



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2008-0071697
(43) 공개일자 2008년08월05일

(51) Int. Cl.

G06F 15/00 (2006.01)

(21) 출원번호 10-2007-0009901

(22) 출원일자 2007년01월31일

심사청구일자 2007년01월31일

(71) 출원인

고려대학교 산학협력단

서울 성북구 안암동5가1 고려대학교 내

한국정보보호진흥원

서울특별시 송파구 가락동 78번지 IT벤처타워 서관

(72) 발명자

홍득조

경기 용인시 수지구 죽전2동 동성1차아파트 102동 1303호

성재철

서울 강북구 수유2동 벽산아파트 3동 1201호

(뒷면에 계속)

(74) 대리인

김동진, 강경찬, 박경찬

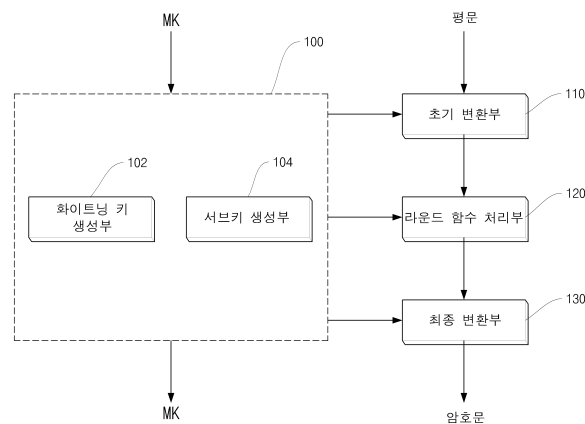
전체 청구항 수 : 총 18 항

(54) 초경량, 저전력 환경에 적합한 암호화 방법

(57) 요약

본 발명은 초경량, 저전력 환경에 적용되는 암호화 방법에 관한 것이다. 본 발명의 암호화 방법은 마스터 키를 입력받아 화이트닝 키 및 서브키를 생성하는 단계와, 평문을 입력받아 상기 생성한 화이트닝 키를 이용하여 초기 라운드 함수값으로 변환하는 단계와, 상기 변환된 초기 라운드 함수값 및 상기 생성한 서브키를 이용하여 복수의 라운드 함수를 처리하는 단계와, 상기 라운드 함수의 처리로 생성된 라운드 함수값과 상기 생성한 화이트닝 키를 이용하여 암호문을 생성하는 단계를 포함하는 것을 특징으로 한다. 상기의 본 발명은 덧셈, 뺄셈, 배타적 논리합이라는 간단한 연산을 사용하므로, 초경량, 저전력 환경에서도 안전하고 효율성이 있는 암호화를 제공할 수 있다.

대표도 - 도1



(72) 발명자

홍석희

서울 은평구 응암2동 451-12

임종인

서울 종로구 내수동 95 삼성파크팰리스 101동 901호

이상진

서울 노원구 중계본동 136 대림벽산아파트 103동 1102호

이창훈

서울 성북구 동소문동6가 143-5 해피엔딩오피스텔 1006호

장동훈

서울 강서구 화곡7동 1082-21

이제상

경기 수원시 영통구 영통동 신나무실 주공아파트 518동 604호

정기태

서울 성북구 안암동5가 12-14 305호

김중성

충북 청주시 흥덕구 사창동 189-15

김현

서울 중구 신당2동 432-498 다오아트빌 101호

지성택

대전 유성구 노은동 열매마을8단지 804-1801호

구본석

대전 유성구 관평동 테크노밸리 쌍용아파트 410-1801

특허청구의 범위

청구항 1

마스터 키를 입력받아 화이트닝 키 및 서브키를 생성하는 단계;

평문을 입력받아 상기 생성한 화이트닝 키를 이용하여 초기 라운드 함수값으로 변환하는 단계;

상기 변환된 초기 라운드 함수값 및 상기 생성한 서브키를 이용하여 복수의 라운드 함수를 처리하는 단계;

상기 라운드 함수의 처리로 생성된 라운드 함수값과 상기 생성한 화이트닝 키를 이용하여 암호문을 생성하는 단계; 를 포함하는 암호화 방법.

청구항 2

제 1항에 있어서,

상기 마스터 키는 8개의 서브 마스터 키로 구성되며, 상기 서브 마스터 키의 길이는 8비트이고, 상기 마스터 키는 16 바이트인 것을 특징으로 하는 암호화 방법.

청구항 3

제 2항에 있어서,

상기 마스터 키는 복수의 서브 마스터 키의 연접으로 구성되는 것을 특징으로 하는 암호화 방법.

청구항 4

제 3항에 있어서, 상기 서브키를 생성하는 단계는

복수의 상수를 생성하는 단계;

상기 생성한 복수의 상수와 상기 서브 마스터 키를 이용하여 서브키를 생성하는 단계; 를 포함하는 것을 특징으로 하는 암호화 방법.

청구항 5

제 3항에 있어서,

상기 화이트닝 키는 하나의 서브 마스터 키를 하나의 화이트닝 키로 매칭시켜 생성되는 것을 특징으로 하는 암호화 방법.

청구항 6

제 5항에 있어서,

상기 제 12 내지 제 15 서브 마스터 키 각각은 제 0 내지 제 3 화이트닝 키에 매칭되며,

상기 제 0 내지 제 3 서브 마스터 키 각각은 제 4 내지 제 7 화이트닝 키에 매칭되는 것을 특징으로 하는 암호화 방법.

청구항 7

제 4항에 있어서,

상기 생성되는 복수의 상수는 128개이며, 각 상수는 1비트의 길이를 갖는 7개의 서브 상수들의 연접으로 구성되는 것을 특징으로 하는 암호화 방법.

청구항 8

제 7항에 있어서,

상기 제 0 상수는 0101101_2 이고, 상기 생성되는 복수의 각 상수는 127의 주기를 갖는 것을 특징으로 하는 암호화 방법.

청구항 9

제 7항에 있어서,

상기 제 $i+6$ 서브상수는 제 $i+2$ 서브상수와 제 $i-1$ 서브상수와의 논리접 배타합의 연산결과를 매칭시킴으로서 생성되고, 상기 i 는 127이하의 자연수인 것을 특징으로 하는 암호화 방법.

청구항 10

제 7항에 있어서,

상기 제 i 상수는 제 i 서브상수 내지 제 $i+6$ 서브상수들의 연접으로 구성되고, 상기 i 는 127이하의 자연수인 것을 특징으로 하는 암호화 방법.

청구항 11

제 10항에 있어서,

상기 서브키는 상기 서브 마스터 키와 상기 상수와의 법 2^8 덧셈연산 결과를 매칭시켜 생성되고, 128개인 것을 특징으로 하는 암호화 방법.

청구항 12

제 11항에 있어서,

상기 제 $16i+j$ 서브키는 제 $(j-i) \bmod 8$ 서브 마스터키와 제 $16i+j$ 상수간의 법 2^8 덧셈연산 결과를 매칭시킴으로서 형성되고,

상기 제 $16i+j+8$ 서브키는 제 $((j-i) \bmod 8)+8$ 서브 마스터키와 제 $16i+j+8$ 상수간의 법 2^8 덧셈연산의 결과를 매칭시킴으로서 형성되며,

상기 i, j 는 0 이상 7 이하의 자연수인 것을 특징으로 하는 암호화 방법.

청구항 13

제 11항에 있어서,

상기 복수의 라운드 함수의 처리는 32번의 라운드 함수를 처리하는 것을 특징으로 하는 암호화 방법.

청구항 14

제 13항에 있어서,

제 $2n$ 초기 라운드 함수값은 제 $2n$ 초기 평문과 제 n 화이트닝 키와의 법 2^8 덧셈연산의 결과를 매칭시킴으로서 형성되며, n 은 짝수이고,

제 $2k$ 초기 라운드 함수값은 제 $2k$ 초기 평문과 제 k 화이트닝 키와의 논리적 배타합의 결과를 매칭시킴으로서 형성되며, k 는 홀수이고,

제 $2r-1$ 초기 라운드 함수값은 제 $2r-1$ 초기 평문을 매칭시킴으로서 형성되며, r 은 자연수이고,

상기 평문은 8바이트인 것을 특징으로 하는 암호화 방법.

청구항 15

제 14항에 있어서,

제 $i+1$ 라운드 함수의 제 $2n-1$ 번째 함수값 각각은 제 i 라운드 함수의 제 $2n$ 번째 함수값을 매칭시킴으로서 형성되며,

제 $i+1$ 라운드 함수의 제 0 번째 함수값은 제 i 라운드 함수의 제 6 번째 함수값의 제 1 순환 이동연산값과 제 $4i+3$ 서브키와의 법 2^8 덧셈연산 결과를 제 i 라운드 함수의 제 7 번째 함수값과의 배타적 논리합 연산으로 결

정되며,

제 i+1 라운드 함수의 제 2 번째 함수값은 제 i 라운드 함수의 제 0 번째 함수값의 제 2 순환 이동연산값과 제 4i+2 서브키와의 배타적 논리합 결과를 제 i 라운드 함수의 제 1 번째 함수값과의 범 2^8 덧셈연산으로 결정되고,

제 i+1 라운드 함수의 제 4 번째 함수값은 제 i 라운드 함수의 제 2 번째 함수값의 제 1 순환 이동연산값과 제 4i+1 서브키와의 범 2^8 덧셈연산 결과를 제 i 라운드 함수의 제 3 번째 함수값과의 배타적 논리합 연산으로 결정되며,

제 i+1 라운드 함수의 제 6 번째 함수값은 제 i 라운드 함수의 제 4 번째 함수값의 제 2 순환 이동연산값과 제 4i 서브키와의 배타적 논리합 결과를 제 i 라운드 함수의 제 5 번째 함수값과의 범 2^8 덧셈연산으로 결정되며,

상기 i는 31이하의 자연수인 것을 특징으로 하는 암호화 방법.

청구항 16

제 15 항에 있어서,

상기 각 제 1 순환 이동연산 및 제 2 순환 이동연산은 하기와 같이 표현되는 것을 특징으로 하는 암호화 방법.

$$F_0(x) = x \ll 1 \oplus x \ll 2 \oplus x \ll 7,$$

$$F_1(x) = x \ll 3 \oplus x \ll 4 \oplus x \ll 6$$

청구항 17

제 16항에 있어서,

상기 암호문은 8개의 서브 암호문의 연접으로 구성되는 것을 특징으로 하는 암호화 방법.

청구항 18

제 17항에 있어서,

제 0 서브 암호문은 제 32 라운드 함수의 제 1 번째 함수값과 제 4 화이트닝 키와의 범 2^8 덧셈연산 결과를 매칭시킴으로서 형성되고,

제 1 서브 암호문은 제 32 라운드 함수의 제 2 번째 함수값을 매칭시킴으로서 형성되며,

제 2 서브 암호문은 제 32 라운드의 제 3 번째 함수값과 제 5 화이트닝 키와의 배타적 논리합의 연산 결과를 매칭시킴으로서 형성되고,

제 3 서브 암호문은 제 32 라운드 함수의 제 4 번째 함수값을 매칭시킴으로서 형성되며,

제 4 서브 암호문은 제 32 라운드 함수의 제 5 번째 함수값과 제 6 화이트닝 키와의 범 2^8 덧셈연산 결과를 매칭시킴으로서 형성되고,

제 5 서브 암호문은 제 32 라운드 함수의 제 6 번째 함수값을 매칭시킴으로서 형성되며,

제 6 서브 암호문은 제 32 라운드 함수의 제 7 번째 함수값과 제 7 화이트닝 키와의 배타적 논리합의 연산 결과를 매칭시킴으로서 형성되고,

제 7 서브 암호문은 제 32 라운드의 제 0 번째 함수값을 매칭시킴으로서 형성되는 것을 특징으로 하는 암호화 방법.

명 세 서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

- <14> 본 발명은 암호화 방법에 관한 것으로, 특히 초경량, 저전력 환경에 적용되는 암호화 방법에 관한 것이다.
- <15> 컴퓨터 보급의 확대와 인터넷 사용자의 증가는 정보 보호의 중요성을 증대시키고 있다. 정보 보호는 정보 보호 서비스의 제공에 의하여 이루어지며, 정보 보호 서비스를 제공하는 가장 일반적인 방법은 암호 시스템을 사용하는 것이다.
- <16> 암호 시스템은 정보에 대한 다양한 공격으로부터 정보를 보호하기 위한 것이다. 공격의 유형은 방해(interruption), 가로채기(interception), 위조/변조(Modification), 위장(fabrication) 등이 있다. 이러한 공격 방법은 서로 유기적인 관계가 있으므로, 정보를 안전하게 보호하기 위해서는 각각의 공격방법으로부터 안전한 메커니즘을 찾고, 이들을 조합한 보안 기술을 이용하여 정보가 안전하도록 해야 한다. 이를 위하여 현재까지 DES(Data Encryption Standard), RSA(Rivest, Shamir, Adleman), AES(Advanced Encryption Standard)등의 다양한 암호 기술이 개발되었다.
- <17> 상기의 다양해진 암호기술이 개발되었으나, 현재까지의 암호 기술들은 효율성보다는 안전성에 중점을 두고 개발되었다. 상기의 암호기술들은 초경량·저전력 계산환경에서 고속의 암호연산이 필요한 유비쿼터스 컴퓨팅환경에 적용이 어려웠다. 즉, 상기의 암호기술들은 최근에 확산되고 있는 RFID(Radio Frequency Identification) 등과 같이 초고속, 초경량을 요구하는 응용환경에 적용하는데 많은 어려움이 있다.

발명이 이루고자 하는 기술적 과제

- <18> 그러므로 본 발명의 목적은 초경량·저전력이 요구되는 응용환경에서도 정보의 보호를 위하여 효율적이고 안전성이 있는 암호화 방법을 제공하는 데 있다.

발명의 구성 및 작용

- <19> 본 발명의 암호화 방법은 마스터 키를 입력받아 화이트닝 키 및 서브키를 생성하는 단계와, 평문을 입력받아 상기 생성한 화이트닝 키를 이용하여 초기 라운드 함수값으로 변환하는 단계와, 상기 변환된 초기 라운드 함수값 및 상기 생성한 서브키를 이용하여 복수의 라운드 함수를 처리하는 단계와, 상기 라운드 함수의 처리로 생성된 라운드 함수값과 상기 생성한 화이트닝 키를 이용하여 암호문을 생성하는 단계를 포함하는 것을 특징으로 한다.
- <20> 상기 서브키를 생성하는 단계는 복수의 상수를 생성하는 단계와, 상기 생성한 복수의 상수와 상기 서브 마스터 키를 이용하여 서브키를 생성하는 단계를 포함하는 것을 특징으로 한다.
- <21> 이하, 첨부된 도면을 참조하여 본 발명의 암호화 방법을 상세히 설명한다.
- <22> 도 1은 본 발명의 암호화 장치의 구성을 나타낸 블록도이다. 도시된 바와 같이 본 발명의 암호화 장치는 키 스케줄러부(100), 초기 변환부(110), 라운드 함수 처리부(120) 및 최종 변환부(130)를 포함한다.
- <23> 상기 키 스케줄러부(100)는 화이트닝 키 생성부(102) 및 서브키 생성부(104)를 포함한다. 상기 키 스케줄러부(100)는 마스터 키(Master Key, 이하 'MK'라 한다.)를 입력받아 화이트닝 키(Whitening Key, 이하 'WK'라 한다.)와 서브키(Subkey, 이하 'SK'라 한다.)를 생성한다.
- <24> 상기 키 스케줄러부(100)는 외부로부터 마스터 키를 입력받는다. 상기 마스터 키는 사용자 인터페이스(User Interface) 등을 통해 입력받는다. 마스터 키는 복수의 서브 마스터 키들로 구성되며, 상기 서브 마스터 키는 그 길이가 8비트(Bit)이다. 상기 키 스케줄러부(100)가 입력받는 마스터 키는 16 바이트(byte)로 구성되어 있으며, 그 총 길이는 128비트가 된다. 즉, 마스터 키는 복수의 서브 마스터 키의 연접으로 구성되며, 이는 하기의 수학식 1과 같다.

수학식 1

- <25>
$$MK = MK_{15} \parallel MK_{14} \parallel \dots \parallel MK_1 \parallel MK_0$$

- <26> 마스터 키는 평문이라는 암호화 하고자 하는 대상, 예를 들면 특정 텍스트(text) 또는 음성(Voice)이 입력되는 시점에 동시에 입력된다.

<27> 화이트닝 키 생성부(102)는 상기 키 스케줄러부(100)가 외부로부터 입력받은 서브 마스터 키를 이용하여 화이트닝 키를 생성한다. 화이트닝 키 생성부(102)는 8비트로 구성된 하나의 서브 마스터 키를 하나의 화이트닝 키로 매칭시킴으로서, 8비트의 화이트닝 키를 생성한다. 단, 화이트닝 키 생성부(102)가 화이트닝 키를 생성하는 데 있어서, 8 바이트의 서브 마스터 키 즉, 8개의 서브 마스터 키를 이용한다.

<28> 도 2는 화이트닝 키 생성부가 화이트닝 키를 생성하는 알고리즘의 실시예를 보인 도면이다. 도시된 바와 같이, 제 12 내지 제 15 서브 마스터 키(MK₁₂ 내지 MK₁₅)는 제 0 내지 제 3 화이트닝 키(WK₀ 내지 WK₃)에 각각 매칭되어, 제 0 내지 제 3 화이트닝 키(WK₀ 내지 WK₃)가 형성된다. 또한, 제 0 내지 제 3 서브 마스터 키(MK₀ 내지 MK₃)는 제 4 내지 제 7 화이트닝 키(WK₄ 내지 WK₇)에 각각 매칭되어, 제 4 내지 제 7 화이트닝 키(WK₄ 내지 WK₇)가 형성된다.

<29> 도 3은 상기 서브키 생성부의 구성을 나타낸 블록도이다. 상기 서브키 생성부(104)는 상수 생성부(300)와 서브키 형성부(310)를 포함한다. 상기 서브키 생성부(104)는 복수의 상수를 생성하고, 상기 생성한 상수와 마스터 키를 이용하여 서브키를 생성한다.

<30> 상수 생성부(300)는 서브키 생성에 이용되는 복수의 상수를 생성한다. 상수 생성부(300)가 생성하는 복수의 상수(δ)는 128개이며, 각 상수는 7비트로 구성된다. 복수의 각 상수는 7개의 서브상수(s)의 연결로 구성되며, 하기의 수학식 2와 같이 표현된다.

수학식 2

<31> $\delta_0 \leftarrow s_6 \parallel s_5 \parallel s_4 \parallel s_3 \parallel s_2 \parallel s_1 \parallel s_0$

<32> 제 0 상수는 초기 상수값이 되며, 0101101₂의 값을 갖는다. 즉, 제 0 서브상수(s₀)는 1에 매칭되며, 제 1 서브상수(s₁)는 0에 매칭되며, 제 2 서브상수(s₂)는 1에 매칭되며, 제 3 서브상수(s₃)는 1에 매칭되며, 제 4 서브상수(s₄)는 0에 매칭되며, 제 5 서브상수(s₅)는 1에 매칭되며, 제 6 서브상수(s₆)는 0에 매칭된다. 상수 생성부(300)는 복수의 상수를 생성하는 데 있어서 7비트 LFSR h(Linear Feedback Shift Register h)를 이용한다. 상기 복수의 상수는 127의 주기를 갖는다. 왜냐하면, LFSR h의 연결 다항식은 하기의 수학식 3과 같이 표현되고,

수학식 3

<33> $x^7 + x^3 + 1 \in \mathbb{Z}_2[x]$

<34> 상기 다항식은 $\mathbb{Z}_2[x]$ 에서 원시 다항식이기 때문에 h는 2^7-1 의 주기를 갖게 된다. 다만, 상수 생성부(300)가 복수의 상수를 생성하는 데 있어서 LFSR h에 한정되는 것은 아니다.

<35> 도 4는 상수 생성부가 복수의 상수를 생성하는 알고리즘의 실시예를 보인 도면이다. 도시된 바와 같이, 초기 상수값(δ_0)을 설정한다. 상기 초기 상수값은 복수의 각 서브상수(s₆, s₅, s₄, s₃, s₂, s₁, s₀)를 0,1,0,1,1,0,1에 매칭시킴으로서 설정된다.

<36> 상기 상수 생성부(300)는 상기 초기 상수값을 설정한 이후에, 127개의 복수의 서브상수(s₇ 내지 s₁₃₃)를 생성한다. 상기한 바와 같이 제 0 서브상수(s₀) 내지 제 6 서브상수(s₆)의 서브상수가 생성되었으므로, 상기 상수 생성부(300)는 제 7 서브상수(s₇) 내지 제 133 서브상수(s₁₃₃)인 127개의 서브상수가 생성된다. 127개의 서브상수를 생성하는 데 있어서, 제 i+2 서브상수(s_{i+2})와 제 i-1 서브상수(s_{i-1})의 논리적 배타합의 연산결과를 매칭시켜 제 i+6 서브상수(s_{i+6})를 생성한다.

<37> 상기 생성한 127개의 서브상수(s)를 생성한 이후에, 127개의 상수(δ)가 생성된다. 즉, 제 i 상수(δ_i)는 제 i 서브상수(s_i) 내지 제 i+6 서브상수(s_{i+6})의 연결을 매칭시킴으로서 형성된다. 상기 연결은 하기의 수학식 4와 같이 표현된다.

수학식 4

- <38> $\delta_i \leftarrow s_{i+6} \parallel s_{i+5} \parallel s_{i+4} \parallel s_{i+3} \parallel s_{i+2} \parallel s_{i+1} \parallel s_i$
- <39> 상기 i 는 127 이하의 자연수이다.
- <40> 서브키 형성부(310)는 상기 상수 생성부(300)가 생성한 복수의 상수 및 상기 서브 마스터 키를 이용하여 서브키를 생성한다. 서브키 형성부(310)는 8비트의 서브 마스터 키와 8비트의 상수와의 법 2^8 덧셈연산 결과를 매칭시켜 8비트의 서브키를 생성한다. 상기 법 2^8 덧셈연산은 2진수 8비트로 구성된 복수의 수들의 합연산을 의미한다. 서브키 형성부(310)가 생성하는 서브키는 128개가 된다. 도 4는 서브키 형성부(310)가 서브키를 생성하는 알고리즘의 실시예에 관한 도면이다. 도시된 바와 같이, 제 $16i+j$ 서브키(SK_{16i+j})는 제 $(j-i) \bmod 8$ 서브 마스터 키($MK_{(j-i) \bmod 8}$)와 제 $16i+j$ 상수(δ_{16i+j})간의 법 2^8 덧셈연산의 결과를 매칭시킴으로서 형성된다. 상기 연산으로 인하여 64개의 서브키가 생성되고, 나머지 64개의 서브키는, 제 $16i+j+8$ 서브키($SK_{16i+j+8}$)는 제 $((j-i) \bmod 8)+8$ 서브 마스터 키($MK_{((j-i) \bmod 8)+8}$)와 제 $16i+j+8$ 상수($\delta_{16i+j+8}$)간의 법 2^8 덧셈연산의 결과를 매칭시킴으로서 형성된다. 상기 연산으로 인하여 128개의 서브키가 생성된다. 상기 서브키 형성부(310)에서의 i, j 는 0 이상 7 이하의 자연수이다. 상기 $\bmod 8$ 은 특정값을 8로 나눈 나머지 값을 의미한다.
- <41> 초기 변환부(110)는 외부로부터 입력받은 평문을 상기 화이트닝 키 생성부(102)가 생성한 화이트닝 키를 이용하여 초기 라운드 함수값으로 변환한다. 상기 평문은 암호의 대상이 되는 특정 텍스트 또는 보이스가 되며, 사용자 인터페이스 등을 통해 입력받는다. 또한, 상기 평문은 8바이트의 길이를 갖는다. 초기 변환부(110)에서 변환되는 초기 라운드 함수값은 8개가 된다. 도 6은 초기 변환부에서 평문이 초기 라운드 함수값으로 변환되는 알고리즘의 실시예를 보인 도면이다. 도시된 바와 같이, 제 $2n$ 초기 라운드 함수값($X_{0,2n}$)은 제 $2n$ 초기 평문(P_{2n})과 제 n 화이트닝 키(WK_n)와의 법 2^8 덧셈연산의 결과를 매칭시킴으로서 형성한다. 상기 n 은 짝수이다. 그리고 제 $2k$ 초기 라운드 함수값($X_{0,2k}$)은 제 $2k$ 초기 평문(P_{2k})과 제 k 화이트닝 키(WK_k)와의 논리적 배타합의 결과를 매칭시킴으로서 형성한다. 상기 k 는 홀수이다. 또한, 제 $2r-1$ 초기 라운드 함수값($X_{0,2r-1}$)은 제 $2r-1$ 초기 평문(P_{2r-1})을 매칭시켜 형성한다. 상기 r 은 자연수이다.
- <42> 라운드 함수 처리부(120)는 상기 초기 변환부(110)에서 변환된 초기 라운드 함수값 및 상기 서브키 생성부(104)가 생성한 서브키를 이용하여 32번의 라운드 함수를 처리한다. 라운드 함수 처리부(120)가 라운드 함수를 처리하는 데 있어서, 제 $i+1$ 라운드 함수는 제 i 번째 함수값($X_{i,0}, X_{i,1}, \dots, X_{i,7}$) 및 제 $4i$ 서브키(SK_{4i}) 내지 제 $4i+3$ 서브키(SK_{4i+3})를 이용한다. 다만, 제 1 라운드 함수는 초기 함수값($X_{0,0}, X_{0,1}, \dots, X_{0,7}$)이 상기 i 번째 함수값이 된다. 그리고 라운드 함수 처리부(120)에서 상기 i 는 32 이하의 자연수이다.
- <43> 도 7은 라운드 함수 처리부가 제 $i+1$ 라운드 함수를 처리하는 알고리즘의 실시예를 보인 도면이다. 도시된 바와 같이, 제 $i+1$ 라운드 함수의 제 $2n-1$ 번째 함수값($X_{i+1,1}, X_{i+1,3}, X_{i+1,5}, X_{i+1,7}$) 각각은 제 i 라운드 함수의 제 $2n$ 번째 함수값($X_{i,0}, X_{i,2}, X_{i,4}, X_{i,6}$)을 매칭시킴으로써 생성된다. 그리고 제 $i+1$ 라운드 함수의 제 0 번째 함수값($X_{i+1,0}$)은 제 i 라운드 함수의 제 6 번째 함수값($X_{i,6}$)의 제 1 순환 이동연산값과 제 $4i+3$ 서브키(SK_{4i+3})와의 법 2^8 덧셈연산 결과를 제 i 라운드 함수의 제 7 번째 함수값($X_{i,7}$)과의 배타적 논리합 연산으로 결정된다. 또한, 제 $i+1$ 라운드 함수의 제 2 번째 함수값($X_{i+1,2}$)은 제 i 라운드 함수의 제 0 번째 함수값($X_{i,0}$)의 제 2 순환 이동연산값과 제 $4i+2$ 서브키(SK_{4i+2})와의 배타적 논리합 결과를 제 i 라운드 함수의 제 1 번째 함수값($X_{i,1}$)과의 법 2^8 덧셈연산으로 결정된다. 그리고 제 $i+1$ 라운드 함수의 제 4 번째 함수값($X_{i+1,4}$)은 제 i 라운드 함수의 제 2 번째 함수값($X_{i,2}$)의 제 1 순환 이동연산값과 제 $4i+1$ 서브키(SK_{4i+1})와의 법 2^8 덧셈연산 결과를 제 i 라운드 함수의 제 3 번째 함수값($X_{i,3}$)과의 배타적 논리합 연산으로 결정된다. 또한, 제 $i+1$ 라운드 함수의 제 6 번째 함수값($X_{i+1,6}$)은 제 i 라운드 함수의 제 4 번째 함수값($X_{i,4}$)의 제 2 순환 이동연산값과 제 $4i$ 서브키(SK_{4i})와의 배타적 논리합 결과를 제 i 라운드 함수의 제 5 번째 함수값($X_{i,5}$)과의 법 2^8 덧셈연산으로 결정된다. 상기 i 는 32보다 작은 자연수이며, 상기 제 1 순환 이동은 하기 수학식 5, 상기 제 2 순환 이동은 하기 수학식 6과 같이 표

현된다.

수학식 5

<44>

$$F_0(x) = x^{\ll 1} \oplus x^{\ll 2} \oplus x^{\ll 7},$$

수학식 6

<45>

$$F_1(x) = x^{\ll 3} \oplus x^{\ll 4} \oplus x^{\ll 6}$$

<46>

상기 최종 변환부(130)는 상기 라운드 처리부가 생성한 라운드 함수값과 상기 화이트닝 키 생성부(102)가 생성한 화이트닝 키를 이용하여 암호문을 출력한다. 상기 최종 변환부(130)는 8개의 서브 암호문을 생성하며, 상기 8개의 서브 암호문으로 암호문이 구성된다. 8개의 서브 암호문이 암호문을 구성하는 것은 수학식 7과 같이 표현된다.

수학식 7

<47>

$$C = C_7 \parallel C_6 \parallel C_5 \parallel C_4 \parallel C_3 \parallel C_2 \parallel C_1 \parallel C_0$$

<48>

도 8은 최종 변환부(130)가 복수의 각 서브 암호문을 생성하는 알고리즘의 실시예를 보인 도면이다. 도시된 바와 같이, 제 0 서브 암호문(C_0)은 제 32 라운드 함수의 제 1 번째 함수값($X_{32,1}$)과 제 4 화이트닝 키(WK_4)와의 법 2^8 덧셈연산 결과를 매칭시킴으로서 형성된다. 제 1 서브 암호문(C_1)은 제 32 라운드 함수의 제 2 번째 함수값($X_{32,2}$)을 매칭시킴으로서 형성된다. 제 2 서브 암호문(C_2)은 제 32 라운드의 제 3 번째 함수값($X_{32,3}$)과 제 5 화이트닝 키(WK_5)와의 배타적 논리합의 연산 결과를 매칭시킴으로서 형성된다. 제 3 서브 암호문(C_3)은 제 32 라운드 함수의 제 4 번째 함수값($X_{32,4}$)을 매칭시킴으로서 형성된다. 제 4 서브 암호문(C_4)은 제 32 라운드 함수의 제 5 번째 함수값($X_{32,5}$)과 제 6 화이트닝 키(WK_6)와의 법 2^8 덧셈연산 결과를 매칭시킴으로서 형성된다. 제 5 서브 암호문(C_5)은 제 32 라운드 함수의 제 6 번째 함수값($X_{32,6}$)을 매칭시킴으로서 형성된다. 제 6 서브 암호문(C_6)은 제 32 라운드 함수의 제 7 번째 함수값($X_{32,7}$)과 제 7 화이트닝 키(WK_7)와의 배타적 논리합의 연산 결과를 매칭시킴으로서 형성된다. 제 7 서브 암호문(C_7)은 제 32 라운드의 제 0 번째 함수값($X_{32,0}$)을 매칭시킴으로서 형성된다.

<49>

상기의 과정들을 역으로 수행하는 경우에는 암호화된 것을 평문으로 복호화할 수 있다. 즉, 복호와 과정에서는 라운드 함수는 법 2^8 덧셈연산 대신에 법 2^8 뺄셈연산을 사용하고 매칭시키는 방향도 역방향으로 형성함으로써 구현이 가능하다.

<50>

한편, 상기에서는 본 발명을 특정의 바람직한 실시 예에 관련하여 도시하고 설명하였지만, 이하의 특허청구범위에 의해 마련되는 본 발명의 정신이나 분야를 이탈하지 않는 한도내에서 본 발명이 다양하게 개조 및 변화될 수 있다는 것을 당업계에서 통상의 지식을 가진자는 용이하게 알 수 있다.

발명의 효과

<51>

이상에서 설명한 바와 같이, 본 발명의 암호화 방법은 기본적인 산술 연산인 배타적인 논리합, 덧셈, 순환 이동과 같은 매우 간단한 구조로 구현이 가능하므로, 매우 간단한 구조를 초경량·저전력 환경에서도 안전성과 효율성을 동시에 제공할 있다.

도면의 간단한 설명

<1>

도 1은 본 발명의 암호화 장치의 구성을 나타낸 블록도.

<2>

도 2는 화이트 키 생성부가 화이트 키를 생성하는 알고리즘의 실시예를 보인 도면.

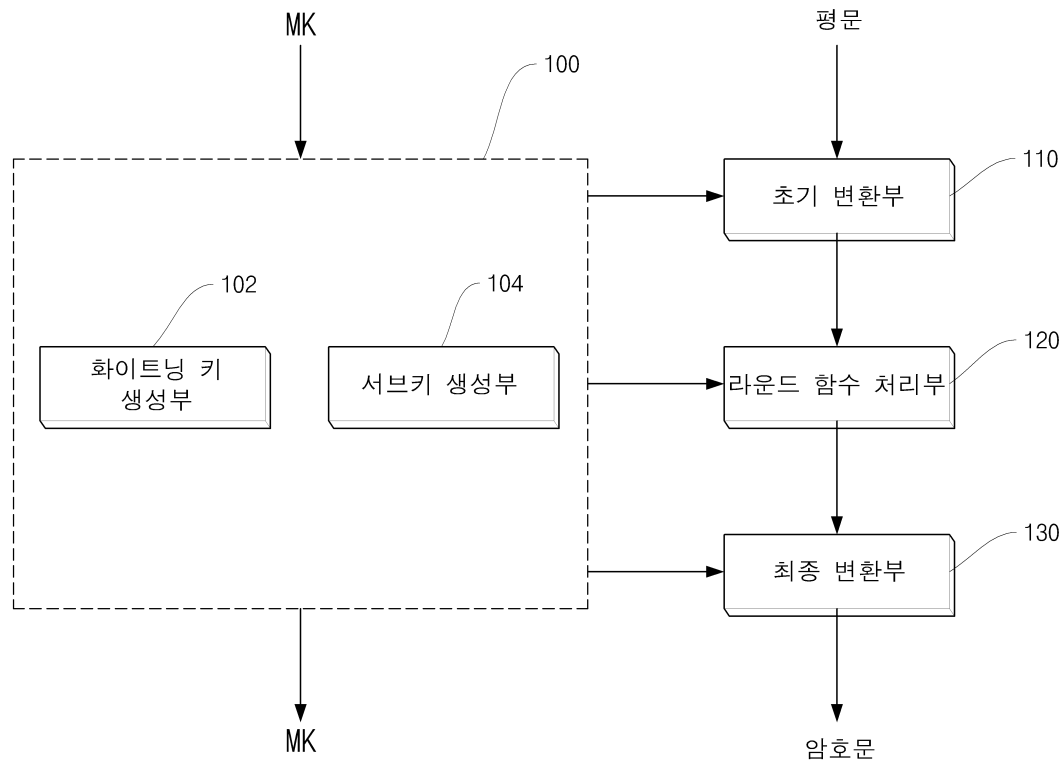
<3>

도 3은 서브키 생성부의 구성을 나타낸 블록도.

- <4> 도 4는 상수 생성부가 복수의 상수를 생성하는 알고리즘의 실시예를 보인 도면.
- <5> 도 5는 서브키 형성부가 서브키를 생성하는 알고리즘의 실시예에 관한 도면.
- <6> 도 6은 초기 변환부에서 평문이 초기 라운드 함수값으로 변환되는 알고리즘의 실시예를 보인 도면.
- <7> 도 7은 라운드 함수 처리부가 제 i+1 라운드 함수를 처리하는 알고리즘의 실시예를 보인 도면.
- <8> 도 8은 최종 변환부가 복수의 각 서브 암호문을 생성하는 알고리즘의 실시예를 보인 도면.
- <9> *도면의 주요부분에 대한 부호설명*
- <10> 100 : 키 스케줄러부 102 : 화이트닝 키 생성부
- <11> 104 : 서브키 생성부 110 : 초기 변환부
- <12> 120 : 라운드 함수 처리부 130 : 최종 변환부
- <13> 300 : 상수 생성부 310 : 서브키 형성부

도면

도면1



도면2

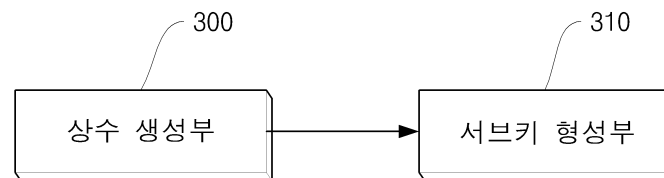
```

WhiteningKeyGeneration {
  For  $i = 0$  to  $7$  {
    If  $0 \leq i \leq 3$ , then  $WK_i \leftarrow MK_{i+12}$ ;
    Else,  $WK_i \leftarrow MK_{i-4}$ ;
  }
}

```

도면3

104



도면4

```

ConstantGeneration (MK, SK) {
   $s_0 \leftarrow 0$ ;  $s_1 \leftarrow 1$ ;  $s_2 \leftarrow 0$ ;  $s_3 \leftarrow 1$ ;
   $s_4 \leftarrow 1$ ;  $s_5 \leftarrow 0$ ;  $s_6 \leftarrow 1$ ;
   $\delta_0 \leftarrow s_6 \| s_5 \| s_4 \| s_3 \| s_2 \| s_1 \| s_0$ ;
  For  $i = 1$  to  $127$  {
     $s_{i+6} \leftarrow s_{i+2} \oplus s_{i-1}$ ;
     $\delta_i \leftarrow s_{i+6} \| s_{i+5} \| s_{i+4} \| s_{i+3} \| s_{i+2} \| s_{i+1} \| s_i$ ;
  }
}

```

도면5

```

SubkeyGeneration (MK,SK) {
  For i=0 to 7 {
    For j=0 to 7 {
       $SK_{16 \cdot i + j} \leftarrow MK_{(j-i) \bmod 8} \boxplus \delta_{16 \cdot i + j}$ 
    }
    For j=0 to 7 {
       $SK_{16 \cdot i + j + 8} \leftarrow MK_{((j-i) \bmod 8) + 8} \boxplus \delta_{16 \cdot i + j + 8}$ 
    }
  }
}

```

도면6

```

InitialTransformation (P, X0, WK3, WK2, WK1, WK0) {
   $X_{0,0} \leftarrow P_0 \boxplus WK_0$ ;  $X_{0,1} \leftarrow P_1$ ;  $X_{0,2} \leftarrow P_2 \oplus WK_1$ ;  $X_{0,3} \leftarrow P_3$ ;
   $X_{0,4} \leftarrow P_4 \boxplus WK_2$ ;  $X_{0,5} \leftarrow P_5$ ;  $X_{0,6} \leftarrow P_6 \oplus WK_3$ ;  $X_{0,7} \leftarrow P_7$ ;
}

```

도면7

```

RoundFunction (Xi, Xi+1, SK4i+3, SK4i+2, SK4i+1, SK4i) {
   $X_{i+1,1} \leftarrow X_{i,0}$ ;  $X_{i+1,3} \leftarrow X_{i,2}$ ;  $X_{i+1,5} \leftarrow X_{i,4}$ ;  $X_{i+1,7} \leftarrow X_{i,6}$ ;
   $X_{i+1,0} = X_{i,7} \oplus (F_0(X_{i,6}) \boxplus SK_{4i+3})$ ;
   $X_{i+1,2} = X_{i,1} \boxplus (F_1(X_{i,0}) \oplus SK_{4i+2})$ ;
   $X_{i+1,4} = X_{i,3} \oplus (F_0(X_{i,2}) \boxplus SK_{4i+1})$ ;
   $X_{i+1,6} = X_{i,5} \boxplus (F_1(X_{i,4}) \oplus SK_{4i})$ ;
}

```

도면8

FinalTransformation ($X_{32}, C, WK_7, WK_6, WK_5, WK_4$) { $C_0 \leftarrow X_{32,1} \boxplus WK_4; C_1 \leftarrow X_{32,2}; C_2 \leftarrow X_{32,3} \oplus WK_5; C_3 \leftarrow X_{32,4};$ $C_4 \leftarrow X_{32,5} \boxplus WK_6; C_5 \leftarrow X_{32,6}; C_6 \leftarrow X_{32,7} \oplus WK_7; C_7 \leftarrow X_{32,0};$ }
--