(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 375 484**

(51) Int. CI.:
G06F 1/00 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Número de solicitud europea: **05771823 .1**
(96) Fecha de presentación: **27.06.2005**
(97) Número de publicación de la solicitud: **1761835**
(97) Fecha de publicación de la solicitud: **14.03.2007**

(54) Título: **MÓDULO DE SEGURIDAD Y MÉTODO DE PERSONALIZACIÓN DE TAL MÓDULO DE SEGURIDAD.**

(30) Prioridad:
29.06.2004 EP 04103053

(45) Fecha de publicación de la mención BOPI:
01.03.2012

(45) Fecha de la publicación del folleto de la patente:
01.03.2012

(73) Titular/es:
NAGRAVISION S.A.
ROUTE DE GENÈVE 22-24
1033 CHESEAUX-SUR-LAUSANNE, CH

(72) Inventor/es:
STRANSKY, Philippe

(74) Agente: **Tomas Gil, Tesifonte Enrique**

DESCRIPCIÓN

Módulo de seguridad y método de personalización de tal módulo de seguridad.

[0001] La presente invención se refiere al ámbito de los módulos de seguridad protegidos comprendiendo al menos un microprocesador y una memoria programa. La invención se refiere también a la personalización de tal módulo de seguridad así como a la identificación de un módulo de seguridad cuyo contenido se ha hecho público.

[0002] Estos módulos de seguridad se utilizan en sistemas que ponen en obra operaciones criptográficas y que se entregan en forma monobloque. Estos se realizan en un único chip de silicio, o bien se ensamblan en un soporte y se introducen en una resina o son protegidos por una hoja, la cual cubre los diversos elementos y actúa como fusible en caso de intento de intrusión.

[0003] Estos módulos protegidos tienen una memoria programa que contiene en particular un programa de arranque y uno o unos programas operativos. El programa de arranque se ejecuta durante la activación del procesador o en cada reiniciación (reset). Este programa de arranque se almacena en una memoria de tipo ROM es decir que el acceso sólo se puede realizar por lectura.

[0004] El programa operativo se almacena en una memoria de tipo reinscribible, en general de tipo EEPROM, NVRAM o Flash.

[0005] Cuando el programa de arranque termina su verificación, inicia la ejecución del programa operativo en una dirección acordada.

[0006] Uno de los ataques conocidos para descubrir el contenido de la memoria de un módulo de seguridad consiste en buscar un fallo de seguridad tal como un desbordamiento de memoria que permitiría tomar el control del procesador. Una vez conseguida dicha toma de control, se puede transferir el contenido de la memoria hacia el exterior y analizar el mecanismo de seguridad y las claves utilizadas.

[0007] Gracias al conocimiento del contenido de la memoria, se puede obtener las claves que sirven para gestionar los diferentes derechos y accesos a los servicios controlados por el procesador. De este modo, si se produce un cambio de claves, ordenado por el centro de gestión, esta orden de cambio será codificada por una clave presente en la memoria programa. Al disponer de esta clave, se podrá descodificar el mensaje de cambio de clave y actualizar también el contenido de esta nueva clave.

[0008] Se comprueba entonces que, cuando una persona malintencionada ha violado una vez la seguridad de un módulo de seguridad, todos los cambios iniciados por el centro de gestión no tienen ningún efecto sobre la seguridad ya que los medios de cambio (nueva clave de transmisión por ejemplo) utilizan claves que ya están en posesión de dicha persona. La cual puede así descodificar el mensaje de actualización y cambiar también su clave de transmisión.

[0009] Cuando se ha violado la seguridad de un módulo de seguridad y que se conoce así el contenido de la memoria programa, la persona malintencionada que ha violado la seguridad de este módulo puede publicar los códigos informáticos correspondientes al contenido de la memoria programa, esta publicación siendo realizada particularmente en una red tal como Internet. Eso permite a terceros que poseen tarjetas vírgenes copiar dichos códigos y crear así tarjetas clones perfectamente funcionales, de manera totalmente ilegal.

[0010] Uno de los medios que permiten limitar estas actividades ilegales consiste en aumentar la seguridad de los módulos, de modo que resulte particularmente difícil violar la seguridad de este módulo.

[0011] Otro medio para limitar en gran medida estas actividades ilegales consiste en detectar el módulo de seguridad cuya seguridad ha sido violada y que ha permitido la clonación, y en actuar sobre este módulo mediante la desactivación de éste y de los clones que ha permitido realizar.

[0012] El documento US 6,725, 374 describe un módulo de seguridad que utiliza el primer medio mencionado anteriormente, es decir la mejora de la seguridad con respecto a los módulos anteriores. De hecho, en el módulo descrito en esta patente, el descubrimiento de claves se vuelve más difícil gracias a la adición, en el código informático del módulo, de elementos de "interferencia" que interfieren en las informaciones que pueden servir para extraer las claves, a saber el consumo eléctrico. Estos elementos de interferencia se forman a partir de módulos cuyo orden de ejecución no tiene importancia para el desarrollo del programa. Estos elementos se utilizan de manera aleatoria, de modo que el tratamiento de dos señales de entrada idénticas no produzca dos señales de salida idénticas. A pesar de esta dificultad adicional, si una persona consigue determinar el contenido del módulo de seguridad, este código podrá ser publicado y reutilizado por terceros, sin que sea posible reencontrar la fuente del código publicado.

[0013] La presente invención se propone utilizar el segundo medio mencionado anteriormente, es decir que propone introducir en el módulo un medio que permite la detección del módulo usado en un fraude.

[0014] De manera bien conocida, cada módulo de seguridad incluye un número de identificación único. Habitualmente, las personas capaces de extraer los códigos informáticos de un módulo de seguridad son también capaces de detectar

el número único de su módulo, a partir de un análisis relativamente sumario del contenido de este módulo. Este número único no se publica durante la publicación de los códigos informáticos.

[0015] Esto impide por un lado que se identifique la persona malintencionada y por otra parte que se desactive el módulo de origen y sus clones.

[0016] El objeto de la presente invención es proponer un método y un módulo de seguridad que comprende los medios de identificación del módulo de seguridad durante la publicación ilegal del código de este módulo, incluso si el tercero malintencionado ha retirado el identificador de este módulo. En la presente invención, la lucha contra la clonación de módulos de seguridad por lo tanto no consiste en mejorar la seguridad de estos módulos, sino en facilitar la detección de los módulos que han servido para la clonación, de modo que estos módulos se vuelvan inoperantes.

[0017] La patente europea EP 1 178 406 describe un método en el que un número de serie único de un circuito impreso se memoriza en una memoria. En esta invención, el número de serie es en primer lugar leído a partir de un código barras, y luego convertido en informaciones numéricas. Estas informaciones son eventualmente cifradas antes de ser introducidas en una o varias memorias. El objetivo de la invención es por un lado que la detección del número de serie sea más difícil y por otra parte impedir que una persona no autorizada conozca y modifique este número de serie. Para ocultar el número de serie, se memoriza en una memoria de gran tamaño, de modo que sea difícil localizarlo entre todas las otras informaciones memorizadas. Para impedir conocer y modificar el número, éste es cifrado.

[0018] El hecho de ocultar el número de serie no permite resolver de manera satisfactoria el problema de la invención. De hecho, el número de serie se memoriza en forma de valor en un emplazamiento dado de la memoria. Si una persona o un grupo de personas determina el emplazamiento del número de serie, este emplazamiento se podrá hacer público. Durante la publicación del código informático necesario para realizar un módulo de seguridad clonado, por lo tanto sólo será necesario evitar la publicación del contenido de este emplazamiento para evitar que su módulo de seguridad sea detectado.

[0019] El objetivo de la invención se alcanza por un módulo de seguridad que incluye un microprocesador, una memoria programa comprendiendo al menos un programa operativo y un medio de identificación único de dicho módulo, caracterizado por el hecho de que este medio de identificación se constituye de un conjunto de códigos informáticos artificial, compatible con su ejecución por dicho microprocesador del módulo y almacenado en la memoria programa.

[0020] Este objetivo se alcanza también por medio de un método de personalización de un módulo de seguridad por un identificador único, este módulo comprendiendo un microprocesador y una memoria programa que contiene al menos un programa operativo, caracterizado por el hecho de que incluye las etapas siguientes de:

- generación de un conjunto único de códigos informáticos, llamados códigos informáticos artificiales, compatibles con su ejecución por dicho microprocesador;

- escritura de este conjunto de códigos en la memoria programa en emplazamientos memoria específicos.

[0021] El objetivo de la invención se alcanza también con un método de identificación de un módulo de seguridad como definido en cualquiera de las reivindicaciones 1 a 6 y cuyos códigos informáticos se han vuelto accesibles al público, este método comprendiendo las etapas de:

- extracción de los códigos informáticos artificiales entre los códigos informáticos que se han vuelto accesibles al público;

- tratamiento de dichos códigos informáticos artificiales según unas reglas predefinidas de manera a deducir el medio de identificación de dicho módulo de seguridad.

[0022] El método de personalización de la invención tiene como principal ventaja que un tercero malintencionado considere que los códigos informáticos artificiales forman parte del programa y resultan por lo tanto necesarios para la reproducción de un módulo clon.

[0023] Estos códigos informáticos artificiales se introducen en el programa operativo, de modo que resulta difícil determinar cuáles son las informaciones realmente necesarias para el buen funcionamiento del módulo y cuáles son las que se utilizan para generar el número de identificación.

[0024] El módulo de seguridad según la invención y el método asociado permiten incitar a una persona malintencionada que ha publicado los códigos informáticos de un módulo de seguridad pirateado, publicar también las informaciones que permiten determinar el número o un número de identificación único del módulo de seguridad. Gracias a esto, es relativamente fácil determinar la procedencia del módulo de seguridad de origen. A partir de ahí, existen métodos que permiten que este módulo de origen se vuelva inoperante al igual que los clones que éste ha permitido realizar. Uno de estos métodos se describe por ejemplo en la solicitud de la patente europea EP 04100969.7 del mismo solicitante.

[0025] La invención se comprenderá mejor gracias a la descripción detallada siguiente que se refiere a los dibujos anexos que se dan a modo de ejemplo no limitativo, en los cuales:

- la figura 1 ilustra de manera general un módulo de seguridad según la presente invención;

- la figura 2 representa un primer modo de realización de una parte del módulo de seguridad de la figura 1;

- la figura 3 ilustra un segundo modo de realización del módulo de seguridad de la figura 1 y

- la figura 4 ilustra un modo de realización particular del método de la invención.

[0026] En referencia a la figura 1, el módulo de seguridad SM es un módulo procesador protegido. En virtud de lo cual, dispone de al menos un microprocesador CPU y de una memoria programa comprendiendo particularmente un programa operativo. En el modo de realización representado, la memoria programa contiene una primera zona Z1 de arranque y una segunda zona Z2 de dicha zona de trabajo. La primera zona de arranque se constituye totalmente o en parte de memoria ROM y por lo tanto no reinscribible. Puede que una parte incluya emplazamientos memoria en RAM o EEPROM, entre otros para unas variables. Esta se llama "de arranque" debido a que es la primera que se ejecuta durante la puesta en tensión del módulo de seguridad.

[0027] De manera convencional, el módulo de seguridad puede contener un número único de identificación UA1 que se puede memorizar en una zona memoria de lectura única. Este número UA1 es generalmente accesible por el usuario en forma de número de serie que puede ser impreso en el mismo módulo de seguridad o en una documentación anexa por ejemplo.

[0028] La zona de trabajo Z2 contiene el programa operativo y los datos. Esta zona se constituye de memoria no volátil, pero con posibilidad de escritura tal como la de EEPROM. La zona Z2 puede contener también cierta memoria volátil como la RAM. De hecho, generalmente esta zona no es homogénea y puede comprender varias memorias del tipo ROM, RAM, EEPROM, NVRAM o Flash.

[0029] El microprocesador CPU se dirige automáticamente sobre la primera zona Z1 durante una activación o un reinicio (reset). Es allí donde se ejecutan las primeras operaciones de seguridad. Estas operaciones utilizarán la primera zona memoria, y también la zona de trabajo Z2 en caso de necesidad.

[0030] En la figura 1, el bloque I/O ilustra los medios de comunicación hacia el exterior del módulo SM, medios indispensables para utilizar las funciones criptográficas y los derechos almacenados en la memoria. Es también por este medio que de los datos se extraen accidentalmente de la zona Z2 por un fallo de seguridad tal como descrito más arriba.

[0031] Como se indica anteriormente, la zona de trabajo Z2 contiene el programa operativo destinado al funcionamiento del módulo. Una forma de realización de la estructura del programa operativo se ilustra de manera detallada en las figuras 2 y 3. Este programa operativo se compone de códigos informáticos que se pueden representar en forma de líneas de instrucciones que poseen funciones determinadas cuando uno se coloca antes de la compilación de tal programa.

[0032] Para la claridad de la descripción, se supone que las instrucciones se reparten en bloques de instrucciones de referencia B1, B2, B3, que responden a una sintaxis determinada.

[0033] En el módulo de la invención, al menos dos tipos de líneas de instrucciones coexisten. El primer tipo corresponde a las instrucciones convencionales llamadas líneas reales, que se ejecutan por el microprocesador según criterios definidos y que producen un resultado "útil" para el funcionamiento del programa. El segundo tipo de instrucciones son instrucciones que no se ejecutan realmente por medio del microprocesador y/o que no producen directamente un resultado. Esta líneas de instrucciones, llamadas líneas artificiales a continuación en el texto, se utilizan en cambio para formar un medio de identificación único UA2 asociado al módulo de seguridad en cuestión. De hecho, las líneas artificiales pueden ser, sea instrucciones que no se ejecutan a través del microprocesador, sea instrucciones que se ejecutan realmente, pero que no producen ningún resultado que influencie el desarrollo del programa operativo. En otro términos, el funcionamiento del programa es el mismo, si tener en cuenta la presencia o no de estos códigos. Los términos "códigos artificiales" o "líneas artificiales" se deben entender como representativos de estos dos modos de realización.

[0034] En referencia más particularmente al modo de realización ilustrado por la figura 2, el programa operativo incluye cierto número de bloques de instrucciones reales B1, B2, que pueden formar unas rutinas de programa, así como un conjunto de códigos informáticos artificiales, formando un bloque de instrucción B3, que tiene el mismo aspecto que un bloque de instrucciones convencional, aunque es diferente para cada módulo de seguridad. Estos códigos informáticos son compatibles para una ejecución por el microprocesador y responden a la sintaxis de dicho microprocesador, de modo que no sea posible, mediante un simple análisis de los códigos, identificar los códigos reales que se ejecutarán y los códigos artificiales que no se ejecutarán o que no tendrán ningún efecto sobre el programa operativo. Las instrucciones contenidas en este conjunto artificial son líneas artificiales que generalmente no se ejecutan a través del microprocesador, o que su ejecución no influencia el funcionamiento del programa; éstas sólo se utilizan para formar el

medio de identificación único UA2 del módulo. Las instrucciones reales se forman por medio de líneas reales indicadas por R en las figuras 2 y 3 y las líneas artificiales se representan con las referencias F en estas figuras. Este bloque de instrucciones B3 se puede insertar de preferencia en el programa operativo para disimularse mejor. Los códigos informáticos artificiales usados para la formación del medio de identificación pueden incluir también valores de registros o de variables por ejemplo.

[0035] Según la variante de realización ilustrada en la figura 3, el módulo de seguridad incluye, a la inversa del ejemplo precedente en el que las líneas artificiales son agrupadas en conjunto en la memoria del programa operativo, un cierto número de líneas de instrucciones artificiales F, repartidas entre las instrucciones reales R. Esta líneas artificiales forman un conjunto de códigos informáticos único y diferente para cada módulo de seguridad.

[0036] Teniendo en cuenta que, generalmente las líneas de instrucciones se efectúan las unas después de las otras, es importante que esta líneas de instrucciones no se ejecuten o que su ejecución no afecte el buen desarrollo del programa operativo. También es importante que estos códigos informáticos específicos no sean o sean detectados difícilmente por una persona malintencionada.

[0037] Para conciliar estas restricciones, varios modos de realización son disponibles. En uno de los modos de realización, las líneas artificiales tienen una información específica que indica que la línea en cuestión es artificial, y que en consecuencia, ésta no debe ser ejecutada por el microprocesador.

[0038] Según otro modo de realización, ciertas instrucciones reales contienen indicaciones relativas al emplazamiento de la líneas artificiales. Tal indicación por ejemplo se puede obtener en forma de instrucción que indica que no se debe tratar una línea dispuesta en un emplazamiento memoria determinado.

[0039] Las instrucciones que consisten en no tratar las líneas artificiales se pueden disimular, por ejemplo mediante la indicación de que sólo se debe saltar la línea en cuestión si se cumple una condición. Se puede obtener así el hecho de que esta condición se cumpla siempre. También se puede añadir a una línea real, una indicación según la cual la línea siguiente es artificial.

[0040] Según otra forma de realización, no hay nada en los códigos informáticos que distinga una línea artificial de una línea real. El módulo de seguridad contiene una información memorizada que indica el emplazamiento de los códigos informáticos que el microprocesador no debe ejecutar.

[0041] Una variante tal y como se ha mencionado brevemente más arriba, puede consistir también en utilizar como línea artificial una instrucción que se ejecute realmente por medio del microprocesador, pero que no tenga efecto sobre la continuación de la ejecución del programa. Tal instrucción podría ser una indicación de que el programa debe pasar a la línea siguiente. Por supuesto este tipo de instrucciones "inútiles" puede ser difícil de localizar, por ejemplo mediante la escritura de la instrucción en forma de salto condicional, indicando que el paso a la línea siguiente se debe realizar sólo si se cumple una condición determinada, siempre que esta condición se cumpla. Otra forma consistiría en enviar el programa a una dirección predeterminada cualquiera si se cumple una condición, asegurándose de que esta condición no se cumpla nunca. Otra forma consistiría en modificar un emplazamiento memoria poco importante. Estas instrucciones "inútiles" se indican en el texto como "sin influencia sobre la ejecución del programa operativo por medio del microprocesador", debido al hecho de que estas instrucciones se pueden suprimir sin que el resultado de la ejecución del programa operativo sea afectado.

[0042] Un modo particularmente bien adaptado para que la detección de líneas artificiales sea difícil para una persona malintencionada es la ofuscación o la disimulación, método que consiste en hacer que la comprensión de un código informático descompilado sea particularmente compleja.

[0043] Según una variante de la invención, también es posible que sólo una parte de las líneas artificiales sirvan para la identificación del módulo de seguridad. Las líneas artificiales que no sirven para la identificación del módulo de seguridad están sólo presentes para complicar la comprensión del código informático y para impedir que un pirata detecte cuáles son las informaciones que debe publicar si quiere permitir la realización de un clon funcional, y cuáles son las informaciones que debe omitir si no quiere que se divulgue también el número único de identificación de su módulo de seguridad.

[0044] Tales líneas artificiales adicionales se pueden insertar tanto en el modo de realización en el que el módulo incluye un bloque artificial como en el modo en el que las instrucciones se diseminan en las instrucciones reales.

[0045] Se debe señalar que también se pueden combinar los dos modos de realización, a saber el que se ilustra en la figura 2 y el que se ilustra en la figura 3, es decir que se pueden introducir instrucciones artificiales en un bloque determinado, mientras que por otra parte se reparten otras instrucciones artificiales entre las instrucciones reales.

[0046] También se puede generar más de un medio de identificación o introducir informaciones que permiten generar varias veces el mismo medio de identificación único UA2, de modo que, aunque ciertas de las líneas artificiales se detectan y que por lo tanto no están publicadas, se puede a pesar de eso determinar el medio de identificación UA2.

[0047] La realización del módulo de seguridad según la invención incluye una fase de personalización en la que se introducen los datos específicos al módulo. La invención se asocia también a una etapa de detección de un módulo cuyos códigos informáticos han sido publicados. Esta etapa de detección consiste en extraer, a partir de las informaciones publicadas, los datos específicos al módulo de seguridad.

[0048] El método de personalización según la invención consiste esencialmente en generar un conjunto único de códigos informáticos, y a escribir después estos códigos en la memoria programa.

[0049] Este método de personalización depende en primer lugar del tipo de módulo de seguridad elegido y más particularmente del emplazamiento de los códigos informáticos artificiales. De hecho, cuando los códigos artificiales se disponen en la memoria programa en forma de bloque separado, los códigos artificiales pueden ser generados en forma de bloque, e introducidos después en el módulo.

[0050] Cuando los códigos artificiales se dispersan en el código informático real, los códigos reales que forman el programa operativo se memorizan de tal modo que incluyen emplazamientos libres. Se generan después códigos artificiales, que se insertan luego en estos emplazamientos libres.

[0051] En el modo de realización en el que los códigos artificiales son códigos ejecutados realmente por el microprocesador, dichos códigos no obstante no teniendo ningún efecto sobre el desarrollo del programa operativo, es posible utilizar un repertorio de códigos. Este repertorio contiene un conjunto de códigos informáticos predefinidos que no influyen sobre el desarrollo del programa operativo. Estos códigos pueden ser, como se ha indicado previamente, un salto condicional, la escritura de un valor en una zona memoria, la modificación de un valor o cualquier instrucción que no modifique el desarrollo del programa, sin tener en cuenta si la instrucción es ejecutada o no.

[0052] También se puede prever un método que genera automáticamente medios de identificación a partir de los códigos artificiales contenidos en el repertorio. De hecho, al conocer el número de líneas de instrucciones libres y eventualmente el tamaño de los bloques a insertar, se puede extraer a partir de las instrucciones de la biblioteca cierto número de códigos para rellenar las líneas vacías del programa operativo y de tal modo que cada módulo de seguridad utiliza un conjunto de instrucciones único. Esta unicidad se puede conseguir tanto por medio de los códigos informáticos utilizados como por el orden de utilización de estos códigos. Este método se representa esquemáticamente en la figura 4 en la que la referencia 10 ilustra el repertorio de los códigos artificiales F1, F2,... . La referencia 11 representa los códigos informáticos reales R1, R2,... que forman el programa operativo. Estos códigos incluyen emplazamientos memoria vacíos.

[0053] Durante la personalización de los módulos de seguridad, un cierto número de códigos informáticos se elige entre los códigos artificiales memorizados en el repertorio, de tal modo que dos módulos de seguridad no contienen los mismos códigos. Estos códigos se introducen en los emplazamientos memoria libres del programa operativo. En el ejemplo ilustrado por la figura 4, los códigos artificiales de los módulos de seguridad que presentan las referencias SM1, SM2 y SM3 son respectivamente los conjuntos (F1, F1, F3), (F3, F2, F4) y (F3, F3, F1).

[0054] El método de personalización puede comprender además una etapa destinada a hacer que la detección de códigos informáticos artificiales sea más compleja. En particular, cuando los códigos artificiales se juntan en un emplazamiento memoria determinado en forma de bloque, es juicioso evitar que una simple comparación de los códigos informáticos de dos módulos de seguridad cuya seguridad ha sido violada permita a una persona malintencionada localizar los códigos artificiales y por lo tanto evitar su publicación. Para resolver este problema, una etapa de ofuscación o de disimulación se adapta adecuadamente.

[0055] La etapa de detección de un módulo cuyos códigos informáticos se han publicado tal como mencionado anteriormente consiste en extraer, a partir de las informaciones publicadas, la identificación única del módulo de seguridad, por un lado para reencontrar eventualmente el titular del módulo de origen y por otra parte para que el módulo y los clones que éste ha permitido realizar se vuelvan inoperantes.

[0056] Esta etapa de detección consiste esencialmente en comparar los códigos informáticos publicados con los que se han introducido en los módulos de seguridad durante la fase de personalización. Para eso, se prevén distintos medios. En particular, se puede tener una comparación "línea por línea" de los códigos publicados y de los códigos generados. Otra forma de realizar esta comparación consiste en extraer los códigos artificiales de los códigos publicados, para después aplicar una operación a estos códigos artificiales. Una operación básica que se puede efectuar es la concatenación de los bits que forman los códigos artificiales. Otra operación puede consistir en determinar una firma (hash) del bloque de instrucciones. De hecho, se puede utilizar cualquier operación que permita obtener un valor único a partir de un bloque de instrucciones único. Esta misma operación se aplica a los códigos informáticos generados durante la fase de personalización, los valores únicos son comparados después.

[0057] Las instrucciones artificiales diseminadas se tratan como en el caso precedente, ilustrado por la figura 2, de manera a determinar el medio único de identificación UA2 del módulo de seguridad.

[0058] Cuando se ha determinado el medio de identificación de un módulo de seguridad cuya seguridad ha sido violada, entonces el módulo de seguridad de origen así como los módulos clonados a partir de este módulo de origen se pueden vuelven inoperantes.

[0059] Otras variantes evidentes de realización no descritas anteriormente en detalle también forman parte de la invención. En particular, se puede introducir códigos informáticos artificiales que permiten generar más de un medio de identificación por módulo de seguridad. A modo de ejemplo, un primer medio de identificación se puede constituir de un bloque de instrucciones separadas y otro medio de identificación de códigos diseminados.

[0060] También es posible introducir códigos artificiales redundantes, de manera a poder extraer el medio de identificación incluso cuando una parte de los códigos artificiales se elimina durante la publicación.

[0061] Es posible prever que un mismo medio de identificación UA2 sea utilizado para un grupo de módulos de seguridad y no sólo para un módulo seguridad único. Esto es interesante cuando el grupo de módulos pertenece a una misma persona o de manera más general a una misma entidad. Una combinación de los diferentes modos de realización susodichos es también previsible, es decir que, por ejemplo un módulo de seguridad puede contener un primer medio de identificación común a un grupo de módulos y un segundo medio de identificación único por módulo.

[0062] El medio de identificación UA2 puede además ser definido a partir de códigos informáticos que representan valores en unos registros.

[0063] Por regla general, no se prevé que el medio de identificación UA2 sustituya el número de identificación UA1 contenido convencionalmente en un módulo de seguridad. El primer número de identificación UA1 está presente en el módulo y podrá por ejemplo ser impreso sobre el módulo si éste tiene la forma de una tarjeta chip o de una clave por ejemplo.

[0064] Por oposición a esto, el medio de identificación UA2 será mantenido secreto, así como la existencia de un segundo número de identificación UA2.

REIVINDICACIONES

1. Módulo de seguridad comprendiendo un microprocesador, una memoria programa que contiene al menos un programa operativo y un medio de identificación único de dicho módulo, **caracterizado por el hecho de que** este medio de identificación (UA2) se constituye de un conjunto de códigos informáticos artificial, compatible con su ejecución por dicho microprocesador del módulo y almacenado en la memoria programa.

2. Módulo de seguridad según la reivindicación 1, **caracterizado por el hecho de que** dichos códigos informáticos se disponen en un bloque de instrucciones específicas.

3. Módulo de seguridad según la reivindicación 1, **caracterizado por el hecho de que** dichos códigos informáticos artificiales se reparten entre los códigos informáticos que forman el programa operativo.

4. Módulo de seguridad según la reivindicación 2 o 3, **caracterizado por el hecho de que** dichos códigos informáticos artificiales no se ejecutan por medio de dicho microprocesador.

5. Módulo de seguridad según la reivindicación 2 o 3, **caracterizado por el hecho de que** dichos códigos informáticos artificiales no modifican el desarrollo del programa operativo ejecutado por dicho microprocesador.

6. Módulo de seguridad según la reivindicación 1, **caracterizado por el hecho de que** dicho módulo incluye además un conjunto de códigos informáticos artificiales que no se utilizan, ni para el funcionamiento del módulo de seguridad, ni para formar el medio de identificación.

7. Método de personalización de un módulo de seguridad por un identificador único, este módulo comprendiendo un microprocesador y una memoria programa incluyendo al menos un programa operativo, **caracterizado por el hecho de que** ésta incluye las etapas siguientes:

   - generación de un conjunto único de códigos informáticos, llamados códigos informáticos artificiales, compatibles con su ejecución por dicho microprocesador;

   - escritura de este conjunto de códigos en la memoria programa en emplazamientos memoria específicos.

8. Método de personalización según la reivindicación 7, **caracterizado por el hecho de que** los códigos informáticos artificiales dispuestos en dichos emplazamientos memoria específicos no se ejecutan por medio de dicho microprocesador.

9. Método de personalización según la reivindicación 7, **caracterizado por el hecho de que** los códigos informáticos artificiales dispuestos en dichos emplazamientos memoria específicos no tienen influencia sobre la ejecución por dicho microprocesador del programa operativo.

10. Método de personalización según la reivindicación 8 o 9, **caracterizado por el hecho de que** dichos códigos informáticos artificiales que forman dicho conjunto único son elegidos a partir de una biblioteca de códigos informáticos.

11. Método de personalización según la reivindicación 7, **caracterizado por el hecho de que** dichos códigos informáticos artificiales forman un bloque de instrucción distinto de los códigos informáticos que constituyen el programa operativo.

12. Método de personalización según la reivindicación 7, **caracterizado por el hecho de que** dichos códigos informáticos artificiales están dispersos entre los códigos informáticos que constituyen el programa operativo.

13. Método de personalización según cualquiera de las reivindicaciones 7 a 12, **caracterizado por el hecho de que** de los códigos informáticos se tratan de manera a disimular la estructura del programa formado por estos códigos.

14. Método de identificación de un módulo de seguridad tal como definido en cualquiera de las reivindicaciones 1 a 6 y cuyos códigos informáticos se han vuelto accesibles al público, este método comprendiendo las etapas de:

   - extracción de los códigos informáticos artificiales de los códigos informáticos que se han vuelto accesibles al público;

   - tratamiento de dichos códigos informáticos artificiales según reglas predefinidas de manera a deducir a partir de éstos el medio de identificación de dicho módulo de seguridad.

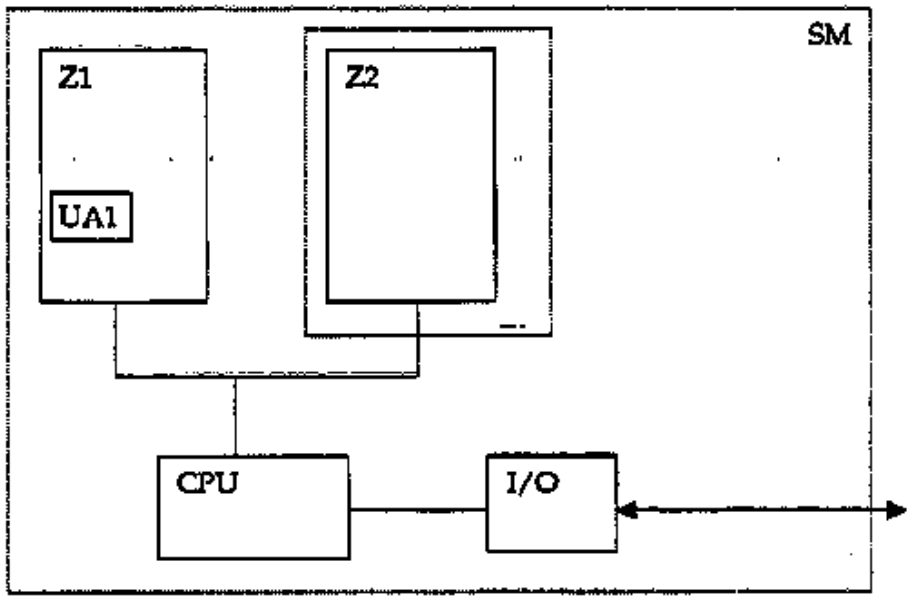

Fig. 1

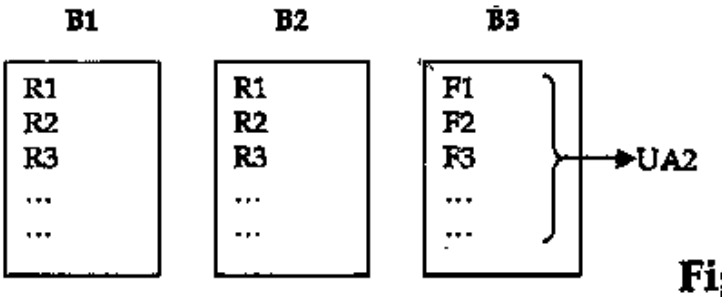

Fig. 2

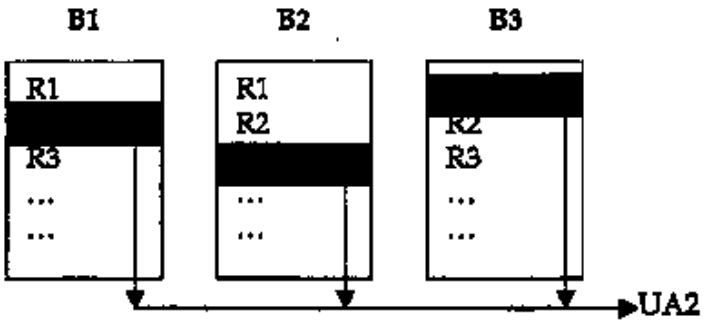

Fig. 3

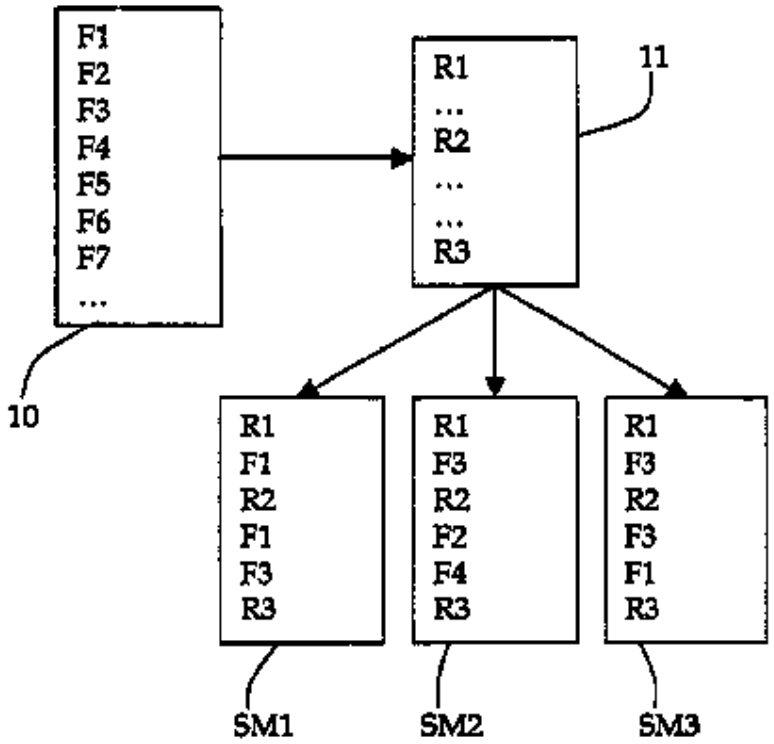
F1
F2
F3
F4
F5
F6
F7
...
10
R1
...
R2
...
...
R3
11
R1
F1
R2
F1
F3
R3
SM1
R1
F3
R2
F2
F4
R3
SM2
R1
F3
R2
F3
F1
R3
SM3

Fig. 4