



(19)



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 283 134**

(51) Int. Cl.:
H04L 9/06 (2006.01)

(12)

TRADUCCIÓN DE PATENTE EUROPEA

T3

- (86) Número de solicitud europea: **99950890 .6**
(86) Fecha de presentación : **29.10.1999**
(87) Número de publicación de la solicitud: **1125394**
(87) Fecha de publicación de la solicitud: **22.08.2001**

(54) Título: **Procedimiento de contramedida en un componente electrónico que aplica un algoritmo de criptografía de clave secreta.**

(30) Prioridad: **29.10.1998 FR 98 13605**

(45) Fecha de publicación de la mención BOPI:
16.10.2007

(45) Fecha de la publicación del folleto de la patente:
16.10.2007

(73) Titular/es: **GEMPLUS**
avenue du Pic de Bertagne
Parc d'Activites de Gémenos
13881 Gémenos Cédex, FR
Christophe Clavier y
Jean-Sébastien Coron

(72) Inventor/es: **Clavier, Christophe y**
Coron, Jean-Sébastien

(74) Agente: **Cañadell Isern, Roberto**

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

Procedimiento de contramedida en un componente electrónico que aplica un algoritmo de criptografía de clave secreta.

La presente invención se refiere a un procedimiento de contramedida en un componente electrónico que aplica un algoritmo de criptografía de clave secreta. Se utilizan en aplicaciones en las que el acceso a servicios o datos está controlado severamente. Su arquitectura está formada en torno a un microprocesador y memorias, entre las cuales una memoria programa que contiene la clave secreta.

Estos componentes se utilizan principalmente en las tarjetas inteligentes, para ciertas de sus aplicaciones. Por ejemplo, aplicaciones de acceso a ciertos bancos de datos, aplicaciones bancarias, aplicaciones de telepeaje, por ejemplo, para la televisión, la distribución de carburante e incluso el paso de peajes de autopistas. Estos componentes o tarjetas, aplican entonces un algoritmo de criptografía de clave secreta, entre los cuales el más conocido es el algoritmo DES (por Data Encryption Standard en la literatura anglosajona). Existen otros algoritmos de claves secreta, como el algoritmo RC5 e igualmente el algoritmo COMP128. Evidentemente, esta lista no es exhaustiva.

De manera general y sucinta, estos algoritmos tienen como función calcular un mensaje codificado a partir de un mensaje aplicado en entrada (en la tarjeta) por un sistema huésped (servidor, distribuidor bancario...) y de la clave secreta contenida en la tarjeta, y suministrar en respuesta al sistema huésped este mensaje codificado, lo que permite por ejemplo al sistema huésped autenticar el componente o la tarjeta, intercambiar datos...

Ahora bien, ha resultado que estos componentes o estas tarjetas son vulnerables a ciertos ataques que consisten en un análisis diferencial de consumo en corriente y que permiten a terceras personas mal intencionadas descubrir la clave secreta. Estos ataques se denominan ataques DPA, acrónimo anglosajón por Differential Power Analysis.

El principio de esos ataques DPA se basa en el hecho de que el consumo en corriente del microprocesador que ejecuta las instrucciones varía según el dato manipulado.

Principalmente, una instrucción del microprocesador que manipula un bit de datos genera dos perfiles de corriente diferentes, según que este bit valga "1" u "0".

Típicamente, si la instrucción manipula un "0", tenemos en este instante de ejecución una primera amplitud de la corriente consumida y si la instrucción manipula un "1", tenemos una segunda amplitud de la corriente consumida, diferente de la primera.

Las características de los algoritmos de criptografía ya se conocen: cálculos efectuados, parámetros utilizados. La única desconocida es la clave secreta contenida en la memoria programa. Ésta no puede deducirse si sólo se conoce el mensaje aplicado en entrada y del mensaje codificado suministrado en retorno.

Sin embargo, en un algoritmo de criptografía, ciertos datos calculados dependen solamente del mensaje aplicado en claro a la entrada de la tarjeta y de la clave secreta contenida en la tarjeta. También pueden calcularse de nuevo en el algoritmo otros datos solamente a partir del mensaje codificado (generalmente suministrado en claro a la salida de la tarjeta hacia el sistema huésped) y de la clave secreta contenida en la tarjeta. Más precisamente, cada bit de estos datos particulares puede determinarse a partir del mensaje de entrada o de salida, y de un número limitado de bits particulares de la clave.

Así, a cada bit de un dato dado particular, corresponde una subclave formada por un grupo particular de bits de la clave.

Los bits de estos datos particulares que pueden predecirse se denominan en adelante, bits punto blanco.

La idea de base del ataque DPA consiste de este modo en utilizar la diferencia del perfil de consumo en corriente de una instrucción según si manipula un "1" o un "0" y la posibilidad de calcular un blanco de bit por las instrucciones del algoritmo a partir de un mensaje conocido de entrada o de salida y de una hipótesis en la subclave correspondiente.

El principio del ataque DPA consiste en testar una hipótesis de subclave dada, aplicándolo en un gran número de curvas de medida en corriente, cada una de ellas relativa a un mensaje de entrada conocido por el agresor, una función booleana de selección, función de la hipótesis de subclave, y definida para cada curva por el valor predicho para un blanco de bit.

Al hacer una hipótesis en la subclave concernida, somos capaces en efecto de predecir el valor "0" o "1" que tomará ese blanco de bit para un mensaje de entrada o de salida dado.

Entonces se puede aplicar como función booleana de selección, el valor predicho "0" o "1" por el blanco de bit para la hipótesis de subclave considerada, para clasificar estas curvas en dos paquetes: un primer paquete agrupa las curvas que vieron la manipulación del blanco de bit en "0" y un segundo paquete agrupa las curvas que vieron la

ES 2 283 134 T3

manipulación del blanco de bit en “1” según la hipótesis de subclave. Al hacer la media de consumo en corriente en cada paquete, obtenemos una curva de consumo media $M0(t)$ para el primer paquete y una curva de consumo media $M1(t)$ para el segundo paquete.

- 5 Si la hipótesis de subclave es justa, el primer paquete agrupa realmente todas las curvas entre las N curvas que vieron la manipulación del blanco de bit en “0” y el segundo paquete agrupa realmente todas las curvas entre las N curvas que vieron la manipulación del blanco de bit en “1”. La curva media de consumo $M0(t)$ del primer paquete tendrá entonces un consumo medio por todas partes excepto en los momentos de la ejecución de las instrucciones críticas, con un perfil de consumo en corriente característico de la manipulación del blanco de bit en “0” (perfil₀). En
10 otros términos, para todas estas curvas todos los bits manipulados tuvieron tantas posibilidades de valer “0” como de valer “1”, excepto el blanco de bit que siempre tuvo el valor “0”. Lo que puede escribirse:

$$M0(t) = [(perfil_0 + perfil_1)/2]_{t \neq t_{ci}} + [perfil_0]_{t_{ci}}$$

- 15 ya sea

$$M0(t) = [Vm_t]_{t \neq t_{ci}} + [perfil_0]_{t_{ci}}$$

- 20 en donde t_{ci} representa los instantes críticos, en los que se ejecutó una instrucción crítica.

- Del mismo modo, la curva media de consumo $M1(t)$ del segundo paquete corresponde a un consumo medio por todas partes, excepto en los momentos de la ejecución de las instrucciones críticas, con un perfil de consumo en corriente característico de la manipulación del blanco de bit en “1” (perfil₁). Podemos escribir:

$$M1(t) = [(perfil_0 + perfil_1)/2]_{t \neq t_{ci}} + [perfil_1]_{t_{ci}}$$

- 30 ya sea

$$M1(t) = [Vm_t]_{t \neq t_{ci}} + [perfil_1]_{t_{ci}}$$

- 35 Hemos visto que los dos perfiles perfil₀ y perfil₁ no son iguales. La diferencia de las curvas $M0(t)$ y $M1(t)$ da entonces una señal DPA(t) cuya amplitud es igual al perfil₀-perfil₁ en los instantes críticos t_{ci} de ejecución de las instrucciones críticas que manipulan este bit, es decir, en el ejemplo representado en la figura 1, en los lugares t_{c0} a t_{c6} y cuya amplitud es más o menos igual a cero fuera de los instantes críticos.

- 40 Si la hipótesis de subclave es falsa, la clasificación no corresponde a la realidad. Estadísticamente, hay entonces en cada paquete tantas curvas que han visto verdaderamente la manipulación del blanco de bit en “0” como curvas que han visto la manipulación del blanco de bit en “1”. La curva media resultante $M0(t)$ se sitúa en este caso en torno a un valor medio dado por $(perfil_0 + perfil_1)/2 = Vm$, ya que para cada una de las curvas, todos los bits manipulados, incluidos el blanco de bit tienen tantas probabilidades de valer “0” como de valer “1”.

- 45 El mismo razonamiento sobre el segundo paquete conduce a una curva media de consumo en corriente $M1(t)$ cuya amplitud se sitúa en torno a un valor medio dado por $(perfil_0 + perfil_1)/2 = Vm$.

- La señal DPA(t) suministrada por la diferencia $M0(t) - M1(t)$ es en este caso prácticamente igual a cero. La señal
50 DPA(t) en el caso de una hipótesis de subclave falsa está representada en la figura 2.

- De este modo, el ataque DPA explota la diferencia del perfil de consumo en corriente durante la ejecución de una instrucción según el valor del bit manipulado, para efectuar una clasificación de curvas de consumo en corriente según una función de selección booleana para una hipótesis de subclave dada. Al efectuar un análisis diferencial del consumo
55 medio en corriente entre los dos paquetes de curvas obtenidos, se obtiene una señal de información DPA(t).

El desarrollo de un ataque DPA consiste entonces globalmente:

- 60 a - en extraer N mensajes aleatorios (p.ej. N igual a 1000);
b - en mandar ejecutar el algoritmo por la tarjeta para cada uno de los N mensajes aleatorios, tomando nota de la curva de consumo en corriente cada vez (medida en el Terminal de alimentación del componente);
c - en hacer una hipótesis en una subclave;
65 d - en predecir, para cada uno de los mensajes aleatorios, el valor que adquiere cada uno de los bits punto blanco cuyo valor sólo depende de los bits del mensaje (de entrada o de salida) y de la subclave tomada en hipótesis, para obtener la función de selección booleana;

e - en clasificar las curvas según esta función de selección booleana (es decir, según el valor “10” o “1” predicho para este blanco de bit para cada curva bajo la hipótesis de subclave;

f - en calcular en cada paquete la curva resultante de consumo media en corriente;

g - en efectuar la diferencia de estas curvas medias, para obtener la señal DPA (t).

Si la hipótesis en la subclave es justa, la función de selección booleana es justa y las curvas del primer paquete corresponden realmente a las curvas para las que el mensaje aplicado en entrada o en salida ha dado un blanco de bit de “0” en la tarjeta y las curvas del segundo paquete corresponden realmente a las curvas para las que el mensaje aplicado en entrada o en salida ha dado un blanco de bit de “1” en la tarjeta.

Nos encontramos en el caso de la figura 1: la señal DPA(t) ya no es nula en los instantes t_{c0} a t_{c6} que corresponden a la ejecución de las instrucciones críticas (aquellas que manipulan el blanco de bit).

Observamos que el agresor no requiere conocer con precisión los instantes críticos. Basta con que tenga por lo menos un instante crítico en el periodo de adquisición.

Si la hipótesis de subclave no es justa, la clasificación no corresponde con la realidad y entonces tenemos en cada paquete tantas curvas que corresponden en realidad a un blanco de bit de “0” como curvas que corresponden a un blanco de bit en “1”. La señal DPA(t) es prácticamente nula por todas partes (representado en fig. 2). Deberá volverse a la etapa c y hacer una nueva hipótesis sobre la subclave.

Si la hipótesis resulta ser justa, se puede pasar a la evaluación de otras subclaves, hasta haber reconstituido la clave al máximo. Por ejemplo, con un algoritmo DES, se utiliza una clave de 64 bits, de los cuales solamente 56 bits útiles. Con un ataque DPA, somos capaces de reconstituir al menos 48 bits de 56 bits útiles.

El artículo de Miyaguchi, “Secret key ciprés that change the encipherment algorithm Ander the control of the key”, publicado el 4 de julio de 1999, describe un algoritmo de clave secreta del tipo DES modificado para soportar un ataque diferencial.

La presente invención tiene como objetivo aplicar en un componente electrónico, un procedimiento de contramedida que produce una señal DPA(t) nula, incluso en el caso en que la hipótesis de subclave sea justa.

De este modo, nada permite distinguir el caso de la hipótesis de subclave justa de los casos de hipótesis de subclave falsas. Mediante esta contramedida el componente electrónico está protegido contra los ataques DPA.

Pero en la invención, nos hemos dado cuenta de que no bastaba con hacer de modo que la señal DPA(t) sea nula relativamente a un blanco de bit dado.

En efecto, si consideramos el valor que tienen varios blancos de bit de un mismo dato manipulado por las instrucciones críticas, deberemos clasificar las curvas no ya en dos paquetes, sino en varios paquetes. Ya no tenemos una función de selección binaria. Podemos demostrar que al agrupar seguidamente estos paquetes de una manera u otra, podemos obtener una señal DPA(t) no nula en el caso de una hipótesis de subclave justa, mientras que hubiera sido nulo si lo hubiéramos clasificado según una función de selección binaria en un sólo blanco de bit.

Tomemos por ejemplo dos blancos de bit de un mismo dato. Ambos blancos de bit pueden tomar los 2^2 valores siguientes: “00”, “01”, “10” y “11”.

Al aplicar la función de selección a las $N=1000$ curvas de consumo en corriente medidas, obtenemos cuatro paquetes de curvas. Si la clasificación es justa, un primer paquete de 250 curvas aproximadamente corresponde al valor “00”, un segundo paquete de 250 curvas aproximadamente corresponde al valor “01”, un tercer paquete de 250 curvas aproximadamente corresponde al valor “10” y un cuarto paquete de 250 curvas aproximadamente corresponde al valor “11”.

Si agrupamos el primero y el cuarto paquete en un primer grupo y el segundo y el tercer paquete en un segundo grupo, obtenemos dos grupos que no son equivalentes.

En el primer grupo, los dos bits tienen tantas probabilidades de valer “00” como de valer “11”. El valor medio en los instantes críticos de todas las curvas de consumo de este grupo puede escribirse:

$$M1(t_{ci}) = [\text{consumo}(\text{“00”}) + \text{consumo}(\text{“11”})]/2$$

En el segundo grupo, los dos bits tienen tantas probabilidades de valer “01” como de valer “10”. El valor medio en los instantes críticos de todas las curvas de consumo de este grupo puede escribirse:

ES 2 283 134 T3

$$M1(t_{ci}) = [\text{consumo}("01") + \text{consumo}("10")]/2$$

Si hacemos la diferencia entre esos dos medios, obtenemos una señal DPA (t) no nula. En otros términos, los dos grupos de los que comparamos los consumos medios no tienen un contenido equivalente.

En la invención, hemos buscado a impedir la obtención de cualquier tipo de señal significativa en el sentido del ataque DPA. Cualquiera que sea el número de blancos de bit tomados, cualquiera que sea la combinación de paquetes efectuados para hacer la comparación de los consumos medios, la señal DPA(t) siempre será nula. Por lo tanto, es preciso obtener paquetes equivalentes, cualquiera que sea el número de blancos de bit considerados.

Se ha hallado una solución a estos diferentes problemas técnicos en la utilización de un valor aleatorio en una operación de O EXCLUSIVO con uno y/u otro de los datos de entrada y de salida de medios utilizados en el algoritmo.

Con una utilización según la invención de dicho valor aleatorio, los datos manipulados por las instrucciones críticas resultan impredecibles a la vez que tienen un resultado justo a la salida del algoritmo.

Como caracterizado, la invención concierne así pues un procedimiento de contramedida en un componente electrónico que aplica un algoritmo criptográfico de clave secreta, la aplicación del algoritmo que comprende la utilización de primeros medios para suministrar un dato de salida a partir de un dato de entrada, el dato de salida y/o de los datos derivados son manipulados por instrucciones críticas. Según la invención, el procedimiento de contramedida prevé la utilización de otros medios, de modo que el dato de salida y los datos derivados resulten impredecibles, esos otros medios se obtienen de dichos primeros medios por una operación de O EXCLUSIVO con un valor aleatorio o un valor aleatorio derivado en uno y/u otro de los datos de entrada y de salida de los dichos primeros medios.

Otras características y ventajas de la invención se detallan en la descripción siguiente que se hace a título indicativo y en absoluto limitativo y en referencia a los dibujos en anexo, en los que:

- las figs. 1 y 2 ya descritas representan la señal DPA(t) que puede obtenerse en función de una hipótesis sobre una subclave de la clave secreta K, según una agresión DPA;

- las figs. 3 y 4 son organigramas detallados de los primeros tours (soluciones iniciales) del algoritmo DES;

- la fig. 5 es un esquema-bloque de la operación SBOX utilizada en el algoritmo DES;

- la fig. 6 demuestra un ejemplo de tabla de constantes elemental de una entrada y una salida utilizada en la operación SBOX;

- la fig. 7, representa un primer ejemplo de organigrama de ejecución del DES con un procedimiento de contramedida según la invención;

- la fig. 8 es un organigrama de los primeros tours del DES correspondiente;

- las figs. 9 y 10 representan respectivamente un organigrama de ejecución del DES y un organigrama detallado de los primeros tours, en un segundo modo de aplicación del procedimiento de contramedida según la invención;

- las figs. 11 y 12 corresponden a un tercer modo de aplicación del procedimiento de contramedida según la invención;

- la fig. 13 representa un organigrama de ejecución del DES en una variante del tercer modo de aplicación;

- la fig. 14 representa un esquema-bloque simplificado de una tarjeta inteligente que comprende un componente electrónico en el que se aplica el procedimiento de contramedida según la invención.

El algoritmo criptográfico de clave secreta DES (en adelante hablaremos sencillamente del DES o del algoritmo DES) comprende 16 tours de cálculo, anotados T1 a T16, como se representa en las figuras 3 y 4.

El DES comienza por una permutación inicial IP en el mensaje de entrada M (figura 3). El mensaje de entrada M es una palabra f de 64 bits. Tras la permutación, se obtiene una palabra e de 64 bits, que se corta en dos para formar los parámetros de entrada L0 y R0 del primer tour (T1). L0 es una palabra d de 32 bits que contiene los 32 bits de peso fuerte de la palabra e. R0 es una palabra h de 32 bits que contiene los 32 bits de peso bajo de la palabra e.

La clave secreta K, que es una palabra q de 64 bits sufre ella misma una permutación y una compresión para suministrar una palabra r de 56 bits.

El primer tour comprende una operación EXP PERM en el parámetro R0, consistente en una expansión y una permutación para suministrar a la salida una palabra l de 48 bits.

ES 2 283 134 T3

Esta palabra l está combinada a un parámetro K_1 , en una operación de tipo O EXCLUSIVO anotada XOR, para suministrar una palabra b de 48 bits. El parámetro K_1 que es una palabra m de 48 bits se obtiene de la palabra r por un desfase de una posición (operación anotada SHIFT en las figuras 3 y 4) seguido de una permutación y de una compresión (operación anotada COM PERM).

La palabra b se aplica a una operación anotada SBOX, a la salida de la cual se obtiene una palabra de 32 bits. Esta operación particular se explicará más detalladamente en relación con las figuras 5 y 6.

La palabra h ha sufrido una permutación P PERM, dando a la salida la palabra c de 32 bits.

Esta palabra c se combina al parámetro de entrada L_0 del primer tour T_1 , en una operación lógica del tipo O EXCLUSIVO, anotada XOR, que suministra a la salida la palabra g de 32 bits.

La palabra h (OR_0) del primer tour suministra el parámetro de entrada L_1 del tour siguiente (T_2) y la palabra g del primer tour suministra el parámetro de entrada R_1 del tour siguiente.

La palabra p del primer tour suministra la entrada r del tour siguiente.

Los demás tours T_2 a T_{16} se desarrollan de manera similar, excepto en lo que se refiere a la operación de desfase SHIFT que se hace en una o dos posiciones según los tours considerados.

Cada tour T_i recibe así pues a la entrada, los parámetros L_{i-1} , R_{i-1} y r y suministra a la salida los parámetros L_i y R_i y r para el tour siguiente T_{i+1} .

Al final del algoritmo DES (figura 4), el mensaje codificado se calcula a partir de los parámetros L_{16} y R_{16} suministrados por el último tour T_{16} .

Este cálculo del mensaje codificado C comprende en la práctica las siguientes operaciones:

- formación de una palabra e' de 64 bits invirtiendo la posición de las palabras L_{16} y R_{16} , y seguidamente concatenándolas;

- aplicación de la permutación IP^{-1} inversa a la del principio de DES, para obtener la palabra f' de 64 bits que forma el mensaje cifrado C .

La operación SBOX se detalla en las figuras 5 y 6. Comprende una tabla de constantes TC_0 para suministrar un dato de salida a en función de un dato de entrada b .

En la práctica, esta tabla de constantes TC_0 se presenta en forma de ocho tablas de constantes elementales TC_{01} a TC_{08} , cada una de ellas recibe en entrada solamente 6 bits de la palabra b , para suministrar a la salida solamente 4 bits de la palabra a .

De este modo, la tabla de constante elemental TC_{01} representada en la figura 6 recibe como dato de entrada, los bits b_1 a b_6 de la palabra b y suministra como dato de salida los bits a_1 a a_4 de la palabra a .

En la práctica estas ocho tablas de constantes elementales TC_{01} a TC_{08} se memorizan en memoria programa del componente electrónico.

En la operación SBOX del primer tour T_1 , un bit particular del dato a de salida de la tabla de constante TC_0 depende solamente de 6 bits del dato b aplicado en entrada, es decir de solamente 6 bits de la clave secreta K y del mensaje de entrada (M).

En la operación SBOX del último tour T_{16} , un bit particular del dato a de salida de la tabla de constante TC_0 puede calcularse de nuevo a partir de solamente 6 bits de la clave secreta k y del mensaje cifrado (C).

Ahora bien, si consideramos el principio del ataque DPA, si elegimos uno o varios bits del dato de salida a como blancos de bit, basta con hacer una hipótesis en bits de la clave K , para predecir el valor del o de los blancos de bit para un mensaje de entrada (M) o de salida (C) dado. En otros términos, APRA el DES, basta con hacer una hipótesis en una subclave de 6 bits.

En una agresión DPA en dicho algoritmo para un conjunto de blancos de bit dado resultante de una tabla de constantes elementales dada, debemos discriminar una hipótesis de subclave justo entre 64 posibles.

De este modo, a partir de los bits de salida de las ocho tablas de constantes elementales TC_{01} a TC_{08} , podemos descubrir hasta $8 \times 6 = 48$ bits de la clave secreta, haciendo ataques DPA en los blancos de bit correspondientes.

En el DES, encontramos instrucciones en el sentido de las agresiones DPA al principio del algoritmo y al final.

ES 2 283 134 T3

Al principio del algoritmo DES, los datos que pueden predecirse a partir de un mensaje de entrada M y de una hipótesis de subclave, son los datos a y g calculados en el primer tour (T_i).

5 El dato a del primer tour T_1 (figura 3) es el dato de salida de la operación SBOX del tour considerado. El dato g se calcula a partir del dato a, por permutación (P PERM) y operación O EXCLUSIVO con el parámetro de entrada L_0 .

De hecho, el dato c del primer tour, es un dato derivado del dato a del primer tour. El dato derivado c corresponde a una simple permutación de bits del dato a.

10 El dato l del segundo tour es un dato derivado del dato g del primer tour, ya que corresponde a una permutación de los bits de la palabra g, ciertos bits de la palabra g están por otra parte duplicados.

Si conocemos a y g, también podemos conocer esos datos derivados.

15 Las instrucciones críticas del principio del algoritmo son las instrucciones críticas que manipulan ya sea el dato que puede predecirse, como el dato a o el dato g del primer tour, ya sea un dato derivado.

20 Las instrucciones críticas que manipulan el dato a del primer tour T_1 o el dato derivado c son así las instrucciones de fin de la operación SBOX, de la operación P PERM y del principio de la operación XOR del primer tour T_1 .

Las instrucciones críticas que manipulan el dato g o los datos derivados son todas las instrucciones de final de operación XOR del final del primer tour T_i hasta las instrucciones del inicio de la operación SBOX del segundo tour T_2 , y las instrucciones del principio de la operación XOR al final del tercer tour $T_3(L_2=h(T_2)=g(T_1))$.

25 Al final del algoritmo DES, los datos que pueden predecirse a partir de un mensaje codificado C y una hipótesis de subclave, son el dato a del decimosexto tour T_{16} y el dato L_{15} igual a la palabra h del decimocuarto tour T_{14} .

30 Las instrucciones críticas que manipulan el dato a del decimosexto tour o de los datos derivados son las instrucciones del decimosexto tour de final de operación SBOX, de la operación de permutación P PERM y del inicio de operación XOR.

35 Para el dato L_{15} , las instrucciones críticas que manipulan este dato o los datos derivados son todas las instrucciones desde las instrucciones de final de operación XOR al final del decimocuarto tour T_{14} , hasta las instrucciones del principio de la operación SBOX del decimoquinto tour T_{15} , más las instrucciones del principio de la operación XOR al final del decimosexto tour T_{16} .

40 El procedimiento de contramedida según la invención aplicado a este algoritmo DES consiste en hacer impredecible cada uno de los datos manipulados por las instrucciones críticas. De este modo, cualquiera que sea el o los blancos de bit utilizados, la señal DPA(t) siempre será nula.

En lo que se refiere a la aplicación del procedimiento de contramedida según la invención del algoritmo DES, es necesario aplicar la contramedida a las instrucciones críticas del principio de DES y a las instrucciones críticas del final de DES, para estar totalmente protegido.

45 En el DES, todos los datos manipulados por instrucciones críticas son un dato de salida o datos derivados de un dato de salida de una operación SBOX.

50 En efecto, al principio de DES, los datos que pueden predecirse son los datos a y g del primer tour T_1 . El dato a es el dato de salida de la operación SBOX del primer tour. El dato g se calcula a partir del dato a, puesto que $g = P \text{ PERM}(a) \text{ XOR } L_0$. g es por tanto un dato derivado del dato de salida a de la operación SBOX del primer tour. De este modo, todos los datos manipulados por las instrucciones críticas del principio de DES derivan directa o indirectamente del dato de salida a de la operación SBOX del primer tour.

55 En lo que se refiere al final de DES, los datos que pueden predecirse son el dato a del decimosexto tour T_{16} y el dato g del decimocuarto tour T_{14} , g es igual a L_{15} .

El dato a es el dato de salida de la operación SBOX del decimosexto tour T_{16} .

60 En cuanto al dato L_{15} , éste se calcula en la ejecución normal del algoritmo DES, a partir del dato de salida a de la operación SBOX del decimocuarto tour T_{14} : $L_{15} = P \text{ PERM}(a) \text{ XOR } L_{14}$.

65 Si hacemos impredecibles los datos de salida a de esas operaciones SBOX particulares, hacemos impredecibles asimismo todos los datos derivados: por tanto hacemos impredecibles todos los datos manipulados por las instrucciones críticas del algoritmo SBOX que constituyen los primeros medios para suministrar un dato de salida $S=a$ a partir de un dato de entrada $E=b$, el procedimiento de contramedida aplicado al algoritmo DES consiste en utilizar otros medios para hacer impredecible el dato de salida, de modo que este dato de salida y/o de datos derivados manipulados por las instrucciones críticas siempre sean impredecibles.

ES 2 283 134 T3

Estos otros medios pueden comprender distintos medios. Son capaces a partir de los primeros medios aplicando un O exclusivo con un valor aleatorio o un valor aleatorio derivado en un y/o en otro de los datos de entrada y de salida de los primeros medios.

5 La utilización de este valor aleatorio es tal que el resultado a la salida, es decir, el mensaje codificado sigue siendo justo.

10 La figura 7 representa un primer modo de realización de la invención. En este modo, se reparten los dieciséis tours del algoritmo DES en cuatro grupos G1 a G4 de cuatro tours sucesivos. El grupo G1 comprende así pues los tours T1 a T4, el grupo G2, los tours T5 a T8, el grupo G3, los tours T9 a T12 y el grupo G4, los tours T13 a T16.

En una ejecución clásica del algoritmo DES, hemos visto que cada tour comprende el uso de primeros medios TC_0 en una operación SBOX.

15 En el primer modo de aplicación del procedimiento de contramedida, calculamos otros medios haciendo un O EXCLUSIVO con un valor aleatorio u y/o con un valor derivado $e(p(u))$ en uno y/u otro de los datos de entrada y de salida de los primeros medios TC_0 . Seguidamente, se aplica una secuencia SEQA de ejecución idéntica en cada grupo, que consiste en utilizar estos otros medios calculados.

20 Según la invención, se utiliza un valor aleatorio u que es un dato de 32 bits. Por ejemplo, se puede extraer un valor aleatorio de 32 bits, o bien extraer un valor aleatorio de 4 bits y copiarlos de nuevo 8 veces para obtener el valor aleatorio u en 32 bits.

25 Entonces calculamos la variable derivada igual a $e(p(u))$, en donde $p(u)$ corresponde al resultado de la operación P PERM aplicada en el valor u y en donde $e(p(u))$ es el resultado de la operación EXP PERM aplicada al valor $p(u)$.

Entonces podemos calcular los demás medios utilizados en la invención.

30 En el ejemplo representado en referencia a la figura 7, estos otros medios comprenden segundos medios TC_2 y terceros medios TC_1 .

Los segundos medios TC_2 se utilizan en el segundo tour y el penúltimo tour de cada grupo: es decir, en T2, T3 de G1, T6, T7 de G2, T10, T11 de G3 y T14 y T15 de G4.

35 Los segundos medios TC_2 se utilizan en el segundo tour y el penúltimo tour de cada grupo: es decir, en T2, T3 de G1, T6, T7 de G2, T10, T11 de G3 y T14 y T15 de G4.

40 Los segundos medios TC_2 se calculan aplicando un O EXCLUSIVO con la variable aleatoria derivada $e(p(u))$ en el dato de entrada E y aplicando un O EXCLUSIVO con el valor aleatorio u en el dato de salida S de los primeros medios TC_0 , lo que puede escribirse: $TC_2 = (E \oplus e(p(u)), S \oplus u)$.

Los terceros medios TC_1 se utilizan en el primer tour y el penúltimo tour de cada grupo: es decir, en T1, T4 de G1, T5, T8 de G2, T9, T12 de G3 y T13 y T16 de G4.

45 Los terceros medios TC_1 se calculan aplicando un O EXCLUSIVO con la variable aleatoria u en el dato de salida S de los primeros medios TC_0 lo que puede escribirse: $TC_1 = (E, S \oplus u)$.

50 El programa de cálculo consiste entonces al principio de la ejecución del algoritmo, en extraer un valor aleatorio u, en el ejemplo en 4 bits, en calcular la variable aleatoria derivada $e(p(u))$, y luego en calcular los distintos medios utilizados en la secuencia de ejecución SEQA. En el ejemplo, deben calcularse los segundos y terceros medios TC_2 y TC_1 .

55 Obtenemos, a la salida de cada grupo, el resultado justo para los parámetros de salida. De este modo, los parámetros de salida L4 y R4 del primer grupo G1, L8 y R8 del segundo grupo G2, L12 y R12 del tercer grupo G3, L16 y R16 del cuarto grupo G4 son justos cualquiera que sea la variable aleatoria extraída.

Cuando se han efectuado todos los tours, se obtienen los parámetros justos L16 y R16 que van a permitir calcular el mensaje cifrado C justo.

60 En cambio, al interior de los grupos, ciertos resultados intermedios no tienen los mismos valores según la secuencia utilizada, sino valores que corresponden a la operación O EXCLUSIVO con el valor aleatorio u o con el valor aleatorio derivado $e(p(u))$, como vamos a demostrarlo por referencia a las figuras 3 y 8.

65 La figura 8, muestra el organigrama detallado de los cuatro tours T1, T2, T3 y T4 del primer grupo G1, en la secuencia SEQA de ejecución según la invención.

En esta secuencia, el tour T1 utiliza los terceros medios TC_1 . A la salida de la operación SBOX, obtenemos así pues el dato modificado aleatoriamente $a \oplus u$ (Figura 8), en vez del dato a según la secuencia normal del DES, es decir sin contramedida (Figura 3).

ES 2 283 134 T3

Con la secuencia SEQA de ejecución según la invención, la operación P PERM del primer tour T1 que es una simple permutación suministrará igualmente a la salida un dato modificado aleatoriamente igual a $c \oplus p(u)$.

El dato que se obtiene por la operación XOR entre un dato $c \oplus p(u)$ y el dato L0, también suministrará a la salida un dato modificado aleatoriamente $g \oplus p(u)$. Este dato aplicado a la operación EXP PERM suministrará a la salida el dato modificado aleatoriamente anotado $l \oplus e(p(u))$.

Así, con los terceros medios TC_1 del tour T1 obtenemos todos los siguientes datos modificados aleatoriamente:

- en el tour T1: $a \oplus u$, $c \oplus p(u)$, $g \oplus p(u)$,
- en el tour T2: $R1 \oplus p(u)$, $h \oplus p(u)$; $l \oplus e(p(u))$, $b \oplus p(u)$;
- en el tour T3: $L2 \oplus p(u)$.

Entonces llegamos a los segundos medios TC_2 utilizados en el tour T2. Según su definición: $E \oplus e(p(u))$, $S \oplus u$, al aplicar a la entrada el dato modificado aleatoriamente $b \oplus e(p(u))$, obtenemos a la salida el dato modificado aleatoriamente $a \oplus u$. Basándonos en este razonamiento hasta el final del tour T4, y observando que $p(u) \oplus p(u) = 0$, obtenemos a la salida del tour T4, los datos L4, R4 no modificados.

Además, constatamos que para todas las instrucciones críticas del principio de DES, las instrucciones críticas manipularán datos modificados de manera aleatoria.

Con este tipo de procedimiento de contramedida, debemos prever al principio de DES la extracción del valor aleatorio u y el cálculo de los medios utilizados en la secuencia de ejecución SEQA. Estos medios calculados en cada ejecución del DES, se memorizan, el tiempo de la ejecución, en memoria de trabajo, los primeros medios TC_0 que sirven para el cálculo se memorizan en memoria programa.

Volviendo a la figura 7, podremos anotar, que no se necesita ninguna contramedida en los grupos del centro G2 y G3, puesto que no contienen instrucciones críticas en el sentido agresión DPA. Entonces, podremos limitarnos a aplicar solamente la secuencia de contramedida al primer y último grupo G1 y G4. Para ello, bastaría con utilizar seguidamente los primeros medios (TC_0) en los grupos G2 y G3.

Pero el hecho de aplicar el procedimiento de contramedida a todos los grupos proporciona una coherencia al conjunto.

De este modo, se aplica la secuencia SEQA a cada uno de los grupos G1 a G4.

Un segundo modo de realización del procedimiento de contramedida está representado en la figura 9. Este segundo modo de realización es de hecho una variante del primero.

El interés de esta variante radica en utilizar únicamente en la secuencia SEQA los segundos medios TC_2 . En efecto, hemos visto que los distintos medios TC_0 , TC_1 , TC_2 corresponden en la práctica a tablas de constantes que comprenden cada una de ellas ocho tablas de constantes elementales, que es preciso calcular de nuevo en lo que se refiere a los medios TC_1 y TC_2 en cada nueva ejecución del DES, y conservar en memoria de trabajo.

Esta variante consiste en utilizar únicamente los segundos medios TC_2 en la secuencia SEQA. Para ello, prevemos en el programa de cálculo de los primeros y últimos tours de cada grupo, una operación O EXCLUSIVO suplementaria CP con la variable aleatoria derivada $e(p(u))$, APRA obtener a la entrada de los segundos medios el dato $b \oplus e(p(u))$. Anotamos esta operación CP($e(p(u))$) en las figuras. Si consultamos la figura 10 que representa el organigrama detallado de la secuencia SEQA de ejecución de los cuatro tours T1 a T4 del primer grupo G1, se trata así pues de aplicar a la entrada de la operación SBOX de los tours T1 y T4 una variable $b \oplus e(p(u))$. La operación suplementaria CP más los segundos medios TC_2 equivalente a los terceros medios TC_1 usados en el primer modo de realización de la invención.

Con ello ganamos en tiempo de cálculo, puesto que la operación CP sólo se ejecuta dos veces en un grupo, ya sea 8 veces para una secuencia SEQA completa en los cuatro grupos, mientras que el cálculo de una tabla necesita hacer esta operación $b \oplus e(p(u))$ en todos los datos de entrada de dicha tabla.

Observaremos que la operación O EXCLUSIVO suplementaria CP con la variable $e(p(u))$ puede colocarse en diversos lugares de los primeros y últimos tours, ya sea entre la operación EXP PERM y la operación XOR o entre la operación XOR y la operación SBOX.

Podemos observar igualmente que se puede utilizar una operación O EXCLUSIVO suplementaria CP con la variable aleatoria derivada $p(u)$, colocando esta operación suplementaria CP ($p(u)$) antes de la operación EXP PERM. Obtenemos a la salida $l \oplus e(p(u))$, y por consiguiente tendremos a continuación $b \oplus e(p(u))$.

En todos estos ejemplos, obtenemos el dato $b \oplus e(p(u))$ a la entrada de la operación SBOX.

ES 2 283 134 T3

La figura 11 representa un tercer ejemplo de realización de un procedimiento de contramedida según el invento.

En este modo de realización, formamos un primer grupo G1 con los tres primeros tours T1, T2, T3 y otro grupo G4 con los tres últimos tours T14, T15, T16. Aplicamos en cada grupo la secuencia de ejecución SEQA con los otros
5 medios para ciertos tours como mínimo.

Para los demás tours no incluidos en los grupos, es decir para los tours T4 a T13, se aplican los primeros medios TC₀.

10 A la salida de cada grupo G1, G4, obtenemos el buen resultado en la salida L3, R3 y L16, R16, cualquiera que sea la variable aleatoria o extraída.

Los demás medios son en el ejemplo los terceros medios TC₁ ya vistos en relación con el primer modo de realización y los cuartos medios TC₃.

15 Estos cuartos medios se calculan con relación a los primeros medios TC₀ aplicando un O EXCLUSIVO en el dato E de entrada, con la variable aleatoria derivada e(p(u)).

De este modo, después de haber extraído el valor aleatorio u, y calculado la variable aleatoria derivada, calculamos
20 los distintos medios utilizados en la secuencia de ejecución SEQA. Después se aplica esta secuencia de ejecución SEQA en el primer grupo. Obtenemos a la salida los parámetros L3, R3. Ejecutamos los siguientes tours T4 a T13 con los primeros medios TC₀. Al final del tour T13, aplicamos la secuencia SEQA en el grupo G4. Obtenemos los parámetros L16, R16 que servirán para calcular el mensaje codificado C.

25 La figura 12 es un organigrama detallado correspondiente.

Aparece claramente en este organigrama que obtenemos datos modificados aleatoriamente para todas las instrucciones críticas de dichos tours. Los datos L3 y R3 a la salida del tercer tour no se modifican, lo que permite continuar la ejecución del algoritmo, pasando al tour T4 al cual se aplica los primeros medios TC₀ según la ejecución normal
30 del algoritmo.

En esta figura, podemos observar que en la operación SBOX del tercer tour T3, podríamos utilizar los primeros medios TC₀ en vez de los terceros medios calculados TC₁, previendo una operación O EXCLUSIVO suplementaria CP a la salida de la operación SBOX, para hacer un O EXCLUSIVO de la salida con la variable aleatoria u, para
35 obtener el dato a⊕u a la entrada de la operación XOR. Esto es una solución equivalente.

La figura 13 representa un organigrama de ejecución que utiliza esta variante. Para el tercer tour de los dos grupos G1 y G4, se utilizan en la secuencia de ejecución SEQA, los primeros medios TC₀ seguidos a la salida de la instrucción O EXCLUSIVO suplementaria con la variable u, lo que se anota T3 (TC₀, CP(u)).

40 De manera general, en el procedimiento de contramedida según el invento, podemos prever en la secuencia de ejecución SEQA y para uno o varios tours, una instrucción O EXCLUSIVO suplementaria CP a la entrada o la salida de los medios utilizados con la variable u o una variable aleatoria derivada p(u) o e(p(u)) según los casos.

45 La presente invención se aplica al algoritmo de criptografía de clave secreta DES, para el cual se han descrito varios ejemplos de aplicación no limitativos.

Se aplica más generalmente a un algoritmo de criptografía de clave secreta de dieciséis tours de cálculos, cuyas instrucciones críticas se sitúan entre las instrucciones de los tres primeros tours o los tres últimos tours.

50 Un componente electrónico 1 que aplica un procedimiento de contramedida según el invento en un algoritmo de criptografía de clave secreta DES, comprende típicamente, como se representa en la figura 10, un microprocesador μP, una memoria programa 2 y una memoria de trabajo 3. Para poder administrar la utilización de los distintos medios TC₀, TC₁, TC₂ según la invención, que son, en la práctica, tablas de constantes memorizadas en memoria programa, medios 4 de generación de un valor aleatoria entre 0 y 1, están previstos que, si se nos referimos a los organigramas
55 de las figuras 7 y 11, suministrarán el valor aleatoria u en cada ejecución del DES. Dicho componente puede utilizarse particularmente en una tarjeta inteligente 5, para mejorar su inviolabilidad.

60

65

REIVINDICACIONES

1. Procedimiento de contramedida contra las agresiones por análisis diferenciales de consumo en corriente en un componente electrónico que aplica un algoritmo criptográfico de llave secreta (K), la aplicación del algoritmo que comprende la utilización de los primeros medios de tratamiento digital (TC0) para suministra un dato de salida (S) a partir de un dato de entrada E, la utilización de medios de cálculo para producir datos derivados del dato de Salida, y la utilización de instrucciones críticas en el sentido de dichos ataques para manipular el dato de salida y/o los datos derivados del dato de salida,

caracterizado porque el procedimiento de contramedida comprende el reemplazo de los primeros medios por otros medios (TC1) de tratamiento digital, definidos a partir de dichos primeros medios, a partir de un valor aleatorio (u) y/o a partir de un valor aleatorio derivado $e(p(u))$ del valor aleatorio para, a partir ya sea del dato de entrada (E) de los primeros medios, ya sea del dato de entrada (E) de los primeros medios combinados por un O EXCLUSIVO al valor aleatorio derivado ($E/e(p(u))$), suministrar ya sea el dato de salida (S) de los primeros medios, ya sea el dato de salida de los primeros medios combinados por un O EXCLUSIVO al valor aleatorio (S/u), de modo que el dato producido por los otros medios y los datos derivados del dato producido por los otros medios sean impredecibles.

2. Procedimiento según la reivindicación 1, en el que los primeros medios de tratamiento digital son una operación de substitución asociando a cada dato de entrada un dato de salida, en el cual los demás medios son igualmente operaciones de substitución que asocian al dato de entrada (E) primeros medios o al dato de entrada (E) de los primeros medios combinados por un O EXCLUSIVO al valor aleatorio derivado ($E/e(p(u))$), ya sea el dato de salida (S) de los primeros medios, ya sea el dato de salida de los primeros medios combinados por un O EXCLUSIVO al valor aleatorio (S/u), y en el que los medios de cálculo que producen datos derivados del dato de salida de los primeros medios son operaciones de permutación y/o operaciones de combinación de tipo O EXCLUSIVO.

3. Procedimiento de contramedida según la reivindicación 1 o 2, en el que la aplicación del algoritmo que comprende dieciséis tours de cálculo (T1, ..., T16), cada tour utiliza primeros medios (TC0) para suministrar un dato de salida a partir de un dato de entrada, el dato de salida y/o los datos derivados son manipulados por instrucciones críticas en los tres primeros (T1, T2, T3) y los tres últimos tours (T14, T15, T16).

4. Procedimiento según la reivindicación 3, en el transcurso de la cual se forma un primer grupo (G1) que comprende los tres primeros tours como mínimo y un último grupo (G4) que comprende los tres últimos tours como mínimo, y porque se asocia al primer grupo (G1) y al último grupo (G4) una secuencia de ejecución (SEQA) que utiliza los demás medios (TC1, TC2) en ciertos tours como mínimo.

5. Procedimiento de contramedida según la reivindicación 4, **caracterizado** porque se forman cuatro grupos (G1,... G4) de cuatro tours sucesivos cada uno (T1,... T4) y, porque se aplica como mínimo al primer grupo (G1) y al último grupo (G4) dicha secuencia de ejecución (SEQA).

6. Procedimiento de contramedida según la reivindicación 5, **caracterizado** porque dicha secuencia (SEQA) se ejecuta en cada uno de los grupos (G1,... G4).

7. Procedimiento según una de las reivindicaciones 5 a 6, en el que los demás medios comprenden:

- segundos medios (TC2) que son operaciones de substitución que asocian al dato de entrada (E) combinado por un O EXCLUSIVO al valor aleatorio derivado ($E/e(p(u))$), el dato de salida combinado por un O EXCLUSIVO al valor aleatorio (S/u), y

- terceros medios (TC1) que son operaciones de substitución que asocian al dato de entrada (E), el dato de salida combinado por un O EXCLUSIVO al valor aleatorio (S/u), y en el que la secuencia de ejecución comprende:

- en el primer y en el último tour de un grupo, el reemplazo de los primeros medios por los terceros medios, y

- en el segundo y en el penúltimo tour de un grupo, el reemplazo de los primeros medios por los segundos medios.

8. Procedimiento según una de las reivindicaciones 5 a 6, en el que los demás medios comprenden terceros medios que son operaciones de substitución que asocian al dato de entrada (E), el dato de salida combinado por un O EXCLUSIVO al valor aleatorio (S/u),

y en el que la secuencia de substitución comprende:

- en todos los tours de un grupo, el reemplazo de los primeros medios (TC1) por los segundos medios (TC2),

- en el primer y el último tour de un grupo, la combinación del dato de entrada con el valor aleatorio derivado, el resultado de la combinación se suministra seguidamente como dato de entrada a los segundos medios.

ES 2 283 134 T3

9. Procedimiento de contramedida según cualquiera de las reivindicaciones anteriores, **caracterizado** en que cada ejecución del algoritmo comprende la extracción de un valor aleatorio (u) y el cálculo de los demás medios.

10. Procedimiento de contramedida según cualquiera de las reivindicaciones anteriores, **caracterizado** en que los distintos medios son tablas de constantes.

11. Procedimiento de contramedida según cualquiera de las reivindicaciones anteriores, **caracterizado** en que los distintos medios se utilizan en combinación con una operación O exclusivo suplementaria (CP) con el valor aleatorio o un valor derivado (p(u)), e(p(u)).

12. Componente electrónico de seguridad que comprende un microprocesador, una memoria programa y una memoria de trabajo que permite la aplicación de un algoritmo criptográfico de clave secreta (K), la memoria programa memoriza los primeros medios (TC0) de tratamiento digital para suministrar un dato de salida (S) a partir de un dato de entrada (E), el componente comprende igualmente medios para aplicar operaciones que producen datos derivados del dato de salida, y medios para aplicar instrucciones críticas del dicho algoritmo en el sentido de las agresiones por análisis diferencial de consumo en corriente, dichas instrucciones críticas manipulan el dato de salida y/o los datos derivados del dato de salida,

el componente se **caracteriza** porque comprende medios de aplicación de un procedimiento de contramedida contra dichos ataques según cualquiera de las reivindicaciones 1 a 11 anteriores, que comprenden,

- medios (4) de generación de un valor aleatorio (u),

- otros medios (TC1, TC2) de tratamiento digital memorizado en memoria de trabajo (3), calculados en cada nueva ejecución del algoritmo a partir de dichos primeros medios, a partir de un valor aleatorio (u) y/ o a partir de un valor aleatorio derivado e(p(u)), del valor aleatorio, los demás medios suministran ya sea el dato de salida (S) de los primeros medios, ya sea el dato de salida de los primeros medios combinados por un O EXCLUSIVO al valor aleatorio (S/u), a partir, ya sea del dato de entrada (E) de los primeros medios, ya sea al dato de entrada (E) de los primeros medios combinados por un O EXCLUSIVO al valor aleatorio derivado (E/e(p(u))).

13. Tarjeta inteligente que comprende un componente electrónico de seguridad según la reivindicación 12.

FIG.1

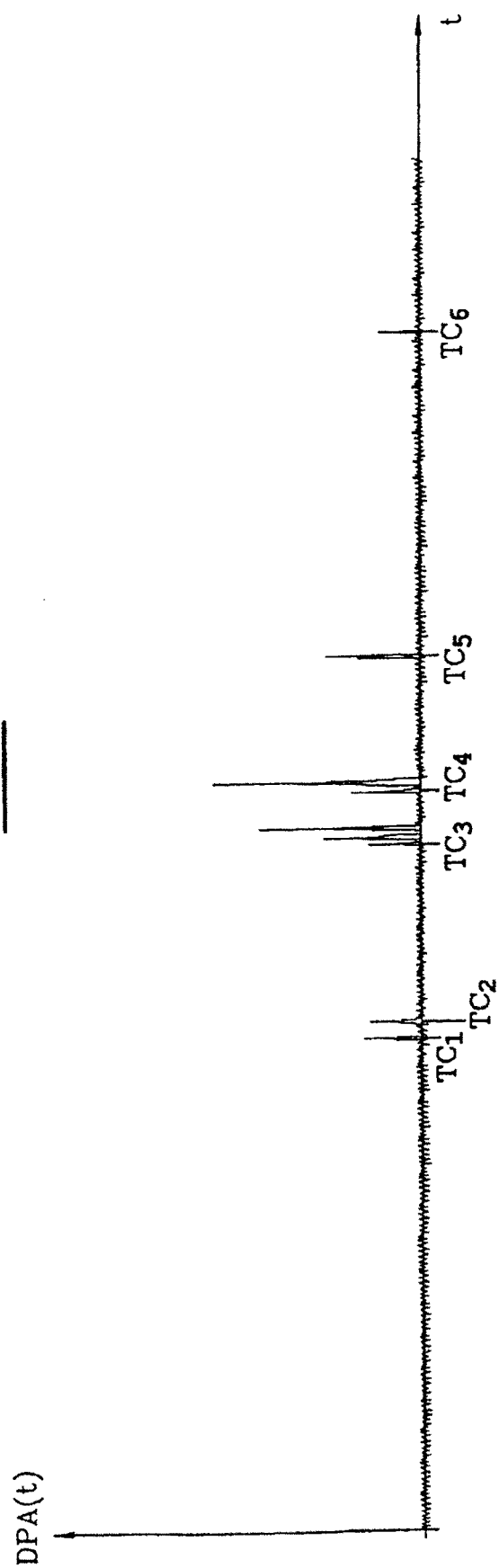
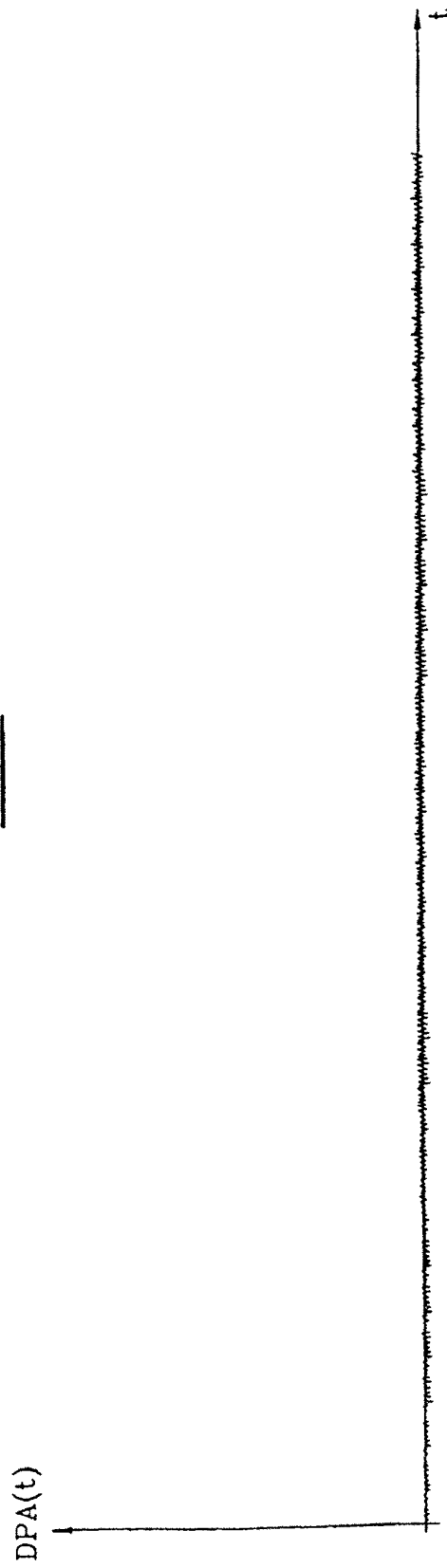
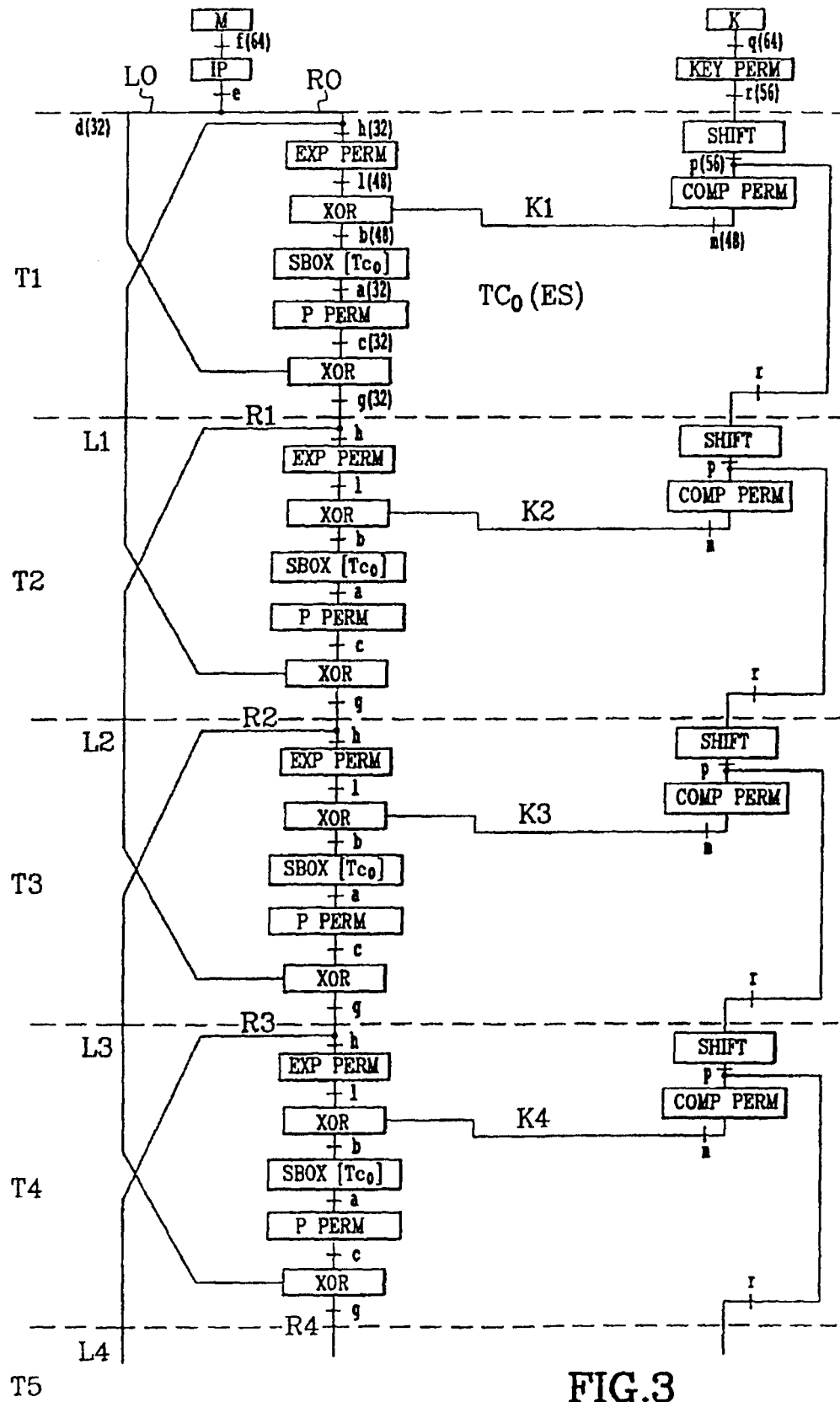


FIG.2





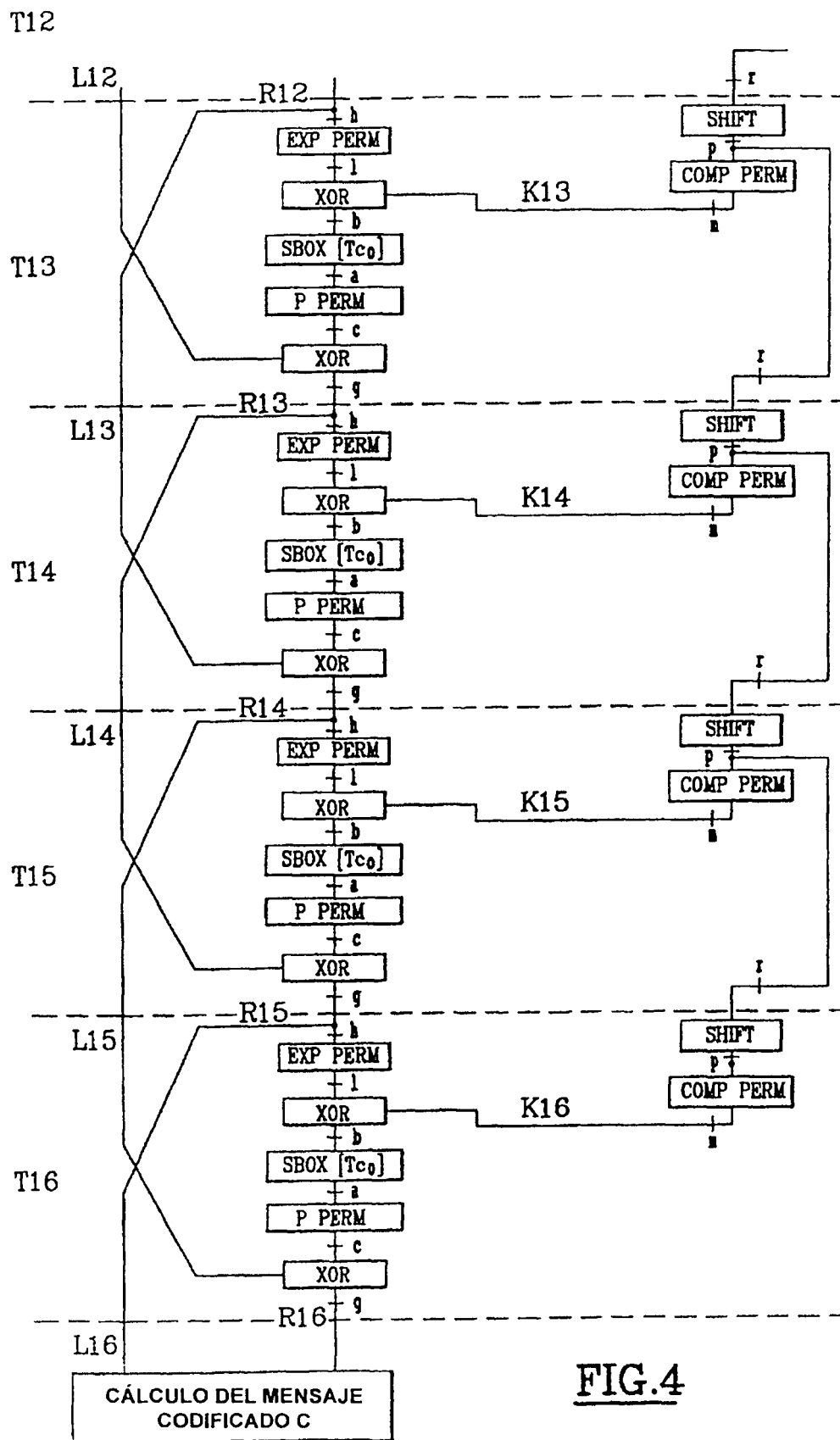


FIG.4

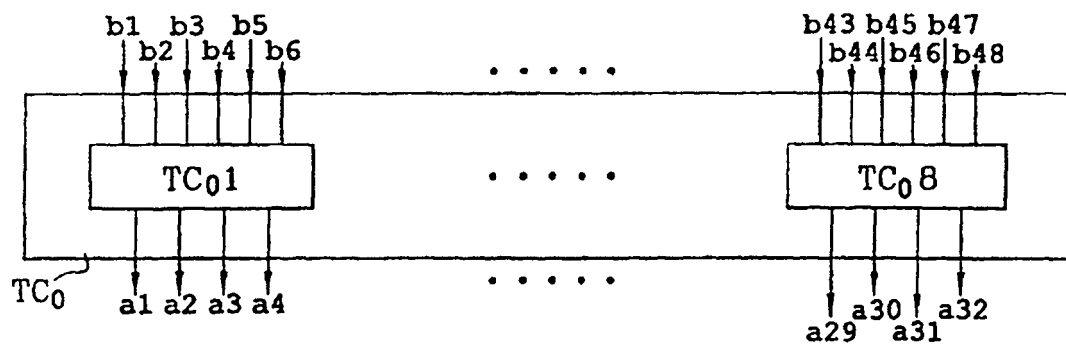


FIG.5

FIG. 6

TC ₀₁	E=b ₁ b ₂ b ₃ b ₄ b ₅ b ₆	S=a ₁ a ₂ a ₃ a ₄
	000000	1101
	000001	0101
	⋮	⋮
	⋮	⋮
	111111	1010

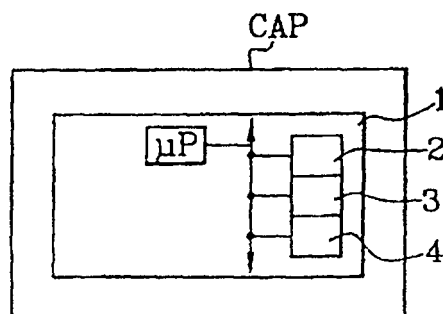


FIG.14

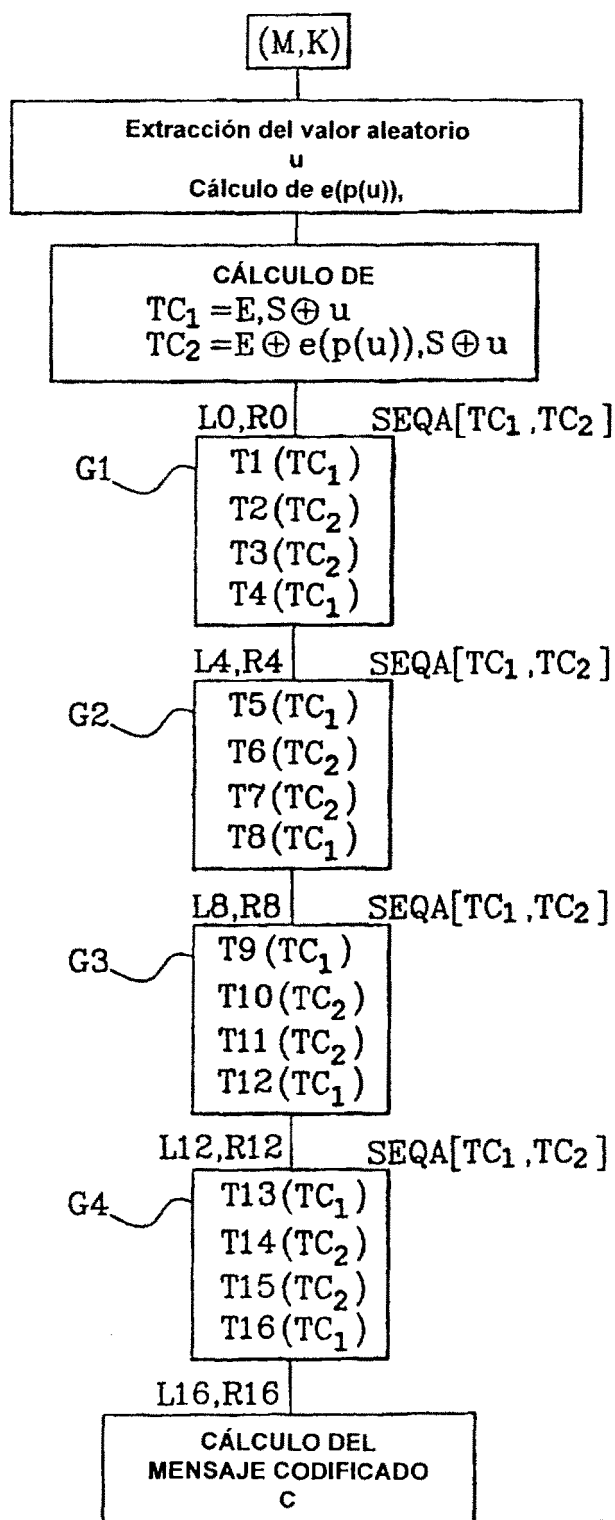


FIG.7

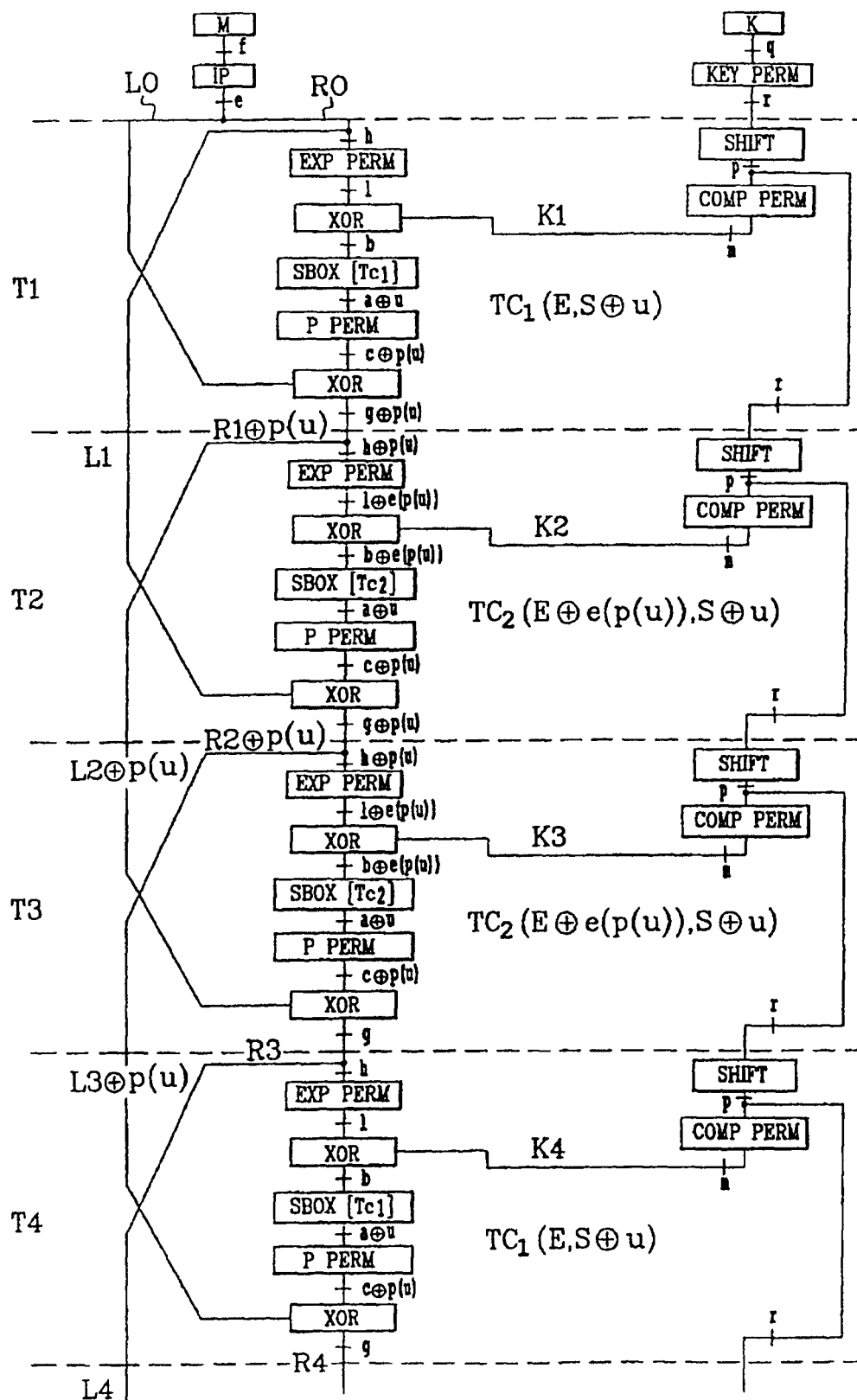


FIG. 8

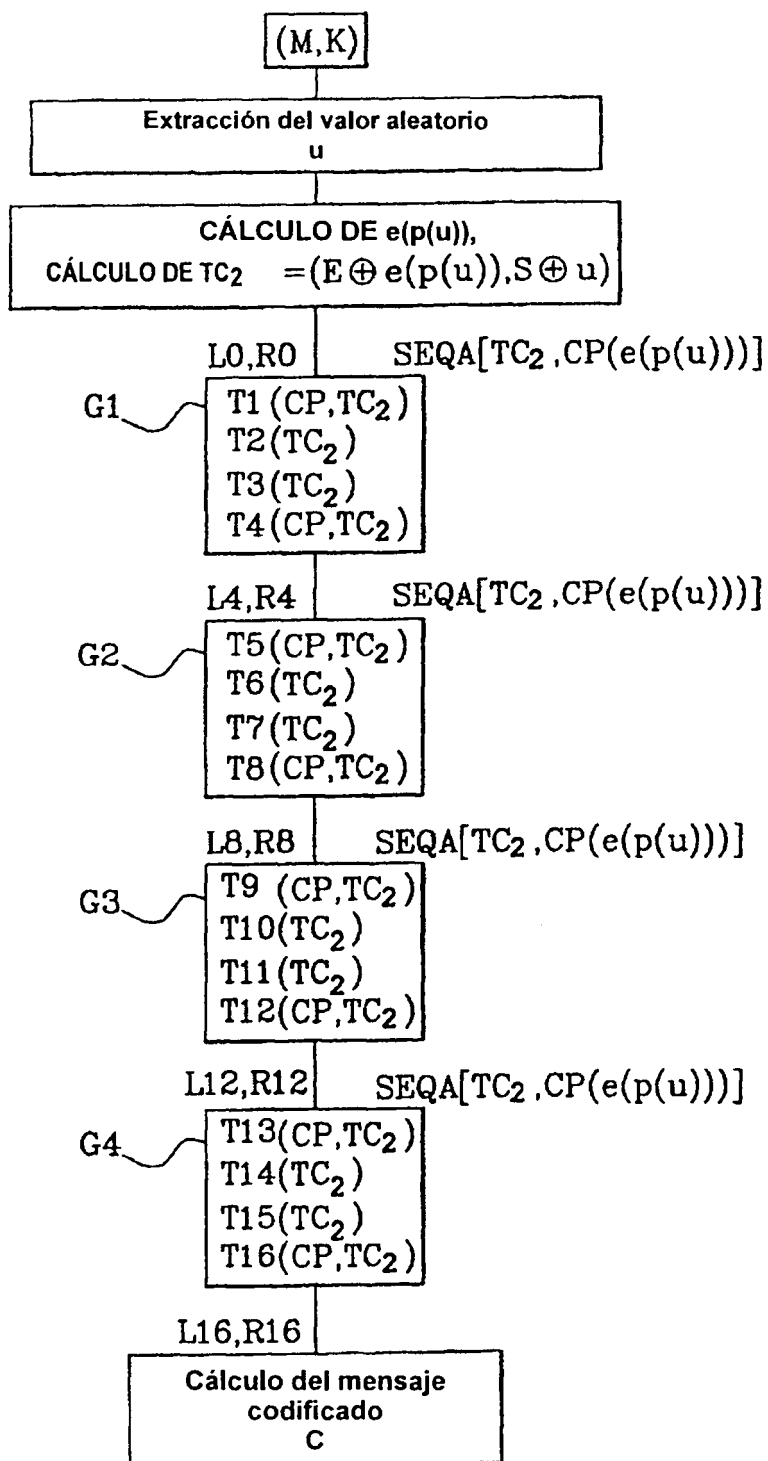
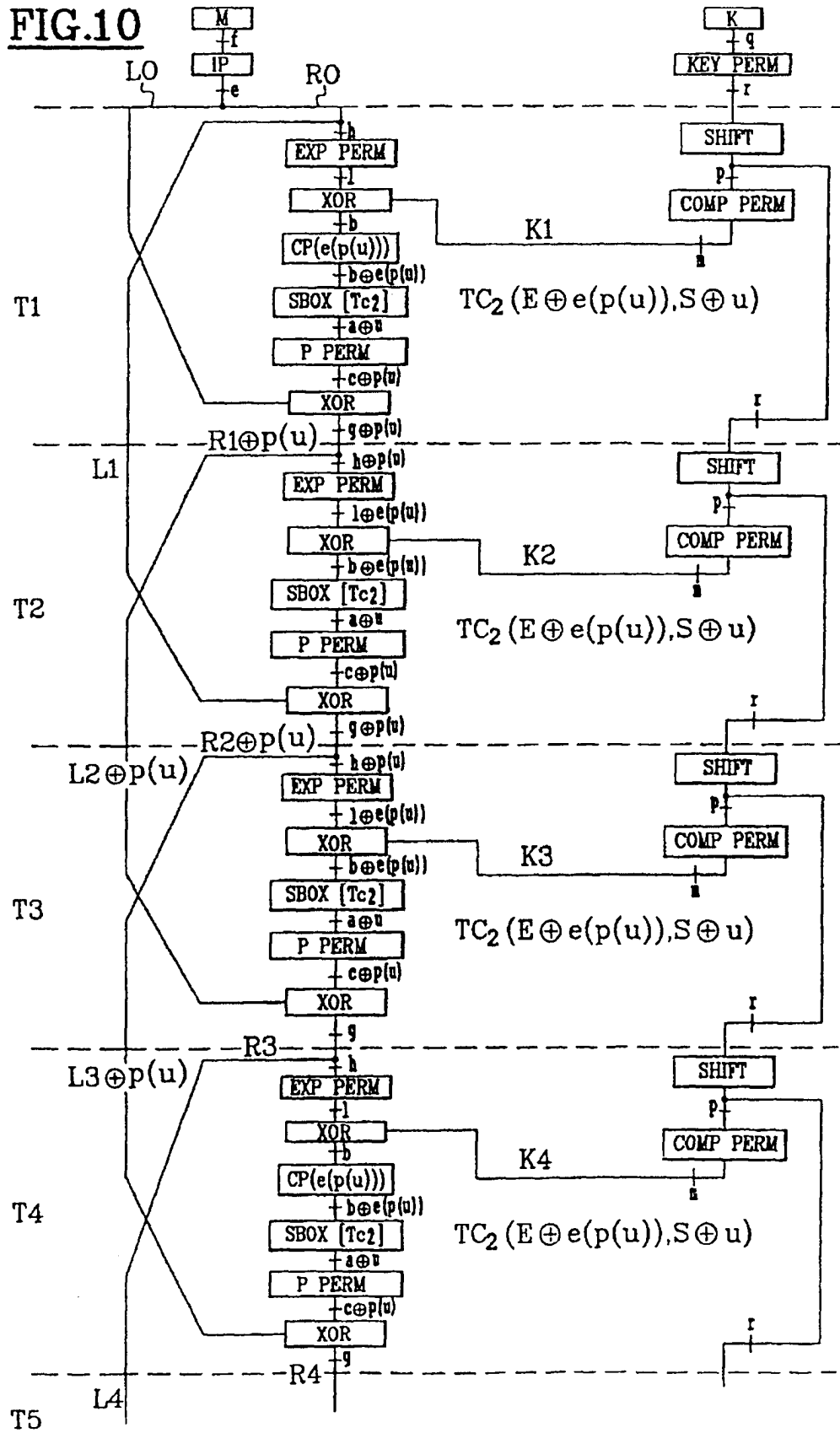


FIG.9

FIG.10



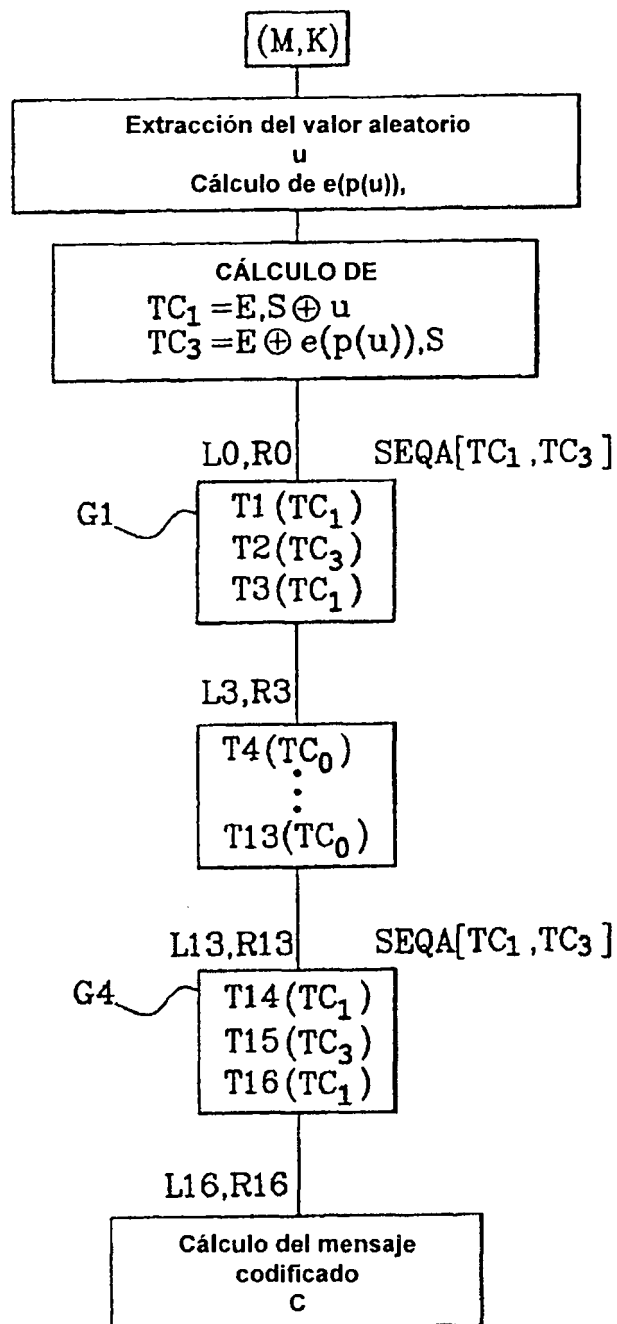


FIG.11

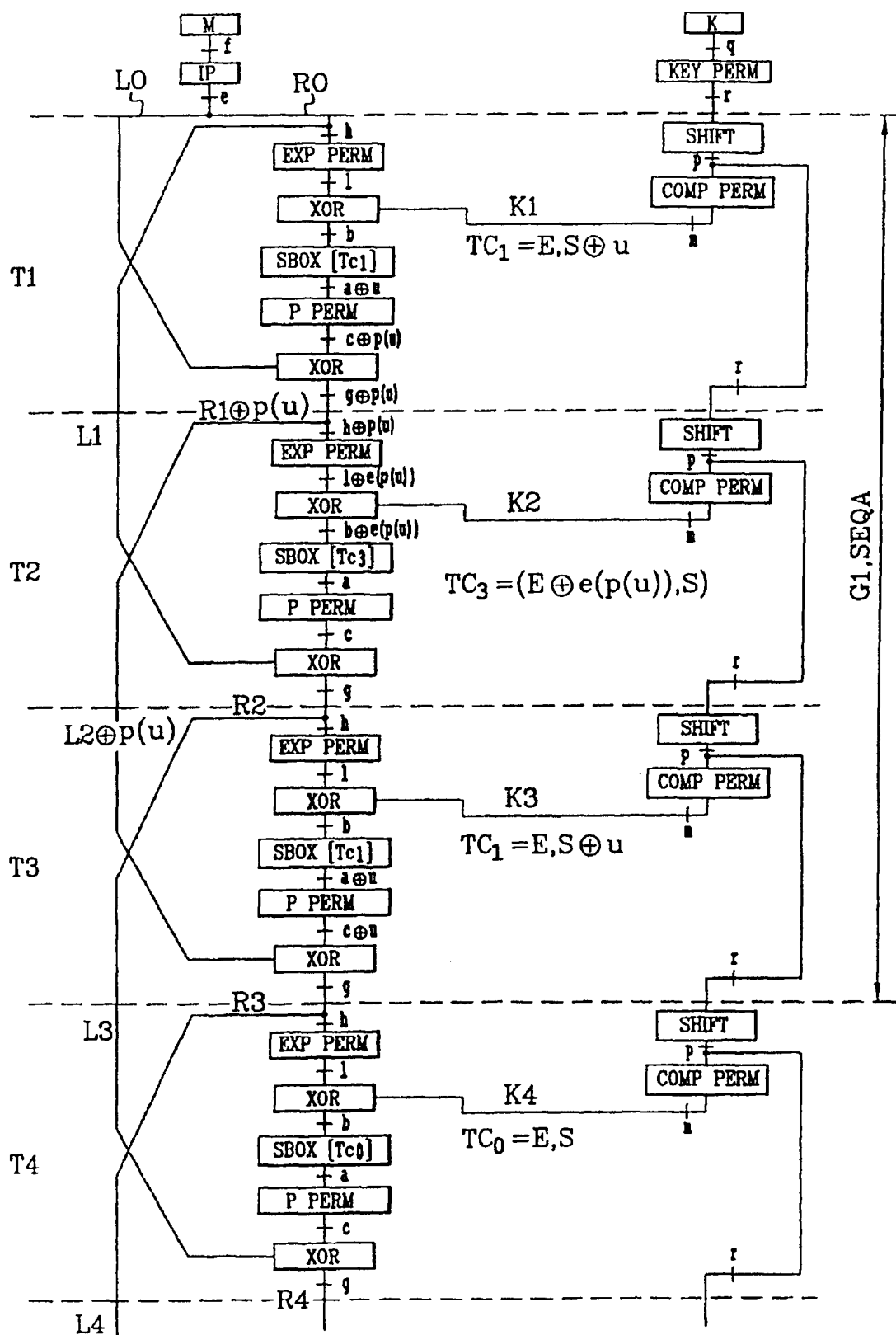


FIG.12

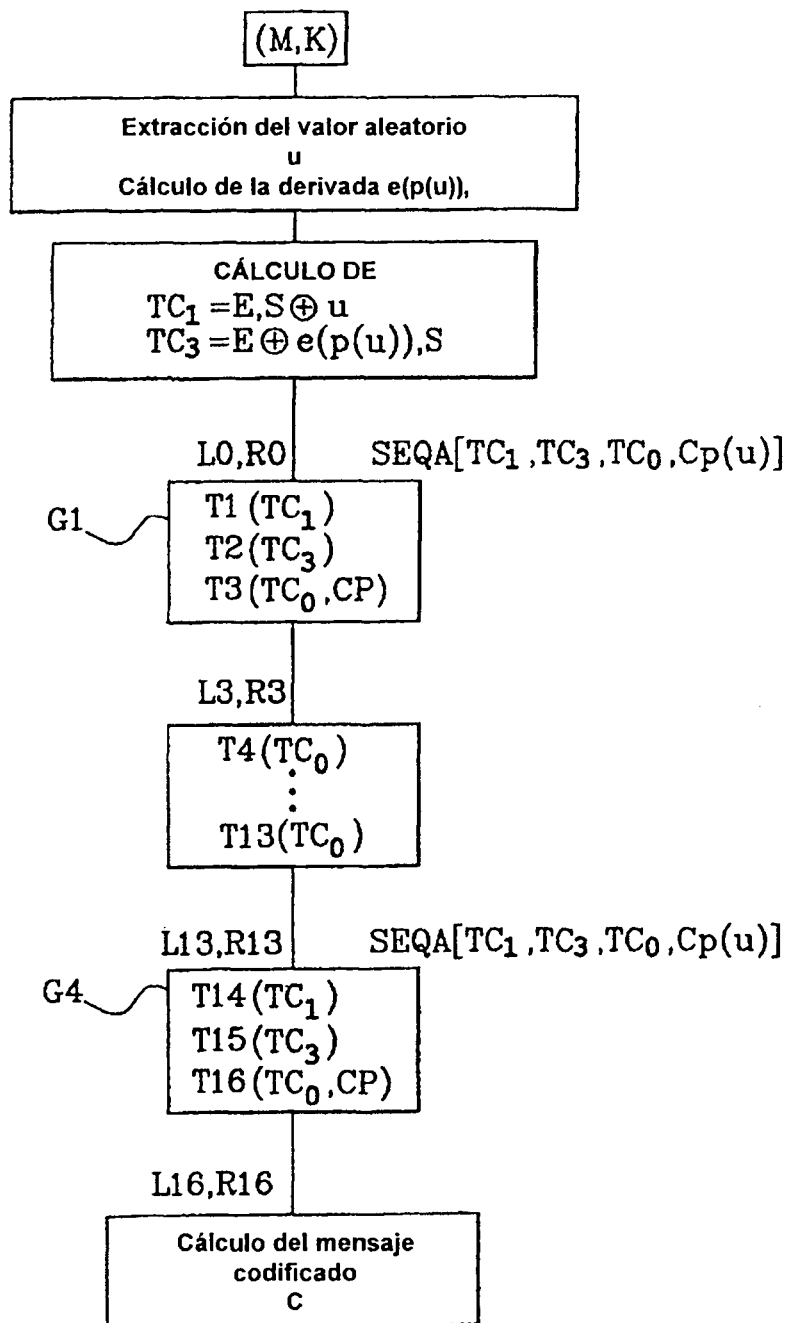


FIG.13