



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2011-0105678
(43) 공개일자 2011년09월27일

(51) Int. Cl.

H04L 9/18 (2006.01) H04L 9/14 (2006.01)

(21) 출원번호 10-2010-0024957

(22) 출원일자 2010년03월19일

심사청구일자 2010년03월19일

(71) 출원인

고려대학교 산학협력단

서울 성북구 안암동5가 1

(72) 발명자

홍석희

서울특별시 도봉구 도봉2동 한신아파트 103-107

소현동

부산광역시 해운대구 반여1동 장산롯데 낙천대2차 아파트 203동 1602호

김성경

서울 성북구 안암동2가 118-2번지 SM빌딩 606호

(74) 대리인

특허법인충현

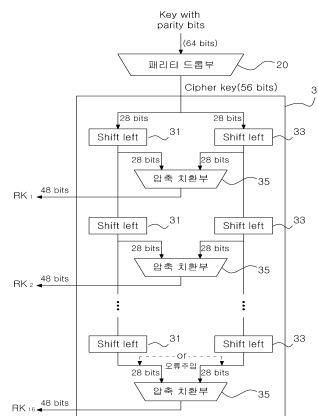
전체 청구항 수 : 총 18 항

(54) 데이터 암호화 표준의 라운드 키 생성과정에서의 차분오류주입공격을 이용한 암호 처리 방법 및 암호 처리 장치

(57) 요약

데이터를 64비트로 나눈 블록을 16라운드 페이스텔(feistel)형 공통키 블록 암호 처리를 실행하는 암호 처리 방법을 개시한다. 상기 방법은 64비트의 평문에 초기 치환을 수행하는 단계, 마스터 키를 페리티 드롭하여 생성된 암호화 키(cipher key)를 비트 왼쪽 순환(left shift) 및 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 비트 왼쪽 순환, 레지스터의 오류 주입 및 압축치환을 통해 제16 라운드 키를 생성하는 단계, 상기 초기 치환이 수행된 64비트의 평문을 두 개의 32비트로 나누어 왼쪽 레지스터 및 오른쪽 레지스터에 저장한 후, 제1 내지 제16 라운드를 통해 상기 제1 내지 제16 라운드 키를 이용한 암호화 연산을 통해 수행하는 단계, 및 상기 제1 내지 제16 라운드의 암호화 연산을 통해 생성된 비트들에 대해 상기 초기 치환의 역과정을 수행하는 단계를 포함한다. 상기 방법은 16라운드 키 생성과정에서 오류를 주입하여 오류 주입 가능 범위를 넓히고 적은 오류를 주입하여 마스터 키를 획득할 수 있다.

대표도 - 도3



특허청구의 범위

청구항 1

데이터를 64비트로 나눈 블록을 16라운드 페이스텔(feistel)형 공통키 블록 암호 처리를 실행하는 암호 처리 방법에 있어서,

64비트의 평문에 초기 치환을 수행하는 단계;

마스터 키를 패리티 드롭하여 생성된 암호화 키(cipher key)를 비트 왼쪽 순환(left shift) 및 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 비트 왼쪽 순환, 레지스터의 오류 주입 및 압축치환을 통해 제16 라운드 키를 생성하는 단계;

상기 초기 치환이 수행된 64비트의 평문을 두 개의 32비트로 나누어 왼쪽 레지스터 및 오른쪽 레지스터에 저장한 후, 제1 내지 제16 라운드를 통해 상기 제1 내지 제16 라운드 키를 이용한 암호화 연산을 통해 수행하는 단계; 및

상기 제1 내지 제16 라운드의 암호화 연산을 통해 생성된 비트들에 대해 상기 초기 치환의 역과정을 수행하는 단계를 포함하는 암호 처리 방법.

청구항 2

제 1 항에 있어서,

상기 제1 내지 제15 라운드 키 및 제16 라운드키를 생성하는 단계는

상기 암호화 키(cipher key)를 두 블록으로 나누어 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 하고 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 한 암호화 키(cipher key)를 1 비트 왼쪽 순환(left shift)을 하고 왼쪽 또는 오른쪽 레지스터에 오류를 주입한 후 압축치환을 거쳐 제16 라운드 키를 생성하는 것을 특징으로 하는 암호 처리 방법.

청구항 3

제 2 항에 있어서,

상기 제1 내지 제15 라운드 키 및 제16 라운드키를 생성하는 단계에서

제1, 제2, 제9 및 제16라운드에서는 1 비트 왼쪽 순환(left shift)을 수행하는 것을 특징으로 하는 암호 처리 방법.

청구항 4

제 1 항에 있어서,

상기 제1 내지 제15 라운드 키 및 제16 라운드 키는 48 비트인 것을 특징으로 하는 암호 처리 방법.

청구항 5

제 4 항에 있어서,

상기 제16 라운드 키는 왼쪽 또는 오른쪽 24비트에 오류가 주입된 값이고 나머지 24비트는 오류가 주입되지 않은 라운드 키값을 가지는 것을 특징으로 하는 암호 처리 방법.

청구항 6

제 1 항 내지 제 5 항 중 어느 한 항에 기재된 암호 처리 방법을 수행하기 위해 디지털 처리 장치에 의해 실행될 수 있는 명령어들의 프로그램이 구현되어 있으며, 상기 디지털 처리 장치에 의해 판독될 수 있는 프로그램이 기록된 기록매체.

청구항 7

데이터를 64비트로 나눈 블록을 16라운드 페이스텔(feistel)형 공통키 블록 암호 처리를 실행하는 암호 처리 장치에 있어서,

암호 처리의 개시 및 종료, 데이터의 송수신을 제어하고 프로그램을 실행하는 중앙처리부;

상기 중앙처리부가 실행하는 프로그램, 연산 파라미터로서의 고정 데이터를 저장하는 메모리; 및

마스터 키를 페리티 드롭하여 생성된 암호화 키(cipher key)를 비트 왼쪽 순환(left shift) 및 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 비트 왼쪽 순환, 레지스터의 오류 주입 및 압축치환을 통해 제16 라운드 키를 생성하여 상기 데이터를 64비트로 나눈 블록에 상기 생성된 제1 내지 제16 라운드 키를 이용하여 암호 처리를 수행하는 암호 처리부를 포함하는 암호 처리 장치.

청구항 8

제 7 항에 있어서,

상기 제1 내지 제15 라운드 키 및 제16 라운드 키는 48 비트인 것을 특징으로 하는 암호 처리 장치.

청구항 9

제 8 항에 있어서,

상기 제16 라운드 키는 왼쪽 또는 오른쪽 24비트에 오류가 주입된 값이고 나머지 24비트는 오류가 주입되지 않은 라운드 키값을 가지는 것을 특징으로 하는 암호 처리 장치.

청구항 10

제 7 항에 있어서,

상기 암호 처리부는

상기 암호화 키(cipher key)를 두 블록으로 나누어 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 하고 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 한 암호화 키(cipher key)를 1 비트 왼쪽 순환(left shift)을 하고 왼쪽 또는 오른쪽 레지스터에 오류를 주입한 후 압축치환을 거쳐 제16 라운드 키를 생성하는 것을 특징으로 하는 암호 처리 장치.

청구항 11

제 10 항에 있어서,

상기 암호 처리부는 제1 내지 제16 라운드 키를 생성하는 라운드 키 생성부를 포함하며,

상기 라운드 키 생성부는 64비트 마스터 키에서 페리티 드롭을 수행하여 56비트의 암호화 키를 생성하는 페리티 드롭부;

상기 56비트의 암호화 키를 28 비트의 두 블록으로 나누어 각 블록의 1비트 또는 2 비트 왼쪽 순환(left shift)을 수행하는 제1 쉬프트부 및 제2 쉬프트부; 및

상기 왼쪽 순환이 수행된 비트들의 압축 치환을 수행하는 압축 치환부를 포함하는 것을 특징으로 하는 암호 처리 장치.

청구항 12

제 11 항에 있어서,

상기 제1 쉬프트부 및 제2 쉬프트부는 제1, 제2, 제9 및 제16라운드에서는 1 비트 왼쪽 순환(left shift)을 수행하는 것을 특징으로 하는 암호 처리 장치.

청구항 13

제 10 항에 있어서,

상기 암호 처리부는

상기 데이터를 64비트로 나눈 블록에 대해 초기 치환을 수행하는 초기치환부;

마스터 키를 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 하고 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 한 암호화 키(cipher key)를 1 비트 왼쪽 순환(left shift)을 하고 왼쪽 또는 오른쪽 레지스터에 오류를 주입한 후 압축치환을 거쳐 제16 라운드 키를 생성하여 상기 초기 치환이 수행된 데이터에 상기 생성된 제1 내지 제16 라운드 키를 이용하여 암호 처리를 수행하는 기본연산부; 및

상기 제1 내지 제16 라운드를 통해 암호 처리가 수행된 비트들에 대해 상기 초기 치환의 역과정을 통해 암호문을 생성하는 역초기치환부를 포함하는 것을 특징으로 하는 암호 처리 장치.

청구항 14

제 13 항에 있어서,

상기 기본연산부는 상기 초기 치환이 수행된 데이터를 두 개의 32비트의 블록으로 나누어 왼쪽레지스터(L_i)와 오른쪽레지스터(R_i)에 저장한 후, 상기 생성된 제1 내지 제16 라운드 키를 이용하여 암호 처리를 수행하는 F-함수부를 포함하는 것을 특징으로 하는 암호 처리 장치.

청구항 15

제 14 항에 있어서,

상기 F-함수부는

상기 32비트의 블록을 48비트의 데이터로 확장 치환하는 확장 치환부;

상기 48비트의 데이터에 상기 라운드 키 생성부로부터 생성된 상기 제1 내지 제16 라운드 키의 배타적 논리합 연산을 수행하는 배타적 논리합 연산부;

상기 배타적 논리합 연산이 수행된 48비트의 데이터를 32비트의 데이터로 대치 치환하는 S-box 치환부; 및

상기 대치 치환된 32비트의 데이터를 복사 치환하는 P-box 치환부를 포함하는 것을 특징으로 하는 암호 처리 장치.

청구항 16

제 15 항에 있어서,

상기 S-box 치환부는 8개의 S-box를 포함하고,

상기 S-box의 입력값 중 1 내지 4번째 또는 5 내지 8번째 S-box에 오류가 주입된 것을 특징으로 하는 암호 처리 장치.

청구항 17

제 16 항에 있어서,

상기 S-box의 출력값 중 1 내지 16번째 또는 17 내지 32번째 비트에 오류가 주입된 것을 특징으로 하는 암호 처리 장치.

청구항 18

제 14 항에 있어서,

상기 F-함수부의 출력값의 1, 2, 6, 9, 10, 13, 16, 17, 18, 20, 23, 24, 26, 28, 30, 32번째 또는 3, 4, 5, 7, 8, 11, 12, 14, 15, 19, 21, 22, 25, 27, 32번째 비트에 오류가 주입된 것을 특징으로 하는 암호 처리 장치.

명세서

기술분야

- [0001] 본 발명은 암호 처리 방법 및 암호 처리 장치에 관한 것으로, 특히 데이터 암호화 표준의 라운드 키 생성과정에서 차분오류주입공격을 이용한 암호 처리 방법 및 암호 처리 장치에 관한 것이다.

배경기술

- [0002] 네트워크 통신, 전자 상거래의 발전에 따라 시큐리티 확보가 점점 중요해지고 있다. 시큐리티 확보의 방법 중 하나가 암호 기술이며, 다양한 암호화 방법을 이용한 통신이 이루어지고 있다.
- [0003] 암호화 방법은 크게 공개키 암호 방식 및 공통키 암호 방식으로 분류된다. 공개키 암호 방식은 암호화키와 복호키를 상이한 키로 설정하는 암호 방식이고 공통키 암호 방식은 암호화키와 복호키를 공통의 키로 설정하는 암호 방식이다. 공통키 암호 방식에도 다양한 알고리즘이 있지만, 그 중 하나는 공통키를 베이스로 하여 복수개의 키를 생성하고 생성한 복수개의 키를 사용하여 블록 단위(64비트, 128비트 등)의 데이터 변환 처리를 반복 실행하는 방식으로 대표적인 것이 공통키 블록 암호 방식이다. 공통키 블록 암호 방식의 대표적인 것으로 미국 표준 암호인 데이터 암호화 표준(DES, Data Encryption Standard)이 다양한 분야에서 널리 이용되고 있다.
- [0004] 암호 알고리즘에 대한 다양한 부채널 공격방법이 소개되었는데, TA(시간공격, Timing Attack), SPA(단순전력분석, Simple Analysis), DPA(차분전력분석, Differential Power Analysis), EMA(전자파 분석, Electromagnetic Analysis)등의 수동적인 공격방법과 FA(오류주입공격, Fault Analysis) 및 DFA(차분오류주입공격, Differential Fault Analysis)등의 능동적인 공격방법에 대한 연구가 이루어지고 있다. 이 중 차분오류주입공격(DFA)은 블록 암호 알고리즘 또는 스트림 암호 알고리즘의 데이터 흐름에서 특정한 부분에 오류를 주입하여 얻은 옳은 암호문과 오류를 주입한 암호문의 차분을 이용하여 마스터키를 구하는 방법이다.
- [0005] 데이터 암호화 표준(DES)에 대한 차분오류주입공격(DFA)은 13라운드 왼쪽 스테이트에 오류를 주입하여 마스터키를 복구하는 것으로 제안되었었다. 그 후 이를 일반화하고 확장하여 중간라운드(9-12라운드)의 왼쪽 스테이트에 오류를 주입하여 마스터 키를 얻을 수 있는 방법이 제안되었는데 이는 암호화 과정과 복호화 과정에서 전 라운드에 걸쳐 차분오류주입공격(DFA)방법을 적용할 수 있음을 의미한다. 또한, 일반적인 암호화 과정에서의 차분오류주입공격(DFA)이 아닌 키 생성과정에서의 차분오류주입공격(DFA)이 제안되기도 하였다. 그러나, 다양하게 제안된 이러한 방법들은 주입되는 오류의 갯수가 지나치게 커서 복잡도가 증가하는 문제점이 있다.

[0006]

발명의 내용

해결하려는 과제

- [0007] 본 발명이 이루고자 하는 기술적 과제는 데이터 암호화 표준(DES, Data Encryption Standard)의 문제점을 해결하기 위한 것으로, 기존의 데이터 암호화 표준(DES)의 문제점을 보완하여 데이터 암호화 표준(DES)의 키 생성과정에 오류를 주입하여 오류의 갯수가 향상된 암호 처리 방법 및 암호 처리 장치를 제공하는 데 있다.

과제의 해결 수단

- [0008] 본 발명의 일 양태에 따르면 데이터를 64비트로 나눈 블록을 16라운드 페이스텔(feistel)형 공통키 블록 암호 처리를 실행하는 암호 처리 방법에 있어서, 64비트의 평문에 초기 치환을 수행하는 단계, 마스터 키를 페리티드롭하여 생성된 암호화 키(cipher key)를 비트 왼쪽 순환(left shift) 및 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 비트 왼쪽 순환, 레지스터의 오류 주입 및 압축치환을 통해 제16 라운드 키를 생성하는 단계, 상기 초기 치환이 수행된 64비트의 평문을 두 개의 32비트로 나누어 왼쪽 레지스터 및 오른쪽 레지스터에 저장한 후, 제1 내지 제16 라운드를 통해 상기 제1 내지 제16 라운드 키를 이용한 암호화 연산을 통해 수행하는 단계, 및 상기 제1 내지 제16 라운드의 암호화 연산을 통해 생성된 비트들에 대해 상기 초기 치환의 역과정을 수행하는 단계를 포함하는 암호 처리 방법을 제공한다.

- [0009] 상기 제1 내지 제15 라운드 키 및 제16 라운드키를 생성하는 단계는 상기 암호화 키(cipher key)를 두 블록으로 나누어 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 하고 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 한 암호화 키(cipher key)를 1 비트 왼쪽 순환(left shift)을 하고 왼쪽 또는 오른쪽 레지스터에 오류를 주입한 후 압축치환을 거쳐 제16 라운드 키를 생성할 수 있다.
- [0010] 상기 제1 내지 제15 라운드 키 및 제16 라운드키를 생성하는 단계에서 제1, 제2, 제9 및 제16라운드에서는 1 비트 왼쪽 순환(left shift)을 수행할 수 있다.
- [0011] 상기 제1 내지 제15 라운드 키 및 제16 라운드 키는 48 비트일 수 있다.
- [0012] 상기 제16 라운드 키는 왼쪽 또는 오른쪽 24비트에 오류가 주입된 값이고 나머지 24비트는 오류가 주입되지 않은 라운드 키값을 가질 수 있다.
- [0013] 본 발명의 다른 양태에 따르면 상기 암호 처리 방법을 수행하기 위해 디지털 처리 장치에 의해 실행될 수 있는 명령어들의 프로그램이 구현되어 있으며, 상기 디지털 처리 장치에 의해 판독될 수 있는 프로그램이 기록된 기록매체를 제공한다.
- [0014] 본 발명의 또 다른 양태에 따르면 데이터를 64비트로 나눈 블록을 16라운드 페이스텔(feistel)형 공통키 블록 암호 처리를 실행하는 암호 처리 장치에 있어서, 암호 처리의 개시 및 종료, 데이터의 송수신을 제어하고 프로그램을 실행하는 중앙처리부, 상기 중앙처리부가 실행하는 프로그램, 연산 파라미터로서의 고정 데이터를 저장하는 메모리, 및 마스터 키를 패리티 드롭하여 생성된 암호화 키(cipher key)를 비트 왼쪽 순환(left shift) 및 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 비트 왼쪽 순환, 레지스터의 오류 주입 및 압축치환을 통해 제16 라운드 키를 생성하여 상기 데이터를 64비트로 나눈 블록에 상기 생성된 제1 내지 제16 라운드 키를 이용하여 암호 처리를 수행하는 암호 처리부를 포함하는 암호 처리 장치를 제공한다.
- [0015] 상기 암호 처리부는 상기 암호화 키(cipher key)를 두 블록으로 나누어 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 하고 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 한 암호화 키(cipher key)를 1 비트 왼쪽 순환(left shift)을 하고 왼쪽 또는 오른쪽 레지스터에 오류를 주입한 후 압축치환을 거쳐 제16 라운드 키를 생성할 수 있다.
- [0016] 상기 암호 처리부는 제1 내지 제16 라운드 키를 생성하는 라운드 키 생성부를 포함하며, 상기 라운드 키 생성부는 64비트 마스터 키에서 패리티 드롭을 수행하여 56비트의 암호화 키를 생성하는 패리티 드롭부, 상기 56비트의 암호화 키를 28 비트의 두 블록으로 나누어 각 블록의 1비트 또는 2 비트 왼쪽 순환(left shift)을 수행하는 제1 쉬프트부 및 제2 쉬프트부, 및 상기 왼쪽 순환이 수행된 비트들의 압축 치환을 수행하는 압축 치환부를 포함할 수 있다.
- [0017] 상기 제1 쉬프트부 및 제2 쉬프트부는 제1, 제2, 제9 및 제16라운드에서는 1 비트 왼쪽 순환(left shift)을 수행할 수 있다.
- [0018] 상기 암호 처리부는 상기 데이터를 64비트로 나눈 블록에 대해 초기 치환을 수행하는 초기치환부, 마스터 키를 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 하고 압축치환(compression permutation)을 거쳐 제1 내지 제15 라운드 키를 생성하고 상기 제1 내지 제15 라운드를 통해 1 비트 또는 2 비트 왼쪽 순환(left shift)을 한 암호화 키(cipher key)를 1 비트 왼쪽 순환(left shift)을 하고 왼쪽 또는 오른쪽 레지스터에 오류를 주입한 후 압축치환을 거쳐 제16 라운드 키를 생성하여 상기 초기 치환이 수행된 데이터에 상기 생성된 제1 내지 제16 라운드 키를 이용하여 암호 처리를 수행하는 기본연산부, 및 상기 제1 내지 제16 라운드를 통해 암호 처리가 수행된 비트들에 대해 상기 초기 치환의 역과정을 통해 암호문을 생성하는 역초기치환부를 포함할 수 있다.
- [0019] 상기 기본연산부는 상기 초기 치환이 수행된 데이터를 두 개의 32비트의 블록으로 나누어 왼쪽레지스터(L_i)와 오른쪽레지스터(R_i)에 저장한 후, 상기 생성된 제1 내지 제16 라운드 키를 이용하여 암호 처리를 수행하는 F-함수부를 포함할 수 있다.
- [0020] 상기 F-함수부는 상기 32비트의 블록을 48비트의 데이터로 확장 치환하는 확장 치환부, 상기 48비트의 데이터에

상기 라운드 키 생성부로부터 생성된 상기 제1 내지 제16 라운드 키의 배타적 논리합 연산을 수행하는 배타적 논리합 연산부, 상기 배타적 논리합 연산이 수행된 48비트의 데이터를 32비트의 데이터로 대치 치환하는 S-box 치환부, 및 상기 대치 치환된 32비트의 데이터를 복사 치환하는 P-box 치환부를 포함할 수 있다.

[0021] 상기 S-box 치환부는 8개의 S-box를 포함하고, 상기 S-box의 입력값 중 1 내지 4번째 또는 5 내지 8번째 S-box에 오류가 주입될 수 있다.

[0022] 상기 S-box의 출력값 중 1 내지 16번째 또는 17 내지 32번째 비트에 오류가 주입될 수 있다.

[0023] 상기 F-함수부의 출력값의 1, 2, 6, 9, 10, 13, 16, 17, 18, 20, 23, 24, 26, 28, 30, 32번째 또는 3, 4, 5, 7, 8, 11, 12, 14, 15, 19, 21, 22, 25, 27, 32번째 비트에 오류가 주입될 수 있다.

발명의 효과

[0024] 본 실시예에 따른 암호 처리 방법은 16라운드 키 생성과정에서 오류를 주입하여 오류 주입 가능 범위를 넓히고 적은 오류를 주입하여 마스터 키를 획득할 수 있다.

도면의 간단한 설명

[0025] 도 1은 데이터 암호화 표준(DES, Data Encryption Standard)의 암호화 모듈을 설명하기 위한 블록도이다.

도 2는 F-함수부의 구성을 나타내는 블록도이다.

도 3은 본 실시예에 따른 라운드 키 생성부를 나타내는 블록도이다.

도 4는 본 실시예에 따른 오류가 주입된 16라운드 키를 통한 암호 처리 방법을 설명하기 위한 개념도이다.

도 5는 본 실시예에 따른 오류가 주입된 16라운드 키를 통한 암호 처리를 통해 마스터 키를 획득하는 방법을 설명하는 순서도이다.

도 6은 본 실시예에 따라 암호 처리를 수행하는 암호 처리 장치를 나타내는 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0026] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.

[0027] 제2, 제1 등과 같이 서수를 포함하는 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 상기 구성요소들은 상기 용어들에 의해 한정되지는 않는다. 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제2 구성요소는 제1 구성요소로 명명될 수 있고, 유사하게 제1 구성요소도 제2 구성요소로 명명될 수 있다. 및/또는 이라는 용어는 복수의 관련된 기재된 항목들의 조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.

[0028] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.

[0029] 본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0030] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가지고 있다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥 상 가지는 의미와 일치하는 의미를 가지는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

- [0031] 이하, 첨부된 도면을 참조하여 실시예를 상세히 설명하되, 도면 부호에 관계없이 동일하거나 대응하는 구성 요소는 동일한 참조 번호를 부여하고 이에 대한 중복되는 설명은 생략하기로 한다.
- [0032] 본 실시예에서, $a \oplus b$ 는 a 와 b 의 배타적 논리합, $\langle a \rangle$ 는 a 의 집합, P 는 평문(64비트), C 는 옴은 암호문(64비트), C_L^* 는 16라운드 키 생성단계에서 왼쪽 레지스터에 오류가 주입된 암호문, C_R^* 는 16라운드 키 생성단계에서 오른쪽 레지스터에 오류가 주입된 암호문, L_i 는 i 번째 라운드의 왼쪽 32비트 출력값, R_i 는 i 번째 라운드의 오른쪽 32비트 출력값, L_i^* 는 오류가 주입된 i 번째 라운드의 왼쪽 32비트 출력값, R_i^* 는 i 번째 라운드의 오른쪽 32비트 출력값, K 는 마스터 키(64비트) 및 RK_i 는 i 번째 라운드 키(48비트)를 나타낸다.
- [0033] 도 1은 데이터 암호화 표준(DES, Data Encryption Standard)의 암호화 모듈을 설명하기 위한 블록도이다.
- [0034] 도 1을 참조하면, 암호화 모듈(10)은 초기치환부(100), 기본연산부(200) 및 역초기치환부(300)를 포함한다. 초기치환부(100)는 입력되는 64비트의 평문(P , plain Text)블럭에 대해 초기 치환(IP, Initial permutation)을 수행한다. 기본연산부(200)는 초기 치환이 수행된 64비트의 블럭을 두 개의 32비트의 블럭으로 나누어 왼쪽 레지스터(L_i)와 오른쪽 레지스터(R_i)에 저장한 후, F-함수로 수행되는 16라운드의 곱 변형(Product Transformation)과 왼쪽 레지스터(L_i)와 오른쪽 레지스터(R_i)를 매 라운드마다 교환하는 16라운드의 블럭 변형(Block Transformation)을 수행한다. 여기서, 곱 변형은 오른쪽 레지스터(R_i)에 저장된 32비트의 블럭을 라운드 키 생성부(도시되지 않음)에 의해서 생성된 라운드 키(RK_i)와 함께 입력시켜서 암호화 연산을 수행하는 F-함수부(210) 및 F-함수부(210)의 결과를 왼쪽 레지스터(L_i)에 저장된 32비트의 블럭과 함께 배타적 논리합하는 제1 배타적 논리합 연산부(220)로 구성되어 있다. 제1 배타적 논리합 연산부(220)의 32비트의 데이터는 다음 라운드의 오른쪽 레지스터(R_{i+1})에 저장되고 오른쪽 레지스터(R_i)에 저장된 32비트의 데이터는 다음 라운드의 왼쪽 레지스터(L_{i+1})에 교환되어 저장됨으로 블럭 변형이 수행된다. 암호화 모듈(10)에서는 이러한 1라운드의 연산이 반복되어 16라운드가 수행된다. 역초기치환부(300)는 16라운드에 걸친 변형이 종료된 후 생성된 비트들을 역초기치환(IP^{-1})을 거쳐서 암호화된 암호문(C , Cipher Text)를 출력한다.
- [0035] 도 2는 F-함수부의 구성을 나타내는 블록도이다.
- [0036] 도 2를 참조하면, F-함수부(210)는 F-함수 연산을 수행하며, 확장 치환부(211), 제2 배타적 논리합 연산부(212), S-box 치환부(213) 및 P-box 치환부(214)를 포함한다. 확장 치환부(211)는 32비트의 텍스트 블럭을 저장하고 있는 레지스터(R_B)로부터 32비트의 데이터를 입력받아 48비트의 데이터로 확장 치환한다. 제2 배타적 논리합 연산부(212)는 확장 치환부(211)로부터 출력된 48비트의 데이터를 입력받고 라운드 키 생성부(도시되지 않음)로부터 라운드 키(RK_i)를 입력받아 배타적 논리합 연산을 수행한다.
- [0037] S-box 치환부(213)는 8개의 S-box를 포함하고, 제2 배타적 논리합 연산부(212)로부터 출력된 48비트의 데이터를 32비트의 데이터로 대치 치환한다. P-box 치환부(214)는 S-box 치환부(213)로부터 출력된 32비트의 데이터를 복사 치환하여 32비트의 데이터를 출력한다.
- [0038] 도 3은 본 실시예에 따른 라운드 키 생성부를 나타내는 블록도이다.
- [0039] 도 3을 참조하면, 데이터 암호화 표준(DES, Data Encryption Standard)의 마스터 키는 56비트의 키와 패리티 체크 비트인 8비트를 포함하여 64비트이다. 마스터 키 64비트를 패리티 드롭부(20)에서 패리티 드롭하여 56비트의 암호화 키를 생성하여 출력한다. 56비트 암호화 키는 28비트의 두 블럭로 나누어 제1 쉬프트부(31) 및 제2 쉬프트부(33)에 의해 1비트(1, 2, 9, 16라운드) 또는 2비트 왼쪽 순환(left shift)을 하여 압축 치환부(35)에서 압축 치환과정을 거쳐 48비트의 라운드 키(RK_i)를 생성하여 출력한다.
- [0040] 이러한 본 실시예에 따른 라운드 키 생성부에서는 16라운드의 압축 치환부(35)에 입력값이 저장되어 있는 28비트의 왼쪽 레지스터 또는 오른쪽 레지스터 중 하나의 레지스터에 오류를 주입한다. 압축 치환부(35)에서 압축 치환과정을 거쳐 생성되어 출력된 16라운드 키(RK_{16})는 왼쪽 또는 오른쪽 절반 24비트에 오류가 주입된 값이고 나머지 절반 24비트는 오류가 주입되지 않은 본래의 라운드 키의 값을 가진다. 따라서, 본 실시예에 따른 16라

라운드 키(RK₁₆)가 입력된 F-함수부(210)의 8개의 S-box 중 1 내지 4번째 또는 5 내지 8번째 S-box에 오류가 주입된 값이 입력된다. S-box의 출력값 중 1 내지 16번째 또는 17 내지 32번째 비트에 오류가 주입되고 F-함수부(210)의 P-box 치환부(214)에 의한 F-함수부(210)의 출력값의 1, 2, 6, 9, 10, 13, 16, 17, 18, 20, 23, 24, 26, 28, 30, 32번째 또는 3, 4, 5, 7, 8, 11, 12, 14, 15, 19, 21, 22, 25, 27, 32번째 비트에 오류가 주입될 수 있다. 여기서, 라운드 키 생성부에서 i번째 라운드는 데이터 암호화 표준(DES, Data Encryption Standard)의 암호화 모듈의 i번째 라운드에 이용되는 라운드 키를 생성하는 라운드를 의미한다.

[0041] 도 4는 본 실시예에 따른 오류가 주입된 16라운드 키를 통한 암호 처리 방법을 설명하기 위한 개념도이다.

[0042] 도 4를 참조하면, R₁₆, L₁₆, L₁₆^{*}은 공격자가 알고 있는 값이므로 F-함수부(210)의 출력 차분과 입력값을 알 수 있다. S-box의 출력 차분 정보를 이용하여 S-box 입력값의 후보군을 생성할 수 있다. 하나의 오류에 여러개의 평문에 대한 암호문으로 후보군의 개수를 줄여 라운드 키의 1 내지 24 비트값 또는 25 내지 48 비트값을 결정할 수 있다.

[0043] 도 5는 본 실시예에 따른 오류가 주입된 16라운드 키를 통한 암호 처리를 통해 마스터 키를 획득하는 방법을 설명하는 순서도이다.

[0044] 도 5를 참조하면, 옳은 암호문과 본 실시예에 따라 오류가 주입된 16라운드 키를 통해 오류가 주입된 암호문을 획득한다(S510). 구체적으로, 임의의 평문(P)에 대하여 오류가 주입되지 않은 암호문(C)을 얻는다. 본 실시예에 따른 라운드 키 생성부에서는 16라운드의 압축 치환부(35)에 입력값이 저장되어 있는 28 비트의 왼쪽 레지스터 또는 오른쪽 레지스터 중 하나의 레지스터에 오류를 주입하여 얻은 암호문(C_L^{*}) 또는 암호문(C_R^{*})을 얻는다.

[0045] 그런 다음, S-box 출력값 후보군의 집합을 생성한다(S520). S510에서 얻은 옳은 암호문(C)과 오류가 주입된 암호문(C_L^{*} 또는 C_R^{*})에 초기 치환과정을 거쳐서 16라운드 출력값 L₁₆, L₁₆^{*}, R₁₆=R₁₆^{*}를 얻는다. F-함수부(210)의 출력 차분은 다음 수학적 식 1과 같다.

수학적 식 1

$$\Delta L_{16} = L_{16} \oplus L_{16}^*$$

[0046]

[0047] F-함수부(210)에서 마지막 치환의 역치환 과정을 거쳐서 S-box의 출력차분을 획득한다. C_L^{*}에 대한 S-box의 출력 차분은 1 내지 4번째 S-box에만 영향을 미치므로 ****0000이고, C_R^{*}에 대한 S-box의 출력 차분은 5 내지 8번째 S-box에만 영향을 미치므로 0000****이다. 여기서, *는 알고 있는 S-box의 출력 차분값이며, 0 내지 15인 정수값이다. 각각의 S-box 출력 차분에 대하여 옳은 출력값이 결정되면 오류가 주입된 출력값이 하나로 결정되므로 가능한 옳은 출력값과 오류가 주입된 출력값의 쌍은 16개이다. 8개의 S-box에 대한 각각의 16개의 출력값 쌍을 집합<output(ith S-box)>에 저장한다. 여기서, i는 1 내지 8인 정수값이다.

[0048] 그런 다음, S-box 입력값 후보군의 집합을 생성한다(S530). 구체적으로, F-함수부(210)의 S-box는 정해진 하나의 출력값에 대하여 4개의 입력값을 가진다. 따라서, 하나의 옳은 출력값과 오류가 주입된 출력값은 4개의 입력값 후보군을 갖는다. S520에서 얻은 S-box의 출력값 집합의 옳은 출력값과 오류가 주입된 출력값의 쌍은 16개의 입력값 후보군 쌍을 갖는다. 하나의 S-box에 대한 옳은 입력값과 오류가 주입된 입력값 후보군 쌍의 개수는 256개(16×16)이다. 8개의 S-box에 대한 각각의 256개 입력값 쌍을 집합<input(ith S-box)>에 저장한다. 여기서, i는 1 내지 8인 정수값이다.

[0049] 그런 다음, 16라운드 키 후보군의 집합을 생성한다(S540). S530에서 얻은 R16으로부터 F-함수부(210)의 확장 치환부(211)를 거쳐 확장된 48비트 중간값을 얻는다. 48비트의 중간값을 8개의 6비트로 나누어 S530에서 저장한 집합<input(ith S-box)> 원소들과 배타적 논리합 연산을 수행한다. 연산결과 RK₁₆와 RK₁₆^{*}의 후보군 6비트 쌍을 8개의 집합<ith RK₁₆, ith RK₁₆^{*}>에 저장한다. 여기서, i는 1 내지 8인 정수값이다.

- [0050] 그런 다음, 마스터 키를 획득한다(S550). 구체적으로, 다른 평문에 대하여 S510 내지 S540를 반복하여(S540) 저장한 각각의 집합 $\{i\text{th } RK_{16}, i\text{th } RK_{16}^*\}$ 에 대하여 일치하지 않는 라운드 키 쌍을 제거한다. 이러한 과정을 반복하여 하나의 48비트 라운드 키를 구한다. 확정된 16라운드 키를 이용하여 라운드 키 생성단계의 역과정을 통해 마스터 키를 획득한다.
- [0051] S510 내지 S560단계를 상세하여 설명하면, S-box 입력값 집합의 원소의 개수는 256개(2^8)이다. S-box의 입력값의 전체 가능한 개수는 $2^{12}(=2^6 \times 2^6)$ 이다. 하나의 평문에 오류를 주입함으로써 S-box 입력값 쌍의 후보군 개수는 $1/2^4(=2^8/2^{12})$ 만큼 줄어든다. 따라서, 3개의 평문에 대하여 오류를 주입하면 $1/256(=(1/2^4)^3)$ 으로 S-box 입력값 쌍의 후보군 개수를 줄일 수 있으므로 하나의 옳은 입력값 쌍을 구할 수 있다. 즉, 라운드 키 생성단계에서 16라운드 압축치환의 왼쪽 레지스터 또는 오른쪽 레지스터에서 오류를 주입하여 각각 3개의 평문에 대한 암호문을 얻으면 옳은 16라운드 키를 얻을 수 있고 확정된 16라운드 키를 이용하여 라운드 키 생성단계의 역과정을 통해 마스터 키를 획득할 수 있다.
- [0052] 도 6은 본 실시예에 따라 암호 처리를 수행하는 암호 처리 장치를 나타내는 블록도이다.
- [0053] 도 6을 참조하면, 암호 처리 장치(600)는 중앙처리부(610), 메모리(620), 암호처리부(630), 난수발생기(640) 및 송수신부(650)를 포함한다.
- [0054] 중앙처리부(610)(CPU, Central processing Unit)는 암호 처리의 개시나, 종료, 데이터의 송수신의 제어, 각 구성부 사이의 데이터 전송 제어, 그 외의 각종 프로그램을 실행하는 프로세서이다. 메모리(620)는 중앙처리부(610)가 실행하는 프로그램, 또는 연산 파라미터로서의 고정 데이터를 저장하는 ROM(Read-Only-Memory), 중앙처리부(610)의 처리에 있어서 실행되는 프로그램 및 프로그램 처리에 있어서 적당히 변화하는 파라미터의 저장 영역, 동작물 영역으로서 사용되는 RAM(Random Access Memory) 등을 포함하여 구성된다. 또한, 메모리(620)는 암호 처리에 필요한 키 데이터 등의 저장 영역으로서 이용할 수 있다.
- [0055] 암호 처리부(630)는 전술한 본 실시예에 따른 방법을 수행하는 암호 처리 및 복호 처리 등을 실행한다. 난수 발생기(640)는 암호 처리에 필요한 키의 생성 등에 있어서 필요한 난수의 발생 처리를 실행한다. 송수신부(650)는 외부와의 데이터 통신을 실행하는 데이터 통신 처리부이다. 예를 들어, 리더 라이터 등, IC모듈과의 데이터 통신을 실행하고, IC모듈 내에서 생성한 암호문의 출력 또는 외부의 리더 라이터 등의 기기로부터의 데이터 입력 등을 실행한다.
- [0056] 본 실시예에서는 암호 처리 장치(600)를 개별 모듈로 한 예를 나타냈으나, 이와 같이 독립된 암호 처리 모듈로 구성되지 않고 암호 처리 프로그램을 ROM에 저장하고 중앙처리부(610)가 ROM 저장 프로그램을 판독하여 실행하도록 구성할 수도 있다.
- [0057] 이하에서는, 본 실시예에 따른 암호 처리 방법의 효과에 대하여 설명하기로 한다.
- [0058] 본 실시예에 따른 암호 처리 방법 및 종래 DES의 암호 처리 방법의 비교를 다음 표 1에서 설명한다.

표 1

오류주입 위치	오류주입 모델	오류 갯수
12라운드 ([10])	1바이트 랜덤 오류	약 20개
11라운드 ([10])	1바이트 랜덤 오류	약 2700개
10라운드 ([10])	1바이트 랜덤 오류	약 23400개
9라운드 ([10])	1바이트 랜덤 오류	10^8 개 이상
제안하는 방법(16라운드)	레지스터(4바이트) 랜덤 오류	약 6개

[0059]

- [0060] 표 1을 참조하면, DES의 중간 라운드에 대해 차분오류주입공격을 수행한 방법과 본 실시예에 따라 16라운드에 대해 차분오류주입공격을 수행한 방법을 비교하였다. 중간 라운드에 대해 차분오류주입공격을 수행한 방법에서는 9 내지 12라운드에 약 20 내지 10^8 개의 1 바이트 랜덤 오류를 주입하여 마스터 키를 획득한다. 반면, 본 실시예에 따라 16라운드에 대해 차분오류주입공격을 수행한 방법은 16라운드 키 생성단계에 약 6개의 레지스터(4 바이트)랜덤 오류를 주입하여 마스터 키를 획득한다. 주입되는 오류의 크기 및 오류의 갯수가 종래 방법에 비해

향상된 것을 확인할 수 있다.

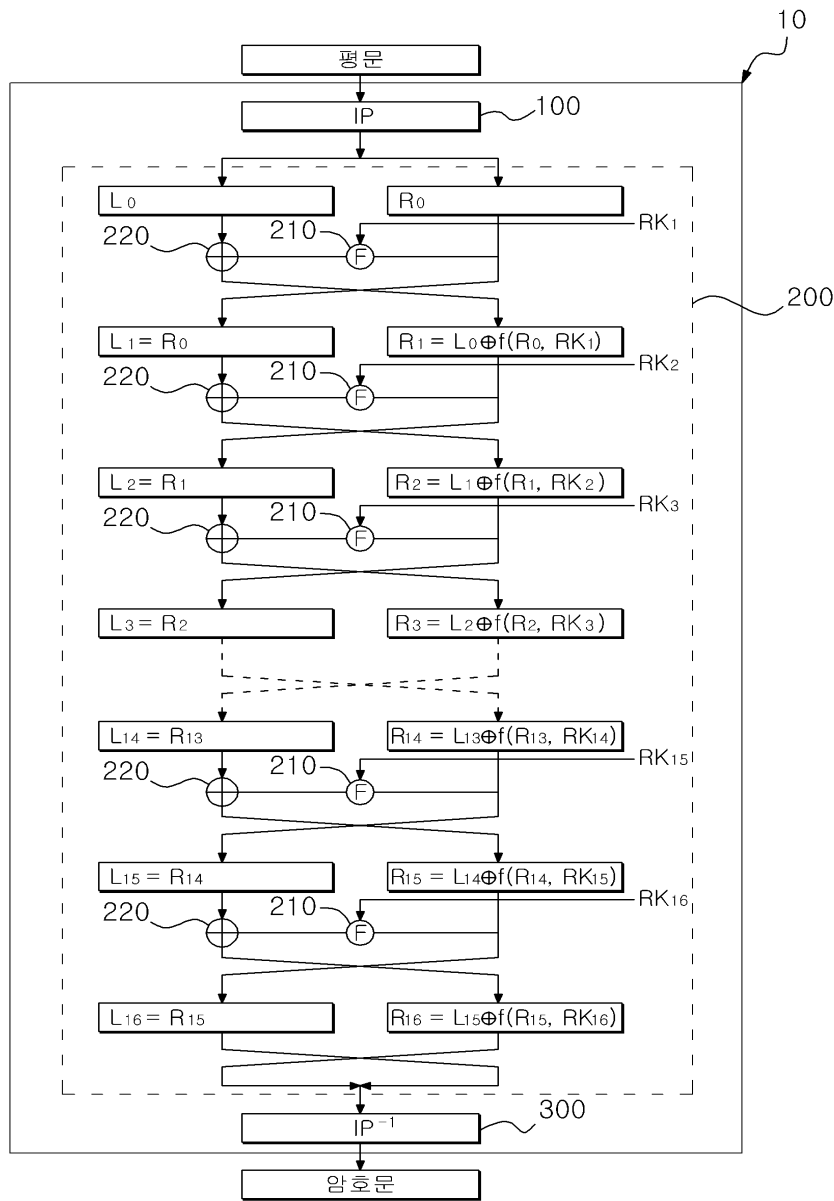
- [0061] 본 실시예에서 사용되는 '~부'라는 용어는 소프트웨어 또는 FPGA(field-programmable gate array) 또는 ASIC과 같은 하드웨어 구성요소를 의미하며, '~부'는 어떤 역할들을 수행한다. 그렇지만 '~부'는 소프트웨어 또는 하드웨어에 한정되는 의미는 아니다. '~부'는 어드레싱할 수 있는 저장 매체에 있도록 구성될 수도 있고 하나 또는 그 이상의 프로세서들을 재생시키도록 구성될 수도 있다. 따라서, 일 예로서 '~부'는 소프트웨어 구성요소들, 객체지향 소프트웨어 구성요소들, 클래스 구성요소들 및 태스크 구성요소들과 같은 구성요소들과, 프로세스들, 함수들, 속성들, 프로시저들, 서브루틴들, 프로그램 코드의 세그먼트들, 드라이버들, 펌웨어, 마이크로코드, 회로, 데이터, 데이터베이스, 데이터 구조들, 테이블들, 어레이들, 및 변수들을 포함한다. 구성요소들과 '~부'들 안에서 제공되는 기능은 더 작은 수의 구성요소들 및 '~부'들로 결합되거나 추가적인 구성요소들과 '~부'들로 더 분리될 수 있다. 뿐만 아니라, 구성요소들 및 '~부'들은 디바이스 또는 보안 멀티미디어카드 내의 하나 또는 그 이상의 CPU들을 재생시키도록 구현될 수도 있다.
- [0062] 이상에서 본 발명의 실시예에 대하여 상세하게 설명하였지만 본 발명의 권리범위는 이에 한정되는 것은 아니고 다음의 청구범위에서 정의하고 있는 본 발명의 기본 개념을 이용한 당업자의 여러 변형 및 개량 형태 또한 본 발명의 권리범위에 속하는 것이다.
- [0063] 상기 프로그램은 컴퓨터가 읽을 수 있는 정보저장매체(computer reader media)에 저장되고, 컴퓨터에 의하여 읽혀지고 실행됨으로써 암호처리방법을 구현한다. 상기 정보저장매체는 자기 기록매체, 광기록매체 및 캐리어 웨이브 매체를 포함한다.
- [0064] 상술한 모든 기능은 상기 기능을 수행하도록 코딩된 소프트웨어나 프로그램 코드 등에 따른 마이크로프로세서, 제어기, 마이크로제어기, ASIC(Application Specific Integrated Circuit) 등과 같은 프로세서에 의해 수행될 수 있다. 상기 코드의 설계, 개발 및 구현은 본 발명의 설명에 기초하여 당업자에게 자명하다고 할 것이다.
- [0065] 이상 본 발명에 대하여 실시예를 참조하여 설명하였지만, 해당 기술 분야의 통상의 지식을 가진 자는 본 발명의 기술적 사상 및 영역으로부터 벗어나지 않는 범위 내에서 본 발명을 다양하게 수정 및 변경시켜 실시할 수 있음을 이해할 수 있을 것이다. 따라서 상술한 실시예에 한정되지 않고, 본 발명은 이하의 특허청구범위의 범위 내의 모든 실시예들을 포함한다고 할 것이다.

부호의 설명

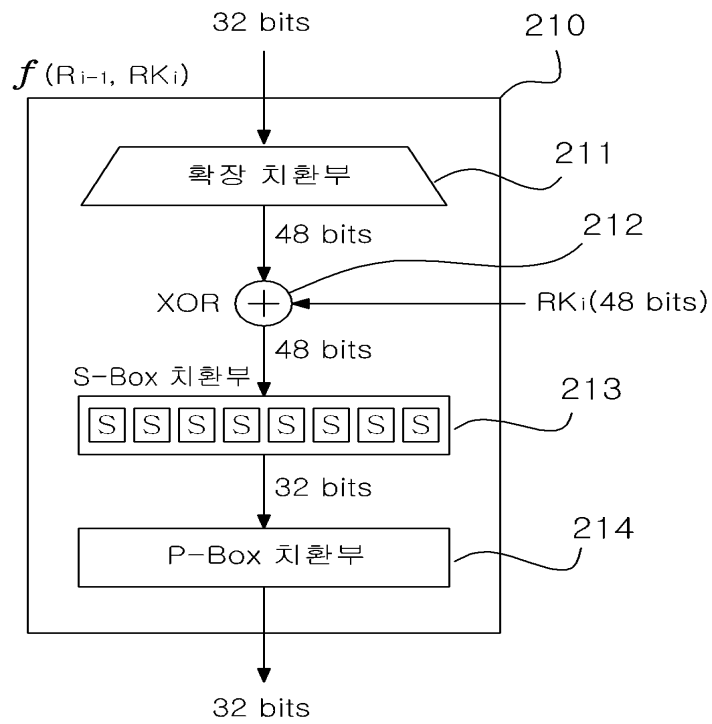
- [0066]
- | | |
|---------------------|---------------------|
| 10: 암호화 모듈 | 100: 초기치환부 |
| 200: 기본연산부 | 210: F-함수부 |
| 211: 확장 치환부 | 212: 제2 배타적 논리합 연산부 |
| 213: S-box 치환부 | 214: P-box 치환부 |
| 220: 제1 배타적 논리합 연산부 | 300: 역초기치환부 |

도면

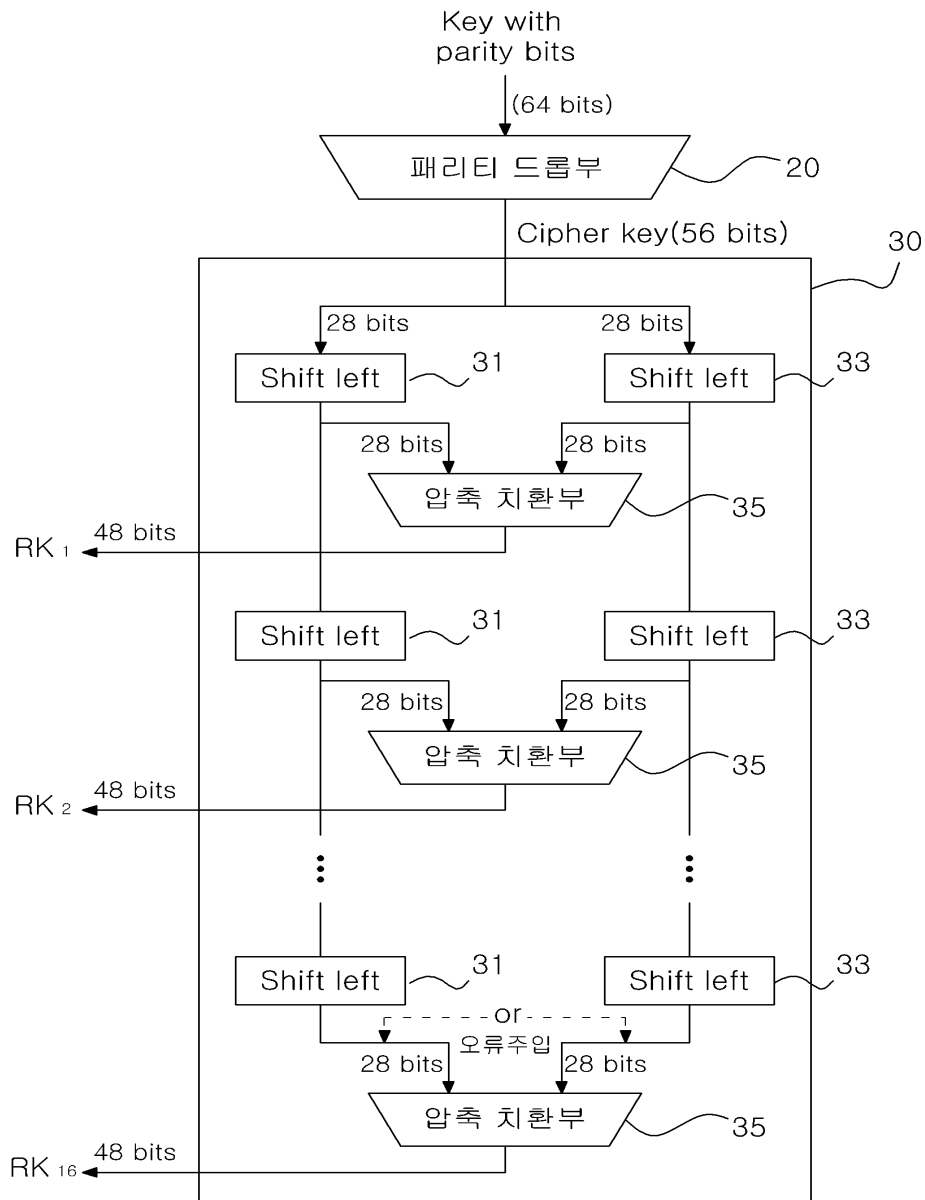
도면1



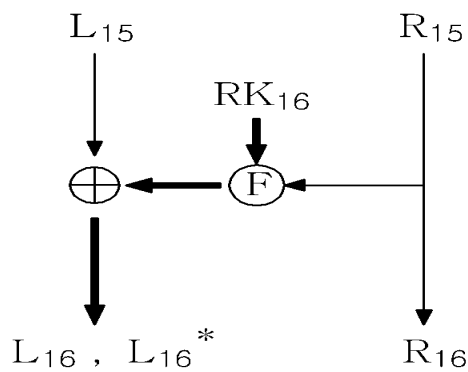
도면2



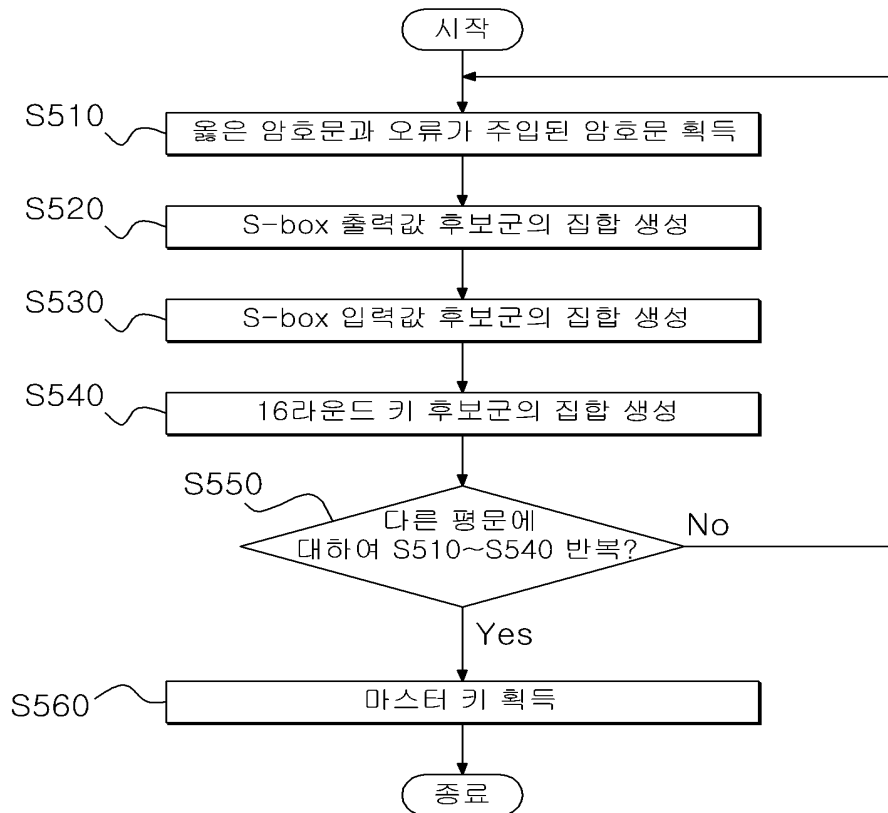
도면3



도면4



도면5



도면6

