initialisation du procédé, au moins un bit du mot générateur d'adresse est combiné de façon logique, à chaque cycle, avec un bit d'un code d'entrée, et au cours d'une phase de production du code d'authentification, un bit du mot générateur d'adresse est prélevé, à chaque cycle, pour former un bit du code d'authentification.

Selon un mode de réalisation, les phases d'initialisation et de production sont réalisées simultanément, le code d'authentification étant produit pendant que le code d'entrée est absorbé.

Selon un mode de réalisation, les phases d'initialisation et de production sont réalisées séquentiellement, la phase de production du code d'authentification commençant lorsque tous les bits du code d'entrée ont été absorbés au cours de la phase d'initialisation.

La présente invention concerne également une machine logique cadencée par un signal d'horloge et comprenant une mémoire secrète dans laquelle est stockée une pluralité de mots binaires, machine dans laquelle la sortie de la mémoire est appliquée sur une première entrée d'un circuit logique et la mémoire est lue à chaque cycle d'horloge, la sortie du circuit logique est renvoyée à chaque cycle d'horloge sur une deuxième entrée du circuit logique par l'intermédiaire d'un circuit tampon, le circuit logique réalise une combinaison de ses deux entrées et délivre à chaque cycle d'horloge un mot binaire générateur d'adresse envoyé sur l'entrée

d'adresse de la mémoire. Ainsi, le circuit logique, grâce à la fonction de combinaison qu'il réalise, démultiplie le nombre d'états internes de la machine logique, en ce sens que le nombre d'états internes que peut présenter la machine logique est très supérieur au nombre de mots présents dans la mémoire.

Le mot générateur d'adresse peut être prélevé à la sortie du circuit logique, ou à la sortie du circuit tampon.

Selon un mode de réalisation, le circuit logique est un circuit additionneur.

Selon un mode de réalisation, la machine logique comprend des moyens logiques pour réduire le nombre de bits du mot générateur d'adresse.

Selon un mode de réalisation, la machine logique comprend des moyens logiques pour combiner entre eux des bits du mot générateur d'adresse.

Selon un mode de réalisation, la machine logique comprend un registre à décalage, et des moyens logiques pour combiner au moins un bit du registre à décalage avec au moins un bit du mot générateur d'adresse. Dans ce cas, le nombre d'états internes de la machine logique devient donc égal au nombre de valeurs possibles du mot générateur d'adresse multiplié par le nombre d'états internes que peut présenter le registre.

La présente invention concerne également un circuit d'authentification à entrée série et sortie série, pour produire un code d'authentification à partir d'un code d'entrée, comprenant une machine logique selon l'invention, des moyens logiques pour injecter, à chaque cycle d'horloge, un bit du code d'entrée dans la machine logique, l'injection du bit du code d'entrée consistant à combiner au moins un bit du mot générateur d'adresse avec le bit du code d'entrée, et des moyens pour extraire un

bit de la machine logique afin de produire le code d'authentification.

Ces objets, caractéristiques et avantages ainsi que d'autres de la présente invention apparaîtront plus clairement à la lecture de la description suivante :

- d'un procédé, d'une machine logique et d'un circuit d'authentification selon l'invention, pour produire un code d'authentification à partir d'un code d'entrée,
- d'un procédé d'authentification mutuelle selon l'invention, applicable entre une carte à mémoire et un terminal, ainsi qu'une structure de microcircuit selon l'invention permettant de mettre en oeuvre le procédé,
- d'un procédé d'authentification selon l'invention d'un utilisateur de carte à mémoire,

faite en relation avec les figures jointes parmi lesquelles :

la figure 1 est le schéma électrique d'un microcircuit de carte à mémoire classique, et a été précédemment décrite,

la figure 2 représente un circuit d'authentification mis en oeuvre dans une carte à mémoire classique, et a été précédemment décrite,

la figure 3 est le schéma de principe d'une machine logique de l'art antérieur, et a été précédemment décrite,

la figure 4 représente sous forme de blocs une machine logique selon l'invention,

la figure 5 représente une variante de réalisation de la machine logique de la figure 4,

la figure 6 représente une autre variante de la machine logique de la figure 4,

la figure 7 représente de façon plus détaillée un circuit d'authentification utilisant la machine logique de la figure 6,

la figure 8 est un schéma électrique représentant sous forme de blocs un microcircuit de carte à mémoire selon l'invention, permettant de mettre en oeuvre un procédé d'authentification mutuelle en une carte à mémoire et un terminal, et

la figure 9 représente le schéma logique d'un bloc de la figure 8.

Machine logique et circuit d'authentification, notamment pour carte à mémoire

On rappelle qu'un objectif de la présente invention est de prévoir une machine logique à grand nombre d'états internes qui soit simple et peu coûteuse à fabriquer. A partir de cette machine logique, il sera ensuite possible de construire un circuit d'authentification performant.

La figure 4 représente une machine logique 20 selon l'invention. La machine logique 20 comprend une mémoire 21 secrète (c'est-à-dire non accessible de l'extérieur) commandée par un signal d'horloge H, un circuit logique 22 à deux entrées A, B et une sortie C de type parallèle. La machine logique 20 comprend également un registre tampon 23 commandé par le signal d'horloge H, un circuit logique mélangeur 24 et un circuit logique réducteur 25. La mémoire 21 contient une pluralité de mots binaires formant la clef secrète Ks de la machine logique 20, et sa sortie est appliquée sur l'entrée A du circuit 22. La sortie C du circuit 22 est ramenée sur son entrée B par l'intermédiaire du registre tampon 23 et est également appliquée à l'entrée du circuit mélangeur 24. La sortie du circuit mélangeur 24 est appliquée sur l'entrée du circuit réducteur 25 dont la sortie attaque l'entrée d'adresse ADR de la mémoire 21. Le circuit 24 est optionnel et a pour fonction de mélanger, c'est-à-dire combiner de façon logique, les bits de la sortie C du circuit 22 pour créer un effet de "brouillage", afin que le fonctionnement de la machine logique 20 soit aussi

complexe et indéchiffrable que possible. Le circuit 25 a pour fonction de réduire, lorsque cela est nécessaire, le nombre de bits délivrés par le circuit 24, afin d'obtenir le nombre de bits nécessaires au pilotage de l'entrée d'adresse ADR de la mémoire 21. Le circuit 22 réalise une fonction de combinaison Fc de ses deux entrées A et B et délivre sur la sortie C un mot binaire GA pouvant s'écrire :

$$GA = A \; Fc \; B$$

Selon l'invention, la fonction de combinaison Fc est une fonction à sens unique, ce qui signifie que le mot binaire GA délivré par la sortie C ne peut pas révéler les valeurs des entrées A et B (la fonction OU EXCLUSIF calculée bit à bit est par exemple une fonction à sens unique). Par la suite, on appellera GA "mot générateur d'adresse", car la valeur appliquée sur l'entrée d'adresse ADR de la mémoire 21 va, à chaque coup d'horloge H, être générée à partir du mot GA.

Lorsque l'on applique à la mémoire 21 et au registre tampon 23 un coup d'horloge H, un mot lu dans la mémoire 21 est appliqué à l'entrée A du circuit 22. Parallèlement, le mot présent sur la sortie C du circuit 22 est recopié par la sortie du registre tampon 23 et appliqué à l'entrée B du circuit 22. L'homme de l'art notera qu'en pratique, pour des raisons de synchronisation, il faut prévoir un petit décalage temporel entre l'application du coup d'horloge H sur la mémoire 21 et son application sur le registre tampon 23.

Ainsi, si l'on applique un nombre n de coups d'horloge H après une remise à zéro de la machine logique 20, on va trouver à la sortie du circuit 22, au coup d'horloge H_n, un mot GA générateur de l'adresse du mot à lire dans la mémoire 21 au coup d'horloge suivant H_{n+1},

ce mot GA étant le résultat de la combinaison de tous les mots M_1, M_2, M_3, M_4....M_n lus dans la mémoire 21 depuis le premier coup d'horloge, et pouvant ainsi s'écrire :

$$GA = M_1 \text{ Fc } M_2 \text{ Fc } M_3 \text{ FC } M_4 \text{ Fc } M_5 \ldots\ldots\ldots\ldots\ldots \text{Fc } M_n$$

Ainsi, grâce à l'opération de combinaison de la présente invention, un grand nombre de mots générateurs d'adresse GA différents peuvent être générés à partir d'un nombre limité de mots stockés dans la mémoire 21, ce qui garantit de nombreuses possibilités en termes de transitions d'états.

Dans un mode de réalisation préféré par la demanderesse en raison de sa simplicité, le circuit 22 est un additionneur huit bits dont la sortie CRY "report de somme" est laissée en l'air, et la mémoire 21 contient huit mots binaires de huit bits chacun. Le mot GA sur la sortie C de l'additionneur 22 est alors un mot de huit bits g0 à g7 qui constitue le résultat de l'addition modulaire modulo 255 des mots M_1, M_2, M_3, ...M_n lus dans la mémoire 21 :

$$GA = \Sigma \; M_1 \text{ à } M_n \text{ (modulo 255)}$$

Dans ce cas, on obtient 256 mots générateurs d'adresse différents, soit 256 états internes et 256 possibilités de transition d'états, à partir d'une clef secrète Ks contenant seulement 8 mots de 8 bits.

Une variante 20-1 de réalisation de la machine logique 20 selon l'invention est représentée en figure 5. Selon cette variante, l'entrée du circuit mélangeur 24 est attaquée par la sortie du registre tampon 23 qui est toujours appliquée sur l'entrée B du circuit 22. Dans ce cas, le mot générateur d'adresse GA est le résultat de

l'addition modulaire modulo 255 des mots M_1, M_2, M_3, ...M_{n-1} lus au cours des cycles d'horloge précédents :

$$GA = \Sigma\ M_1 \text{ à } M_{n-1} \text{ (modulo 255)}$$

On rappelle maintenant qu'un objectif particulier de la présente invention est d'obtenir une machine logique présentant environ 65000 transitions possibles entre ses états internes, afin de pouvoir exploiter la pleine échelle d'un code de 16 bits que l'on souhaite produire. Ce résultat peut être atteint simplement en remplaçant l'additionneur 22 de huit bits par un additionneur de seize bits (soit 65536 valeurs possibles pour le mot générateur d'adresse GA) tout en conservant la mémoire secrète de huit mots de huit bits. Toutefois, cette solution n'est pas avantageuse industriellement en raison du surcoût de fabrication qu'elle entraînerait. On cherche donc à se limiter à une structure de 8 bits.

Ainsi, une idée de la présente invention, pour multiplier les possibilités de la machine logique, est de faire intervenir un registre à décalage fonctionnant en mode pseudo-aléatoire, par exemple un registre de 8 bits, et d'injecter au moins un bit du registre pseudo-aléatoire dans le mot générateur d'adresse GA. Dans ce cas, le nombre d'états internes de la machine logique va être modifié et sera porté à 256 x 256 soit environ 65000 possibilités, chaque état interne du registre pseudo-aléatoire pouvant se combiner avec chaque état interne du mot générateur d'adresse GA.

La figure 6 représente une machine logique 30 mettant en oeuvre ce deuxième aspect de l'invention. On y retrouve la mémoire 21 ainsi que les circuits 22, 23, 24, 25, disposés de la manière décrite en relation avec la figure 4. La machine logique 30 comprend en outre un registre à décalage 26 de huit bits r0, r1,...r7, cadencé

par l'horloge H et agencé en mode fonctionnement pseudo-aléatoire.

Le mode de fonctionnement pseudo-aléatoire du registre 26 est assuré quand au moins un bit interne r0-r7 du registre 26 et au moins un bit du mot générateur d'adresse GA sont combinés ensemble de façon logique pour former le bit d'entrée du registre 26 au coup d'horloge suivant. Ainsi, dans l'exemple montré par la figure 6, on a choisi de combiner trois bits r1, r4 et r6 du registre 26 dans un circuit logique 27 (on aurait pu en prendre plus, ou moins). La sortie du circuit 27 délivre un bit qui est combiné dans un circuit logique 28 avec trois bits du mot générateur d'adresse GA (ici aussi, on aurait pu en prendre plus, ou moins). La sortie du circuit 28 est appliquée à l'entrée du registre 26.

D'autre part, pour que les 256 états internes possibles du registre pseudo-aléatoire 26 se combinent avec les 256 états possibles du mot générateur d'adresse GA, et confèrent à la machine logique 30 environ 65000 possibilités en termes de transitions d'états, au moins un bit du registre pseudo-aléatoire 26 doit être combiné avec au moins un bit du mot générateur d'adresse GA. Dans l'exemple de la figure 6, on a choisi d'envoyer dans le circuit réducteur 25 le bit délivré par le circuit 27 (ce bit étant représentatif des trois bits r1, r4, r6 du registre 26).

De préférence, le circuit réducteur 25, le circuit 27 et le circuit 28 sont des circuits logiques linéaires, c'est-à-dire comprenant des fonctions logiques à base de portes OU EXCLUSIF.

La figure 7 montre comment on réalise un circuit d'authentification 40 à partir de la machine logique 30 qui a été décrite en relation avec la figure 6. Le circuit 40 délivre un code d'authentification CA de type série à partir d'un code d'entrée CE également de type

série. On y retrouve la mémoire secrète 21 contenant la clef secrète formée par les 8 mots de 8 bits, l'additionneur 22 et le registre tampon 23, et les autres éléments précédemment décrits.

Le circuit mélangeur 24 comprend huit sous-ensembles logiques 24-0, 24-1, 24-2...24-7 délivrant des bits g'0, g'1, g'2...g'7 résultant du mélange logique des bits g0, g1, g2, g3,...g7 du mot générateur d'adresse GA. Par exemple, chaque sous-ensemble 24-0 à 24-7 comprend une porte NON OU à deux entrées dont la sortie est appliquée sur une entrée d'une porte NON ET à deux entrées. Chaque bit g'_i de rang i délivré par un sous-ensemble 20-i correspondant est par exemple de la forme logique suivante (le symbole "/" représentant le NON logique) :

$$g'_i = /(g_i \text{ ET } /(g_{i+1} \text{ OU } g_{i-1})),$$

à l'exception du bit de plus faible poids g'0 qui est de la forme :

$$g'0 = /(g0 \text{ ET } /g1)$$

et du bit de plus fort poids g'7 qui est de la forme :

$$g'7 = /(g7 \text{ ET } /g6)$$

Par ailleurs, on a choisit d'introduire le code d'entrée CE au niveau du circuit 27. Le circuit 27 est par exemple une porte OU EXCLUSIF à quatre entrées recevant le bit du code d'entrée CE et les trois bits r1, r4, r6 du registre pseudo-aléatoire.

Le circuit 28 dont la sortie attaque l'entrée du registre pseudo-aléatoire 26 est par exemple une porte OU EXCLUSIF à quatre entrées recevant par exemple les bits

g'2 g'5 g'7 du mot générateur d'adresse GA transformé par le circuit mélangeur 24 et le bit délivré par la porte OU EXCLUSIF 27.

Le circuit réducteur 25 comprend par exemple trois portes OU EXCLUSIF 25-1, 25-2, 25-3 à quatre entrées, délivrant respectivement des bits a0, a1, a2 appliqués sur l'entrée d'adresse ADR de la mémoire secrète 21. La porte 25-1 reçoit par exemple sur son entrée la sortie de la porte 27 et les bits g'0, g'1, g'2, la porte 25-2 les bits g'2, g'3, g'4, g'5 et enfin la porte 25-3 reçoit les bits g'4, g'5, g'6, g'7.

Enfin, le bit du code d'authentification CA peut être prélevé en amont du point où est injecté le code d'entrée CE, selon le sens de circulation des états logiques des bits, par exemple à la sortie de l'additionneur 22, à la sortie du bit g2 du mot générateur d'adresse GA. Dans ce cas, le bit g2 forme donc le bit du code d'authentification CA.

Grâce à la présente invention, on dispose d'un circuit d'authentification 40 simple à réaliser tout en étant apte à délivrer en 16 coups d'horloge H seulement un code d'authentification CA de 16 bits et en exploitant les 65000 possibilités offertes par le code. Comme dans l'art antérieur, le code CA pourra s'écrire

$$CA = F_{Ks}\ (CE)$$

F_{Ks} étant la fonction de transformation réalisée par le circuit 40 à partir de la clef secrète Ks qui est seulement constituée de 8 mots de 8 bits.

Il apparaîtra clairement à l'homme de l'art que le circuit d'authentification selon l'invention peut faire l'objet de nombreuses variantes et modes de réalisation. Toutefois, comme les performances d'un tel circuit reposent sur des paramètres aléatoires et statistiques,

l'homme de l'art s'assurera, notamment au moyen d'outils de simulation informatique, que le mode de réalisation particulier envisagé conduit bien aux performances recherchées.

Par ailleurs, les performances offertes par le circuit d'authentification selon l'invention permettent de l'utiliser de deux manières. La première manière consiste à injecter le code d'entrée CE et sortir le code d'authentification CA simultanément, en synchronisation avec l'horloge. La deuxième manière consiste à injecter d'abord le code d'entrée CE, puis à sortir le code d'authentification CA lorsque tout le code CE a été absorbé. Par cette deuxième méthode, même si deux codes d'entrée CE sont très ressemblants et ne diffèrent par exemple que d'un seul bit, les codes d'authentification produits seront très différents, ce qui augmente le degré d'inviolabilité du circuit d'authentification.

Enfin, selon un mode d'utilisation préféré, si l'on souhaite produire un code d'authentification CA de 32 bits à partir d'un code d'entrée CE de 32 bits et que le circuit d'authentification présente de l'ordre de 65000 possibilités en termes de transitions d'états internes, on propose de découper le code d'entrée CE en deux mots CE1, CE2 de 16 bits chacun, d'injecter d'abord CE1 (16 coups d'horloge), de produire un premier code d'authentification CA1 de 16 bits (16 autres coups d'horloge), puis d'injecter CE2 (16 coups d'horloge) et produire ensuite un deuxième code d'authentification CA2 de 16 bits (16 autres coups d'horloge), le code d'authentification final étant obtenu par chaînage de CA1 et CA2.

Microcircuit permettant de mettre en oeuvre un procédé d'authentification mutuelle entre une carte à mémoire et un terminal

On rappelle qu'un autre objectif de la présente invention est de prévoir des moyens pour empêcher des interrogations multiples d'une carte à mémoire.

La figure 5 représente le schéma électrique d'un microcircuit 50 à logique câblée de carte à mémoire pouvant résister aux interrogations multiples. Le microcircuit 50 comprend, comme le microcircuit classique déjà décrit en relation avec la figure 1, une mémoire de données 51 à lecture série (c'est-à-dire bit par bit) comprenant le numéro de série NI de la carte et des données DA de transaction. Egalement, on retrouve un circuit d'authentification 52, un séquenceur logique 53 assurant la commande des divers éléments du circuit 50, et un plot d'entrée-sortie I/O. Le circuit d'authentification 52 peut être classique ou réalisé comme proposé précédemment.

Selon l'invention, le circuit 50 comprend en outre un générateur 54 de mots binaires aléatoires, par exemple un registre à décalage dont des bits internes sont ramenés sur son entrée par une porte OU EXCLUSIF, un comparateur logique série 55 à deux entrées E1, E2, trois basculeurs logiques 56, 57, 58, un interrupteur logique 59 et une porte logique ET 60. Le basculeur logique 56 est un basculeur à trois positions et présente trois plots d'entrée P1, P2, P4 et un plot de sortie PS. Le basculeur 57 est un basculeur à quatre positions et présente quatre plots d'entrée P2, P3, P4, P5 et un plot de sortie PS. Le basculeur 58 est un basculeur à deux positions et présente deux plots d'entrée P3, P5 et un plot de sortie PS. Enfin, l'interrupteur 59 présente un plot d'entrée P1 et un plot de sortie PS.

La sortie de la mémoire 51 est connectée aux plots P1 du basculeur 56 et de l'interrupteur 59, la sortie du générateur aléatoire 54 aux plots P2 des basculeurs 56 et 57, et la sortie du circuit d'authentification 52 au plot

P3 du basculeur 58. Les entrées E1, E2 du comparateur 55 sont connectées aux plots P3 des basculeurs 57 et 58, les entrées de la porte ET 60 sont connectées au plot P5 du basculeur 58 et à la sortie du comparateur 55. La sortie de la bascule ET 60 est connectée au plot P5 du basculeur 57 dont le plot PS est connecté au plot d'entrée-sortie I/O du microcircuit 50. Enfin, le plot PS du basculeur 56 est connecté à l'entrée du circuit d'authentification 52, les plots P4 des basculeurs 56 et 57 sont connectés ensemble, et le plot PS de l'interrupteur 59 est connecté au plot P2 du basculeur 57.

Selon l'invention, le séquenceur logique 53 est câblé pour qu'une procédure d'authentification d'un terminal, par exemple le terminal 10 de la figure 1, soit réalisée par le microcircuit 50 dès la réception des premiers coups d'horloge qui suivent l'introduction de la carte dans le terminal 10. Bien entendu, cette procédure d'authentification ne nécessite pas de modification de la structure du terminal 10, qui doit simplement être programmé pour répondre au microcircuit 50 comme on le décrira maintenant.

Authentification du terminal

- Etape 1 : Le basculeur 56 et l'interrupteur 59 sont mis en position P1/PS et le basculeur 57 en position P2/PS. Les données NI et DA de la mémoire 51 sont envoyées en série dans le circuit d'authentification 52 et sur le plot I/O.
- Etape 2 : l'interrupteur 59 est ouvert, les basculeurs 56 et 57 sont mis en position P2/PS. Le générateur aléatoire 54 génère un code aléatoire ALINT (une suite aléatoire de "1" et de "0") qui est envoyé dans le circuit d'authentification 52 et sur le plot I/O.
- Etape 3 : les interrupteurs 57 et 58 sont mis en position P3/PS. Le circuit d'authentification 52 produit un code série CA de la forme

$$CA = F_{Ks} (NI, DA, ALINT)$$

NI, DA et ALINT étant les données chaînées formant le code d'entrée CE à partir desquelles est généré le code d'authentification CA. Au fur et à mesure de sa production, le code CA est appliqué sur l'entrée E1 du comparateur logique série 55. Au même instant, c'est-à-dire en synchronisation, l'entrée E2 du comparateur 55 reçoit un code série CA' envoyé par le terminal 10 par l'intermédiaire du plot I/O et du basculeur 57.

L'étape 4 décrite ci-après commence lorsque les deux codes série CA et CA' ont été injectés dans le comparateur 55. A cet instant, la sortie du comparateur 55 est à 1 si les deux codes CA et CA' étaient identiques ou à 0 si les deux codes étaient différents.

Authentification de la carte

- Etape 4 : les basculeurs 56 et 57 sont mis en position P4/PS, l'entrée du circuit d'authentification 52 étant ainsi reliée au plot d'entrée-sortie I/O. On entre dans une procédure classique d'authentification de la carte bien connue de l'homme de l'art et déjà décrite au préambule. Etant donné que cette procédure a été précédée des étapes 1 à 3 selon l'invention, elle peut se décliner en plusieurs variantes selon que l'on décide de remettre à zéro le circuit d'authentification 52 ou de le laisser dans l'état logique interne dans lequel il est au terme de l'étape 3.

(a) Si l'on remet le circuit 52 à zéro, on peut prévoir que le terminal 10, qui a déjà lu les données NI et DA au cours de l'étape 1, renvoie NI et DA dans le circuit d'authentification 52 accompagnées d'un code aléatoire ALEXT qu'il a lui-même produit.

(b) Si l'on ne remet pas à zéro le circuit d'authentification 52, le terminal n'envoie que ALEXT, il

n'est pas nécessaire de renvoyer NI et DA dans le circuit d'authentification 52 qui les a déjà reçues à l'étape 1 (on rappelle que le fait d'injecter NI et DA est optionnel et permet simplement, pour des raisons de sécurité, de placer le circuit d'authentification 52 dans un état logique interne le plus éloigné que possible de son état logique "zéro", c'est-à-dire son état logique après remise à zéro des circuits qui le constituent).

- Etape 5 : on suppose que l'option (a) ci-dessus a été choisie. Les basculeurs 57 et 58 sont en position P5/PS. La sortie du circuit d'authentification est reliée au plot d'entrée sortie I/O par l'intermédiaire de la porte ET 60, et délivre vers le terminal 10 un code d'authentification CA de la forme

$$CA = F_{Ks} (NI, DA, ALEXT)$$

qui permettra au terminal 10 de vérifier l'authenticité de la carte.

Toutefois, si la sortie du comparateur 55 est passée à 0 au cours de l'étape 3, la porte ET ne sera pas passante et le code CA émis par le circuit d'authentification 52 n'atteindra pas le terminal 10 qui est considéré comme non authentique. En effet, le terminal 10 ayant reçu du microcircuit 50 les paramètres NI, DA et ALINT au cours des étapes 1 et 2 aurait dû être en mesure de délivrer au cours de l'étape 3 un code CA' en tout point identique au code CA.

On voit ainsi que grâce à la présente invention, la carte ne va émettre un code d'authentification que si elle a elle-même authentifié préalablement le terminal qui l'interroge. Aucune interrogation à répétition de la carte ne pourra donc être faite par une personne non habilitée.

A titre d'exemple, la figure 9 représente un mode de réalisation du comparateur série 55. Le comparateur 55 comprend en entrée une porte OU EXCLUSIF 61 recevant les codes CA et CA'. La sortie de la porte 61 est appliquée sur l'entrée d'une porte NON ET 62 dont l'autre entrée reçoit un signal COMP. La sortie de la porte 62 est appliquée sur un première entrée d'une bascule à mémoire 63 de type NON ET comprenant de façon classique deux portes NON ET 64, 65 dont les sorties sont réciproquement ramenées sur les entrées. L'autre entrée de la bascule à mémoire 63 reçoit un signal /SET de remise à 1 de sa sortie. La sortie de la bascule 63 forme la sortie du comparateur 55.

Avant l'étape 3, le signal /SET est mis à 0 un court instant pour s'assurer que la sortie du comparateur est à 1. Pendant l'étape 3, quand CA est comparé à CA', le signal COMP est mis à 1. Si au cours de l'étape 3 les codes CA et CA' présentent deux bits différents, ou plus, la sortie du comparateur 55 passe à 0 et reste définitivement à 0, de sorte que la porte ET sera bloquée au cours des étapes suivantes. Pour remettre la sortie du comparateur à 1 (/SET mis à 0), il faudra remettre à zéro le séquenceur 53 et recommencer les étapes 1, 2 et 3 précédemment décrites, de sorte qu'un code d'authentification CA émis par la carte ne pourra jamais être lu sur le plot I/O tant que l'entité qui questionne ne se sera pas préalablement habilitée auprès de la carte.

Bien entendu, d'autres moyens de blocage, total ou partiel, du microcircuit 50 peuvent être prévus par l'homme de l'art en vue d'empêcher la carte d'émettre un code d'authentification quand le terminal ne s'est pas préalablement habilité. Toutefois, la solution préférée de l'invention, qui consiste à bloquer la sortie du circuit d'authentification 52, présente l'avantage d'être

particulièrement simple à mettre en oeuvre et de ne pas nécessiter l'intervention du séquenceur câblé 53.

Procédé pour authentifier un utilisateur de carte à mémoire

On a vu au préambule en relation avec la figure 1 que la prévision d'un procédé d'identification de l'utilisateur au moyen d'un code secret CS nécessitait, dans un microcircuit 1 à logique câblée, l'ajout de moyens de test 5 coûteux et encombrants.

La présente invention propose une solution particulièrement simple et de grande sécurité qui peut être mise en oeuvre moyennant uniquement une modification de la logique câblée du séquenceur d'un microcircuit.

Plus précisément, et si l'on se réfère par exemple à la figure 1, la présente invention propose tout d'abord de disposer le code secret CS de la carte dans un zone de la mémoire 2 non accessible par le terminal, représentée par une zone hachurée.

Ensuite, la présente invention prévoit de chaîner le code secret CS avec les autres données formant le code d'entrée CE, de telle sorte que le code d'authentification CA soit fonction du code secret CS.

De façon générale, le procédé de vérification du code secret CS' de l'utilisateur selon la présente invention peut être mis en oeuvre :

(1) soit au cours de la procédure classique d'authentification de la carte par le terminal 10 telle qu'elle a été décrite au préambule en relation avec la figure 1.

(2) soit au cours d'une procédure d'authentification du terminal 10 par la carte, telle qu'elle a été proposée plus haut en relation avec la figure 8.

(3) enfin, le procédé selon l'invention peut être mis en oeuvre indépendamment des procédures d'authentification, c'est-à-dire après qu'elles ont été réalisées, bien que

cela ne présente pas un grand intérêt pratique puisqu'il est plus économique de vérifier le code secret de l'utilisateur pendant que s'effectue la vérification de l'authenticité de la carte (1) ou l'authenticité du terminal 10 (2).

(1) Vérification du code secret CS' pendant la procédure classique d'authentification de la carte par le terminal

Dans ce cas, et en se référant à la figure 1, le code d'entrée CE injecté dans le circuit d'authentification 3 va comprendre :

- le mot aléatoire ALEXT envoyé par le terminal 10,
- le code secret CS de la carte lue dans la mémoire 2 (zone hachurée) et envoyé directement dans le circuit d'authentification (la zone hachurée contenant CS pouvant bien entendu être lue par le microcircuit 1 mais pas par le terminal 10),
- éventuellement, le numéro de série NI et des données de transaction DA,

Le code d'authentification émis par la carte est alors le suivant :

$$CA = F_{Ks} (NI, DA, ALEXT, CS)$$

Le code calculé par le terminal 10 est le suivant :

$$CA' = F_{Ks} (NI, DA, ALEXT, CS')$$

CS' étant le code secret donné au terminal 10 par l'utilisateur.

La comparaison des deux codes CA et CA' est effectuée par le terminal 10. Si le terminal constate que les deux codes sont différents, c'est peut être parce que le code CS' donné par l'utilisateur est mauvais ou parce que la carte est falsifiée. Il n'est pas nécessaire que

le terminal 10 discerne la cause exacte, il suffit qu'il interrompe la transaction. Eventuellement, le terminal peut demander à l'utilisateur de composer à nouveau le code secret CS' et relancer une procédure d'authentification.

(2) Vérification du code secret CS' pendant la procédure d'authentification du terminal par la carte

La procédure d'authentification du terminal 10 par la carte a été décrite plus haut en relation avec la figure 8, à laquelle on se référera à nouveau.

Dans ce cas, le code d'entrée CE injecté dans le circuit d'authentification 52 comprend :

- le mot aléatoire ALINT émis par le générateur aléatoire 54 du microcircuit 50,
- le code secret CS de la carte envoyé directement dans le circuit d'authentification 52 depuis la mémoire 2 (zone hachurée),
- éventuellement, le numéro de série NI et les données de transaction DA,

Le code d'authentification CA envoyé par le circuit d'authentification 52 sur l'entrée E1 du comparateur logique 55 est alors le suivant :

$$CA = F_{Ks} (NI, DA, ALINT, CS)$$

Et le code calculé par le terminal 10 et envoyé sur l'entrée E2 du comparateur logique 55 est le suivant :

$$CA' = F_{Ks} (NI, DA, ALINT, CS')$$

CS' étant le code secret donné au terminal par l'utilisateur.

La comparaison des deux codes CA et CA' est effectuée par le comparateur 55. Si les deux codes sont

différents le comparateur 55 passe à 0 et la porte logique 60 sera bloquée lors de la tentative par le terminal 10 de tester l'authenticité de la carte. Le terminal 10 saura donc que le code CS' était mauvais quand il ne recevra pas de réponse de la carte après l'avoir interrogée.

(3) Vérification du code secret CS' indépendamment des procédures d'authentification

La démarche à suivre est la même qu'en (1) et ne sera pas décrite à nouveau, la seule différence étant que la ou les procédures d'authentification (de la carte par le terminal et du terminal par la carte) sont déjà effectuées quand la vérification commence.

Bien entendu, de nombreuses autres variantes de ce procédé peuvent être prévues par l'homme de l'art. En particulier, on pourrait prévoir une procédure semblable à celle décrite en (1) mais dans laquelle le code ALEXT serait remplacé par un code ALINT émis par la carte, ou une procédure semblable à (2) dans laquelle ALINT serait remplacé par un code ALEXT émis par le terminal.

Egalement, la carte pourrait produire un code CA :

$$CA = F_{Ks} (CS) ,$$

et le terminal un code CA' :

$$CA' = F_{Ks} (CS')$$

Enfin, on peut prévoir que la zone mémoire où se trouve le code secret devienne accessible en écriture une fois l'utilisateur habilité (pour des raisons de sécurité, il est préférable qu'elle soit interdite en lecture de façon permanente). En effet, l'utilisateur, après s'être habilité, peut souhaiter inscrire un nouveau code secret dans la carte. Toutefois, il est évident que,

toujours pour des raisons de sécurité, la zone mémoire du code secret ne doit pas être rendue accessible en écriture pour inscription d'un nouveau code tant que l'utilisateur ne s'est pas habilité avec l'ancien code secret.

Dans un souci de simplicité, on ne décrira pas dans la présente demande de brevet les mécanismes logiques permettant de rendre une zone mémoire inaccessible en lecture et/ou en écriture de façon définitive ou provisoire, qui sont du ressort de l'homme de l'art et sont basés sur un contrôle logique des valeurs d'adresse correspondant à la zone mémoire considérée.

Dans ce qui précède, on a décrit

- un procédé d'authentification d'un terminal par une carte à mémoire permettant d'empêcher les interrogations à répétition de la carte par une personne non habilitée,
- un procédé d'authentification de l'utilisateur, faisant intervenir le circuit d'authentification et permettant d'éviter l'installation de moyens matériels coûteux dans le microcircuit d'une carte à mémoire,
- une structure de circuit d'authentification performante.

L'homme de l'art notera que ces trois aspects de la présente invention sont indépendants et peuvent être réalisés chacun indépendamment de l'autre, ou être combinés, comme cela a été décrit. De plus, les deux procédures d'authentification de la présente invention, bien qu'elles puissent être réalisées au moyen de circuits d'authentification de l'art antérieur, peuvent avantageusement être mises en oeuvre au moyen du circuit d'authentification selon l'invention.

D'autre part, bien que l'on ait utilisé au cours de la description qui précède le terme de "carte à mémoire" dans le souci d'en faciliter la compréhension et de clarté du texte, il est bien évident que la présente

invention n'est pas destinée à une seule application au domaine des cartes plastiques, mais concerne de façon générale tout support portable susceptible de recevoir une puce.

REVENDICATIONS

1. Procédé pour produire un code d'authentification (CA), comprenant de façon cyclique les étapes consistant à lire un mot binaire (M_n) dans une mémoire secrète (21) comportant une pluralité de mots binaires, procédé caractérisé en ce que, à chaque cycle, il est prévu une opération de combinaison (Σ) des mots (M_1, M_2, ...M_n) lus au cours des cycles précédents, et en ce que le résultat de ladite combinaison est utilisé comme mot générateur (GA) de l'adresse du mot à lire au cycle suivant.

2. Procédé selon la revendication 1, caractérisé en ce que ladite opération de combinaison (Σ) inclut le mot (M_n) qui vient d'être lu dans la mémoire.

3. Procédé selon l'une des revendications 1 et 2, caractérisé en ce que ladite opération de combinaison consiste à faire l'addition des mots binaires (M_1 à M_n) lus dans la mémoire secrète (21).

4. Procédé selon l'une des revendications 1 à 3, caractérisé en ce qu'il comprend en outre une première opération de transformation dudit mot générateur d'adresse (GA), consistant à combiner entre eux, de façon logique, au moins une partie des bits (g0-g7) du mot (GA) générateur d'adresse.

5. Procédé selon l'une des revendications 1 à 4, caractérisé en ce qu'il comprend en outre une deuxième opération de transformation du mot (GA) générateur d'adresse, consistant à combiner, de façon logique, des bits (g'2,g'5,g'7) du mot (GA) générateur d'adresse avec des bits internes (r1,r4,r6) d'un registre à décalage (26).

6. Procédé selon l'une des revendications 1 à 5, caractérisé en ce qu'il comprend en outre une troisième opération de transformation du mot (GA) générateur d'adresse, consistant à réduire le nombre de bits (g0-g7) du mot générateur d'adresse (GA) dans une quantité

correspondant au nombre d'entrées d'adresse (a0,a1,a2) de la mémoire secrète (21).

7. Procédé selon l'une des revendications 1 à 6, caractérisé en ce que, à chaque cycle, un bit d'un code d'entrée (CE) est combiné, de façon logique, avec au moins un bit (g'0,g'1,g'2) du mot (GA) générateur d'adresse (GA).

8. Procédé selon l'une des revendications 1 à 7, caractérisé en ce que :

- au cours d'une phase d'initialisation du procédé, au moins un bit (g'0,g'1,g'2) du mot (GA) générateur d'adresse est combiné de façon logique, à chaque cycle, avec un bit d'un code d'entrée (CE), et
- au cours d'une phase de production du code d'authentification (CA), un bit (g2) du mot (GA) générateur d'adresse est prélevé, à chaque cycle, pour former un bit du code d'authentification (CA).

9. Procédé selon la revendication 8, caractérisé en ce que lesdites phases d'initialisation et de production du code d'authentification (CA) sont réalisées simultanément, le code d'authentification (CA) étant produit pendant que ledit code d'entrée (CE) est absorbé.

10. Procédé selon la revendication 8, caractérisé en ce que lesdites phases d'initialisation et de production du code d'authentification (CA) sont réalisées séquentiellement, la phase de production du code d'authentification commençant lorsque tous les bits du code d'entrée (CE) ont été absorbés au cours de la phase d'initialisation.

11. Procédé selon la revendication 10, caractérisé en ce que, pour produire un code d'authentification (CA) de grande longueur, on divise au moins en deux parties (CE1, CE2) le code d'entrée (CE), on absorbe une première partie (CE1) du code d'entrée, puis on produit une première partie (CA1) du code d'authentification (CA), on

absorbe la deuxième partie (CE2) du code d'entrée, puis on produit la deuxième partie (CA2) du code d'authentification (CA).

12. Procédé selon l'une des revendications 1 à 11, caractérisé en ce qu'il est réalisé sous forme de logique câblée (52) dans un microcircuit (50), et en ce que le code d'entrée (CE) utilisé pour produire le code d'authentification (CA) comprend un code aléatoire interne (ALINT) généré par le microcircuit (50,54).

13. Procédé selon l'une des revendications 1 à 12, caractérisé en ce qu'il est réalisé sous forme de logique câblée (52) dans un microcircuit (50), ledit microcircuit comprenant une mémoire (51) comportant une zone qui n'est accessible en lecture que par le microcircuit, ladite zone comprenant un code secret (CS) supposé connu de l'utilisateur légitime du microcircuit, et en ce que le code d'entrée (CE) utilisé pour produire le code d'authentification (CA) comprend ledit code secret (CS).

14. Machine logique (20, 20-1, 30, 40) cadencée par un signal d'horloge (H) et comprenant une mémoire secrète (21) dans laquelle est stockée une pluralité de mots binaires, caractérisée en ce que :

- la sortie de ladite mémoire (21) est appliquée sur une première entrée (A) d'un circuit logique (22) et la mémoire (21) est lue à chaque cycle d'horloge (H),
- la sortie (C) dudit circuit logique (22) est renvoyée à chaque cycle d'horloge sur une deuxième entrée (B) du circuit logique (22) par l'intermédiaire d'un circuit tampon (23),
- le circuit logique (22) réalise une combinaison (Fc, +) de ses deux entrées (A, B) et délivre à chaque cycle d'horloge un mot binaire (GA) générateur d'adresse envoyé sur l'entrée d'adresse (ADR) de la mémoire.

15. Machine (20, 30) selon la revendication 14, caractérisée en ce que ledit mot (GA) générateur

d'adresse est prélevé à la sortie dudit circuit logique (22).

16. Machine (20-1) selon la revendication 14, caractérisée en ce que ledit mot (GA) générateur d'adresse est prélevé à la sortie dudit circuit tampon (23).

17. Machine selon l'une des revendications 14 à 16, caractérisée en ce que ledit circuit logique (22) est un circuit additionneur.

18. Machine selon l'une des revendications 14 à 17, caractérisée en ce qu'elle comprend des moyens logiques (25) pour réduire le nombre de bits (g0-g7) du mot générateur d'adresse (GA).

19. Machine selon l'une des revendications 14 à 18, caractérisée en ce qu'elle comprend des moyens logiques (24) pour combiner entre eux des bits (g0-g7) du mot générateur d'adresse (GA).

20. Machine (30) selon l'une des revendications 14 à 19, caractérisée en ce qu'elle comprend un registre à décalage (26), et des moyens logiques (25-1, 27) pour combiner au moins un bit (r1, r4, r6) du registre à décalage (26) avec au moins un bit (g'0,g'1,g'2) du mot (GA) générateur d'adresse.

21. Circuit d'authentification (40) de type entrée série et sortie série, pour produire un code d'authentification (CA) à partir d'un code d'entrée (CE), caractérisé en ce qu'il comprend une machine logique (30) selon l'une des revendications 14 à 20, des moyens logiques (25-1,27) pour injecter, à chaque cycle d'horloge, un bit du code d'entrée (CE) dans la machine logique, l'injection du bit du code d'entrée consistant à combiner au moins un bit (g'0,g'1,g'2) du mot (GA) générateur d'adresse (GA) avec ledit bit, et des moyens pour extraire un bit (g2) de la machine logique afin de produire ledit code d'authentification (CE).

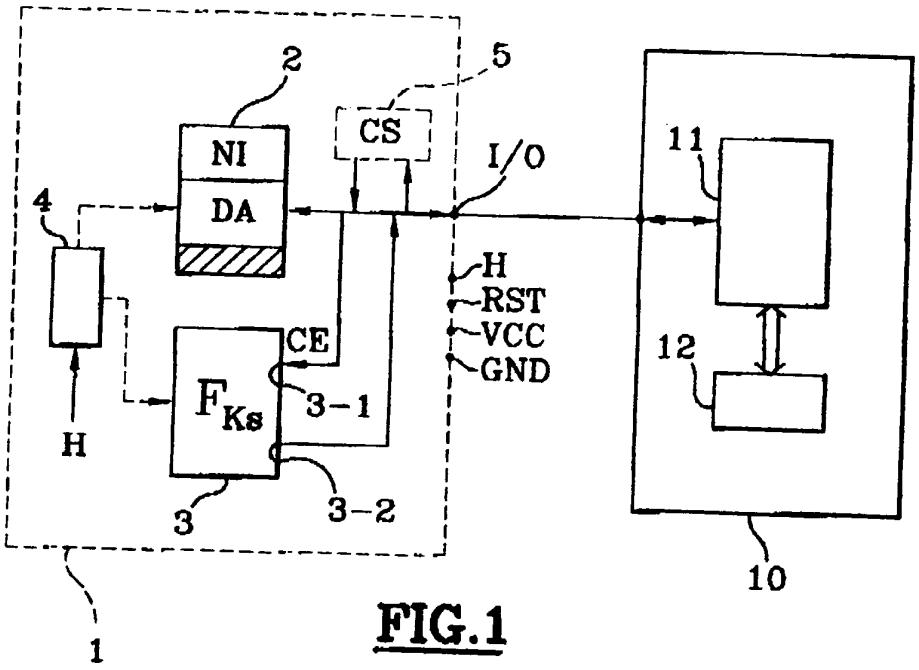


FIG.1

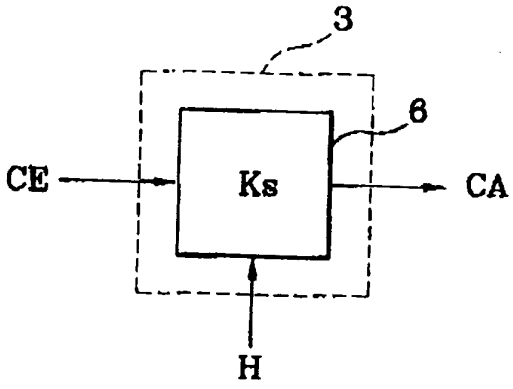


FIG.2

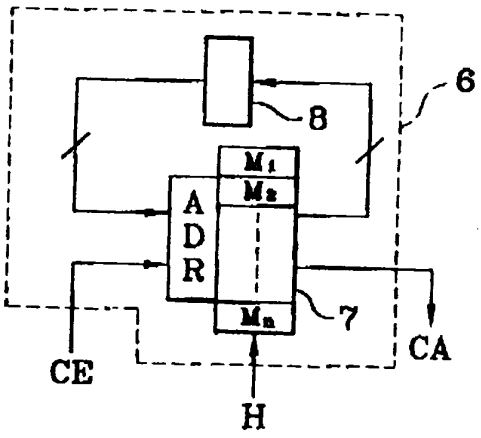


FIG.3

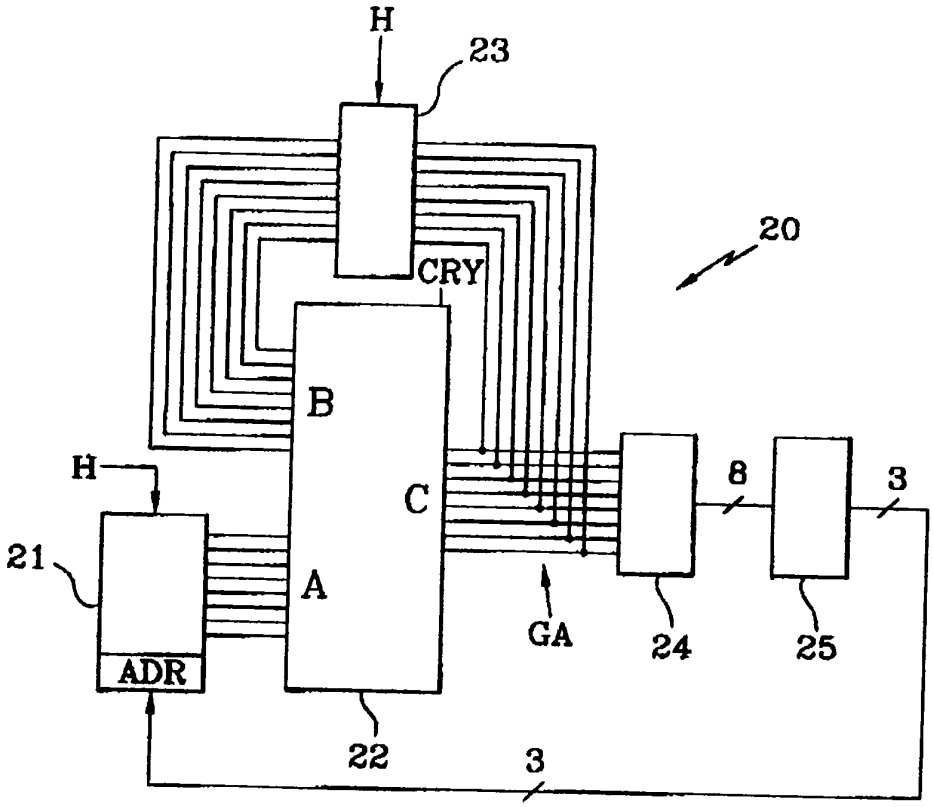

H
23
20
CRY
B
C
A
H
21
ADR
GA
24
25
8
3
22
3

FIG.4

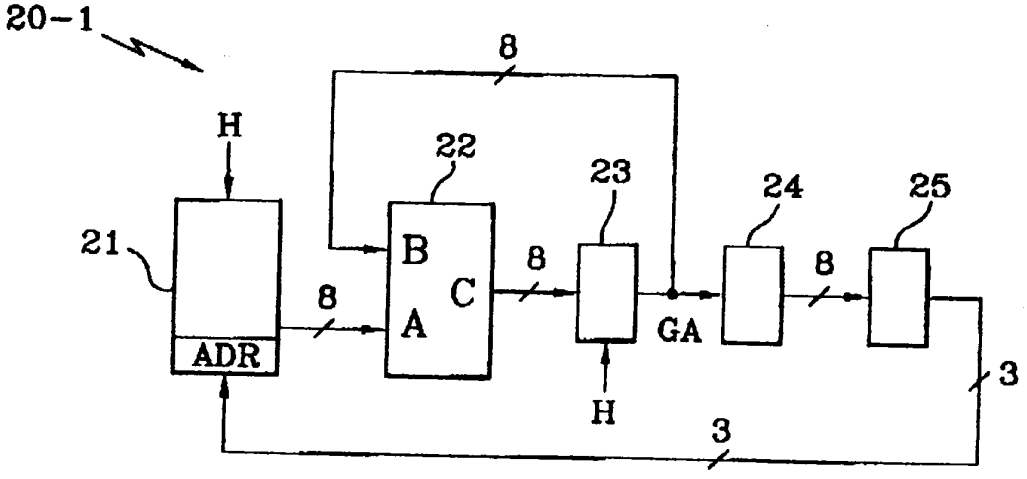

20-1
8
H
22
23
24
25
21
B
C
A
8
8
8
8
ADR
GA
H
3
3

FIG.5

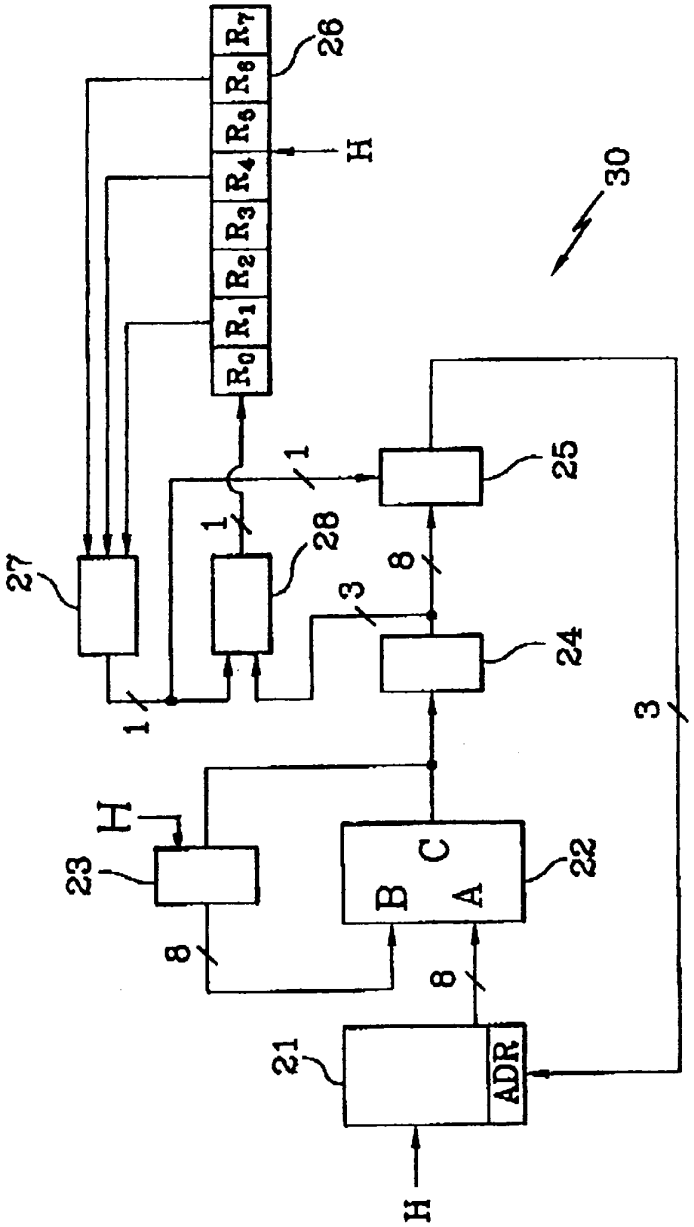
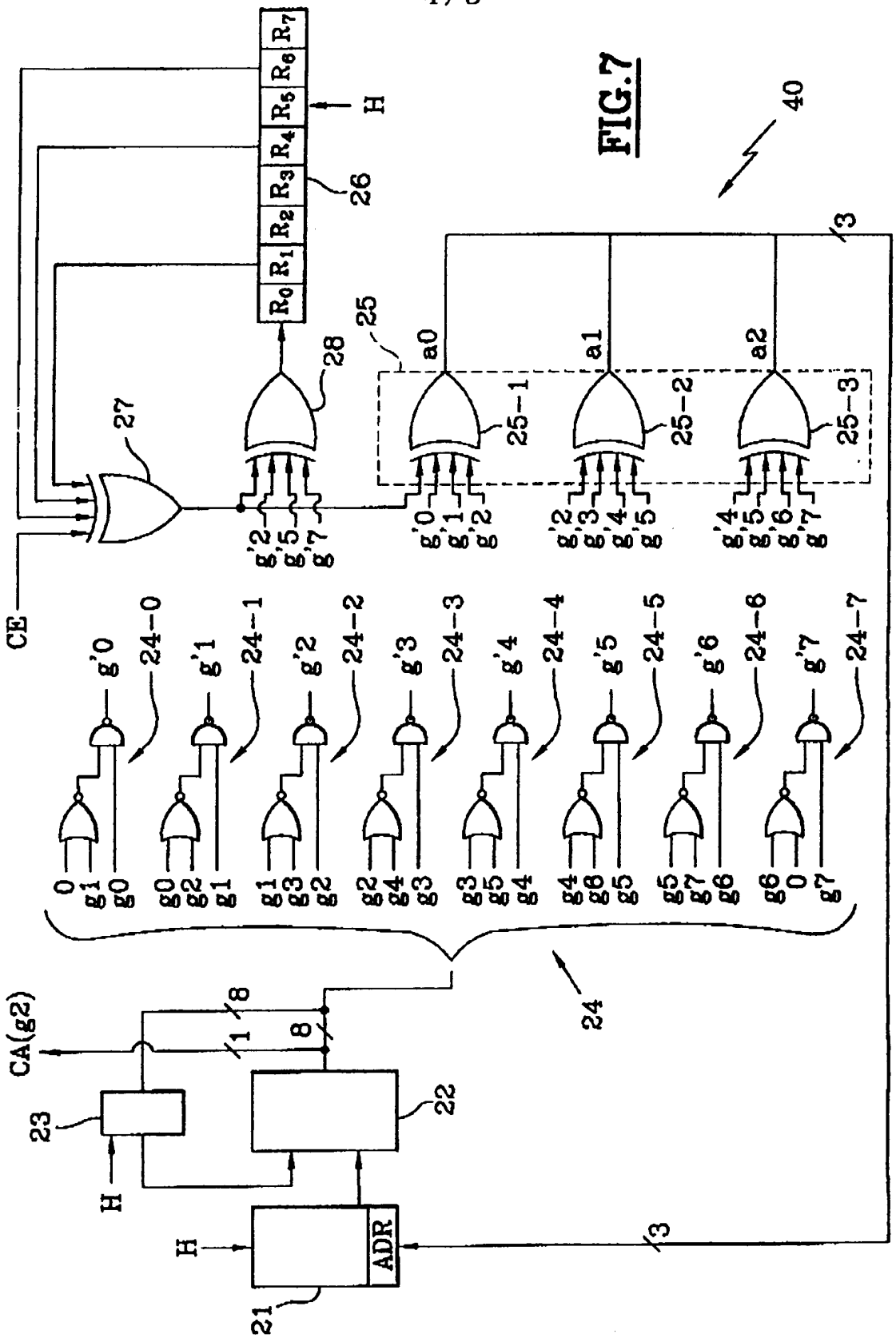

R0
R1
R2
R3
R4
R5
R6
R7
26
H
30
25
1
1
28
27
8
3
24
1
3
H
23
B
C
A
22
8
8
21
ADR
H

FIG.6

2739706

FIG.7



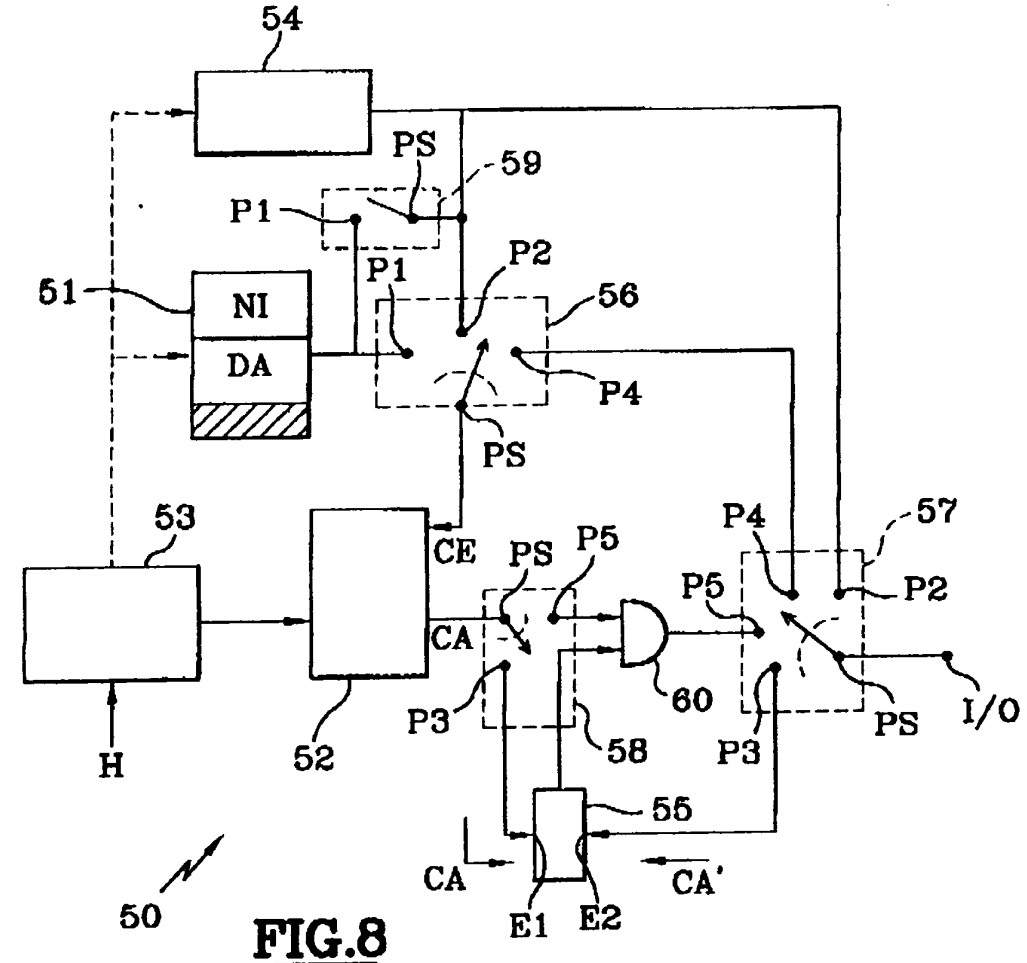

54
PS
59
P1
P1
P2
56
51
NI
DA
P4
PS
53
CE
PS
P5
P4
57
P5
P2
CA
60
I/O
P3
58
P3
PS
H
52
55
CA
CA'
E1
E2
50

FIG.8

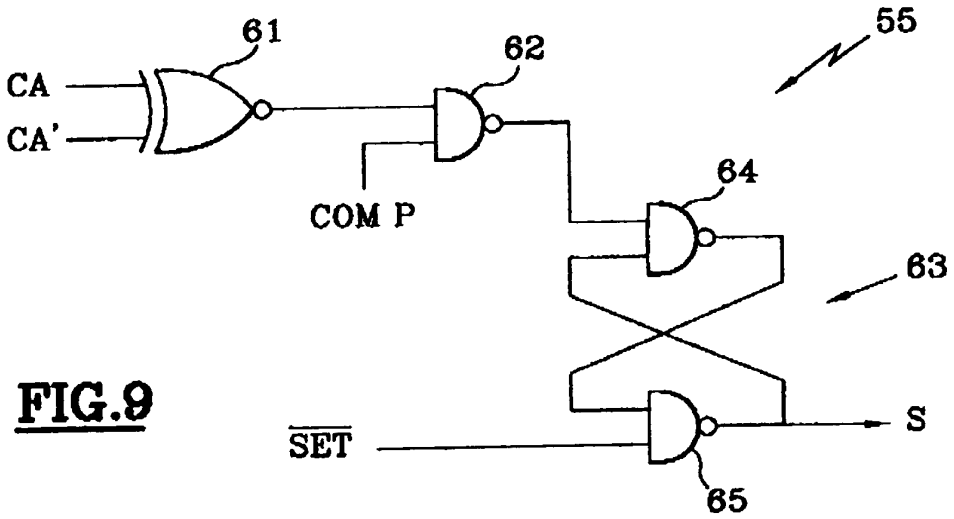

61
62
55
CA
CA'
64
COM P
63
S
SET
65

FIG.9

REPUBLIQUE FRANÇAISE

INSTITUT NATIONAL
de la
PROPRIETE INDUSTRIELLE

RAPPORT DE RECHERCHE PRELIMINAIRE

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

2739706
N° d'enregistrement national

FA 519314
FR 9512176

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
D,A	FR-A-2 698 195 (GEMPLUS CARD INTERNATIONAL) * abrégé; revendications; figures *	1,2,7,8, 12-15,21
D,A	EP-A-0 409 701 (ÉTAT FRANCAIS)	
A	FR-A-2 471 003 (ÉLECTRONIQUE MARCEL DASSAULT)	
A	FR-A-2 164 939 (IBM)	

DOMAINES TECHNIQUES RECHERCHES (Int.Cl.6)

G07F
H04L

Date d'achèvement de la recherche: 26 Juillet 1996

Examinateur: David, J

CATEGORIE DES DOCUMENTS CITES

X : particulièrement pertinent à lui seul
Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie
A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général
O : divulgation non-écrite
P : document intercalaire

T : théorie ou principe à la base de l'invention
E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure.
D : cité dans la demande
L : cité pour d'autres raisons

& : membre de la même famille, document correspondant

EPO FORM 1503 03.82 (P04C13)