(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2011년12월08일
(11) 등록번호 10-1091749
(24) 등록일자 2011년12월02일

(51) Int. Cl.
G09C 1/00 (2006.01)
(21) 출원번호 10-2006-7006887
(22) 출원일자(국제출원일자) 2005년08월30일
심사청구일자 2010년07월16일
(85) 번역문제출일자 2006년04월10일
(65) 공개번호 10-2007-0058370
(43) 공개일자 2007년06월08일
(86) 국제출원번호 PCT/JP2005/015815
(87) 국제공개번호 WO 2006/025416
국제공개일자 2006년03월09일
(30) 우선권주장
JP-P-2004-00256465 2004년09월03일 일본(JP)
(56) 선행기술조사문헌
JP2004245988 A
JP2002091297 A
JP2002091295 A
전체 청구항 수 : 총 21 항

(73) 특허권자
소니 주식회사
일본국 도쿄도 미나토쿠 코난 1-7-1
(72) 발명자
시라이 다이조
일본국 도쿄도 시나가와쿠 키타시나가와 6쵸메 7반 35고 소니가부시끼 가이샤내
바트 프레넬
벨기에 루완 비3000 그로트 베게인호프 58-59 루완루벤 리서치앤드 디벨롭먼트내
(74) 대리인
유미특허법인

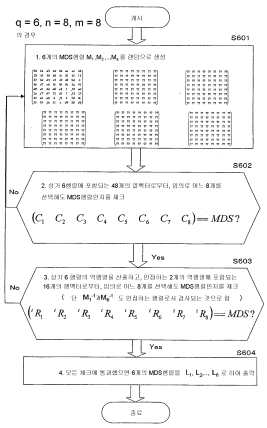
심사관 : 장상배

(54) 암호 처리 장치, 암호 처리 방법 및 기록매체

(57) 요 약

본 발명은, 해석 곤란성을 높인, 안정성이 높은 암호 처리 장치 및 방법을 실현한다. 비선형(非線形) 변환부 및 선형 변환부를 가지는 SPN형의 F함수를, 복수의 라운드로 반복 실행하는 페이스텔(Feistel)형 공통키 블록 암호 처리에 있어서, 복수의 라운드 각각에 대응하는 F함수의 선형 변환 처리를 정방 MDS 행렬을 적용한 선형 변환 처리에 의해 행한다. 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서 설정되는 정방(正方) MDS(Maximum Distance Separable) 행렬의 역행렬에 포함되는 임의의 m개의 열 벡터가 정방 MDS 행렬인 설정으로 한다. 본 구성에 의해, 공통키 블록 암호에 있어서의 선형 공격에 대한 내성이 향상되고, 암호 처리가 실현된다.

대 표 도 - 도17



특허청구의 범위

청구항 1

입력 정보를 비선형(非線形) 변환하고, 제1 비선형 변환 정보를 출력하는 제1 비선형 변환부와, 상기 제1 비선형 변환 정보를 선형 변환하여 제1 선형 변환 정보를 출력하는 제1 선형 변환부를 가지는 제1 암호 처리부와,

입력 정보를 비선형 변환하고, 제2 비선형 변환 정보를 출력하는 제2 비선형 변환부와, 상기 제2 비선형 변환 정보를 선형 변환하여 제2 선형 변환 정보를 출력하는 제2 선형 변환부를 가지는 제2 암호 처리부와,

상기 제2 암호 처리부로부터의 출력과, 상기 제1 암호 처리부로부터의 출력이 입력되는 배타적 논리합부

를 포함하고,

상기 제1 비선형 변환 정보를 제1 열 벡터로 나타내는 동시에 상기 제1 선형 변환 정보를 제2 열 벡터로 나타낸 경우에, 상기 제1 열 벡터를 상기 제2 열 벡터로 변환하는 제1 행렬의 역행열로부터 선택한 제1 행 벡터와,

상기 제2 비선형 변환 정보를 제3 열 벡터로 나타내는 동시에 상기 제2 선형 변환 정보를 제4 열 벡터로 나타낸 경우에, 상기 제3 열 벡터를 상기 제4 열 벡터로 변환하는 제2 행렬의 역행열로부터 선택한 제2 행 벡터는,

어느 행 벡터를 선택한 경우라도 서로 선형 독립인,

암호 처리 장치.

청구항 2

제1항에 있어서,

상기 암호 처리 장치는,

상기 배타적 논리합부에 있어서의 배타적 논리합 결과를 상기 제1 암호 처리부 또는 상기 제2 암호 처리부에 재차 입력하고, 상기 제1 암호 처리부 및 상기 제2 암호 처리부에 있어서의 암호 처리를 반복 실행하는 구성인,

암호 처리 장치.

청구항 3

페이스텔(Feistel)형 공통키 블록 암호 처리를 실행하는 암호 처리 장치로서,

비선형 변환부 및 선형 변환부를 가지는 SPN형의 F함수를, 복수의 라운드(round)로 반복 실행하는 구성을 가지고,

상기 복수의 라운드 각각에 대응하는 F함수의 선형 변환부는, m개의 비선형 변환부 각각이 출력하는 n비트, 총계 mn비트의 입력에 대한 선형 변환 처리를, 정방(正方) MDS(Maximum Distance Separable) 행렬을 적용한 선형 변환 처리로서 실행하는 구성이며,

적어도 연속하는 짝수 라운드 및 연속하는 홀수 라운드의 각각에 있어서는, 상이한 정방 MDS 행렬 : La, Lb가 적용되고, 또한 상기 정방 MDS 행렬의 역행열 : La^{-1}, Lb^{-1}을 구성하는 행 벡터로부터 임의로 선택한 m개의 행 벡터가 선형 독립인,

암호 처리 장치.

청구항 4

제3항에 있어서,

상기 암호 처리 장치에 있어서, 또한

상기 역행열 : La^{-1}, Lb^{-1}을 구성하는 행 벡터로부터 임의로 선택한 m개의 행 벡터에 의해 구성하는 행렬은 정방 MDS 행렬인, 암호 처리 장치.

청구항 5

제3항에 있어서,

상기 페이스텔형 공통키 블록 암호 처리의 알고리즘은, 라운드수 2r의 암호 처리 알고리즘이며,

상기 F함수의 선형 변환부는,

r개의 모든 짝수 라운드 및 r개의 모든 홀수 라운드에 있어서 $2 \leq q \leq r$의 q 종류가 상이한 정방 MDS 행렬을 차례로 반복 적용한 선형 변환 처리를 실행하는 구성인, 암호 처리 장치.

청구항 6

제3항에 있어서,

상기 F함수의 선형 변환부에 있어서 적용하는 상이한 복수 개의 정방 MDS 행렬의 각각은, 상기 복수 개의 정방 MDS 행렬의 역행열을 구성하는 행 벡터로부터 임의로 선택한 m개의 행 벡터가 선형 독립인, 암호 처리 장치.

청구항 7

제3항에 있어서,

상기 F함수의 선형 변환부에 있어서 적용하는 상이한 복수 개의 정방 MDS 행렬의 각각은, 상기 복수 개의 정방 MDS 행렬의 역행열을 구성하는 행 벡터로부터 임의로 선택한 m개의 행 벡터에 의해 구성하는 행렬도 정방 MDS 행렬로 되는, 암호 처리 장치.

청구항 8

제3항에 있어서,

상기 F함수의 선형 변환부에 있어서 적용하는 상이한 복수의 행렬 각각의 역행열은, 상기 복수 개의 정방 MDS 행렬을 구성하는 요소를 모두 포함하는 MDS 행렬 M으로부터 선택된 열 벡터에 의해 구성되는 행렬 M'로부터 추출된 행 벡터에 의해 구성되는 행렬에 의해 구성되어 있는, 암호 처리 장치.

청구항 9

암호 처리 장치에 있어서 암호 처리를 실행하는 암호 처리 방법으로서,

상기 암호 처리 장치의 제1 암호 처리부의 제1 비선형 변환부에 있어서 입력 정보를 비선형 변환하여 제1 비선형 변환 정보를 출력하고, 상기 제1 암호 처리부의 제1 선형 변환부에 있어서 상기 제1 비선형 변환 정보를 선형 변환하여 제1 선형 변환 정보를 출력하는 제1 암호 처리 스텝과,

상기 암호 처리 장치의 제2 암호 처리부의 제2 비선형 변환부에 있어서 입력 정보를 비선형 변환하여 제2 비선형 변환 정보를 출력하고, 상기 제2 암호 처리부의 제2 선형 변환부에 있어서 상기 제2 비선형 변환 정보를 선형 변환하여 제2 선형 변환 정보를 출력하는 제2 암호 처리 스텝과,

상기 암호 처리 장치의 배타적 논리합부는, 상기 제2 암호 처리부로부터의 출력과, 상기 제1 암호 처리부로부터의 출력을 입력하여 배타적 논리합 처리를 실행하는 배타적 논리합 스텝

을 포함하고,

상기 제1 암호 처리 스텝의 제1 선형 변환 처리는, 상기 제1 비선형 변환 정보를 제1 열 벡터로 나타내는 동시에 상기 제1 선형 변환 정보를 제2 열 벡터로 나타낸 경우에, 상기 제1 열 벡터를 상기 제2 열 벡터로 변환하는 제1 행렬을 적용한 제1 선형 변환 처리 실행 스텝이며,

상기 제2 암호 처리 스텝의 제2 선형 변환 처리는, 상기 제2 비선형 변환 정보를 제3 열 벡터로 나타내는 동시에 상기 제2 선형 변환 정보를 제4 열 벡터로 나타낸 경우에, 상기 제3 열 벡터를 상기 제4 열 벡터로 변환하는 제2 행렬을 적용한 제2 선형 변환 처리 실행 스텝이며,

상기 제1 선형 변환 처리 실행 스텝에 있어서 적용하는 상기 제1 행렬의 역행열로부터 선택한 제1 행 벡터와, 상기 제2 선형 변환 처리 실행 스텝에 있어서 적용하는 상기 제2 행렬의 역행열로부터 선택한 제2 행 벡터는 서로 선형 독립인,

암호 처리 방법.

청구항 10

제9항에 있어서,

상기 암호 처리 방법은,

상기 배타적 논리합 스텝에 있어서의 배타적 논리합 결과를 상기 제1 암호 처리부 또는 상기 제2 암호 처리부에 재차 입력하고, 상기 제1 암호 처리 스텝 및 상기 제2 암호 처리 스텝의 암호 처리를 반복 실행하는, 암호 처리 방법.

청구항 11

암호 처리 장치에 있어서, 페이스텔형 공통키 블록 암호 처리를 실행하는 암호 처리 방법으로서,

상기 암호 처리 장치의 비선형 변환 처리부 및 선형 변환 처리부에 있어서, 비선형 변환 처리 및 선형 변환 처리를 실행하는 SPN형의 F함수를, 복수의 라운드로 반복 실행하고,

상기 복수의 라운드 각각에 대응하는 F함수의 선형 변환 처리는, m개의 비선형 변환부 각각이 출력하는 n비트, 총계 mn비트의 입력에 대한 선형 변환 처리를, 정방 MDS 행렬을 적용한 선형 변환 처리로서 실행하고,

적어도 연속하는 짝수 라운드 및 연속하는 홀수 라운드의 각각에 있어서는, 상이한 정방 MDS 행렬 : La, Lb가 적용되고, 또한 상기 정방 MDS 행렬의 역행열 : La^{-1}, Lb^{-1}을 구성하는 행 벡터로부터 임의로 선택한 m개의 행 벡터는 선형 독립인,

암호 처리 방법.

청구항 12

제11항에 있어서,

상기 암호 처리 방법에 있어서, 또한

상기 역행열 : La^{-1}, Lb^{-1}을 구성하는 행 벡터로부터 임의로 선택한 m개의 행 벡터에 의해 구성하는 행렬은 정방 MDS 행렬인, 암호 처리 방법.

청구항 13

제11항에 있어서,

상기 페이스텔형 공통키 블록 암호 처리의 알고리즘은, 라운드수 2r의 암호 처리 알고리즘이며,

상기 F함수의 선형 변환 처리는,

r개의 모든 짝수 라운드 및 r개의 모든 홀수 라운드에 있어서 $2 \leq q \leq r$의 q 종류가 상이한 정방 MDS 행렬을 차례로 반복 적용한 선형 변환 처리를 실행하는, 암호 처리 방법.

청구항 14

제11항에 있어서,

상기 F함수의 선형 변환 처리에 있어서 적용하는 상이한 복수 개의 정방 MDS 행렬의 각각은, 상기 복수 개의 정방 MDS 행렬의 역행열을 구성하는 행 벡터로부터 임의로 선택한 m개의 행 벡터에 의해 구성하는 행렬은 선형 독립인, 암호 처리 방법.

청구항 15

제11항에 있어서,

상기 F함수의 선형 변환 처리에 있어서 적용하는 상이한 복수 개의 정방 MDS 행렬의 각각은, 상기 복수 개의 정방 MDS 행렬의 역행열을 구성하는 행 벡터로부터 임의로 선택한 m개의 행 벡터에 의해 구성하는 행렬도 정방 MDS 행렬로 되는, 암호 처리 방법.

청구항 16

제11항에 있어서,

상기 F함수의 선형 변환 처리에 있어서 적용하는 상이한 복수의 행렬 각각의 역행열은, 상기 복수 개의 정방 MDS 행렬을 구성하는 요소를 모두 포함하는 MDS 행렬 M으로부터 선택된 열 벡터에 의해 구성되는 행렬 M'로부터 추출된 행 벡터에 의해 구성되는 행렬에 의해 구성되어 있는, 암호 처리 방법.

청구항 17

암호 처리 장치에 있어서 암호 처리를 실행시키는 컴퓨터·프로그램이 기록된, 컴퓨터로 판독 가능한 기록 매체로서,

상기 암호 처리 장치의 제1 암호 처리부의 제1 비선형 변환부에 입력 정보를 비선형 변환시켜 제1 비선형 변환 정보를 출력시키고, 상기 제1 암호 처리부의 제1 선형 변환부에 상기 제1 비선형 변환 정보를 선형 변환시켜 제1 선형 변환 정보를 출력시키는 제1 암호 처리 스텝과,

상기 암호 처리 장치의 제2 암호 처리부의 제2 비선형 변환부에 입력 정보를 비선형 변환시켜 제2 비선형 변환 정보를 출력시키고, 상기 제2 암호 처리부의 제2 선형 변환부에 상기 제2 비선형 변환 정보를 선형 변환시켜 제2 선형 변환 정보를 출력시키는 제2 암호 처리 스텝과,

상기 암호 처리 장치의 배타적 논리합부에, 상기 제2 암호 처리부로부터의 출력과, 상기 제1 암호 처리부로부터의 출력을 입력하여 배타적 논리합 처리를 실행시키는 배타적 논리합 스텝

을 포함하고,

상기 제1 암호 처리 스텝의 제1 선형 변환 처리는, 상기 제1 비선형 변환 정보를 제1 열 벡터로 나타내는 동시에 상기 제1 선형 변환 정보를 제2 열 벡터로 나타낸 경우에, 상기 제1 열 벡터를 상기 제2 열 벡터로 변환하는 제1 행렬을 적용한 제1 선형 변환 처리 실행 스텝이며,

상기 제2 암호 처리 스텝의 제2 선형 변환 처리는, 상기 제2 비선형 변환 정보를 제3 열 벡터로 나타내는 동시에 상기 제2 선형 변환 정보를 제4 열 벡터로 나타낸 경우에, 상기 제3 열 벡터를 상기 제4 열 벡터로 변환하는 제2 행렬을 적용한 제2 선형 변환 처리 실행 스텝이며,

상기 제1 선형 변환 처리 실행 스텝에 있어서 적용하는 상기 제1 행렬의 역행열로부터 선택한 제1 행 벡터와, 제2 선형 변환 처리 실행 스텝에 있어서 적용하는 상기 제2 행렬의 역행열로부터 선택한 제2 행 벡터는 서로 선형 독립인 컴퓨터·프로그램이 기록된, 컴퓨터로 판독 가능한 기록 매체.

청구항 18

암호 처리 장치에 있어서, 페이스텔형 공통키 블록 암호 처리를 실행하는 컴퓨터·프로그램이 기록된, 컴퓨터로 판독 가능한 기록 매체로서,

상기 암호 처리 장치의 비선형 변환부 및 선형 변환부에, 비선형 변환 처리 및 선형 변환 처리를 실행하는 SPN형의 F함수를, 복수의 라운드로 반복 실행하는 스텝을 포함하고,

상기 복수의 라운드 각각에 대응하는 F함수의 선형 변환 처리는, m개의 비선형 변환부 각각이 출력하는 n비트, 총계 mn비트의 입력에 대한 선형 변환 처리를, 정방 MDS 행렬을 적용한 선형 변환 처리로서 실행하는 선형 변환 스텝이며,

상기 선형 변환 스텝에 있어서, 적어도 연속하는 짝수 라운드 및 연속하는 홀수 라운드의 각각에서는, 상이한 정방 MDS 행렬 : La, Lb가 적용되고, 또한 상기 정방 MDS 행렬의 역행열 : La^{-1}, Lb^{-1}을 구성하는 행 벡터로부터 임의로 선택한 m개의 행 벡터는 선형 독립인 컴퓨터·프로그램이 기록된, 컴퓨터로 판독 가능한 기록 매체.

청구항 19

암호 처리에 사용하는 키를 저장하는 메모리와, 암호 처리 프로그램을 실행하는 프로세서를 구비하는 암호 처리 장치로서,

상기 프로세서는, 입력 정보를 비선형 변환하고, 제1 비선형 변환 정보를 출력하는 제1 비선형 변환부와, 상기

제1 비선형 변환 정보를 선형 변환하여 제1 선형 변환 정보를 출력하는 제1 선형 변환부를 가지는 제1암호 처리부와, 입력 정보를 비선형 변환하고, 제2 비선형 변환 정보를 출력하는 제2 비선형 변환부와, 상기 제2 비선형 변환 정보를 선형 변환하여 제2 선형 변환 정보를 출력하는 제2 선형 변환부를 가지는 제2 암호 처리부와, 상기 제2 암호 처리부로부터의 출력과, 상기 제1 암호 처리부로부터의 출력이 입력되는 배타적 논리합부

를 포함하고,

상기 제1 비선형 변환 정보를 제1 열 벡터로 나타내는 동시에 상기 제1 선형 변환 정보를 제2 열 벡터로 나타낸 경우에, 상기 제1 열 벡터를 상기 제2 열 벡터로 변환하는 제1 행렬의 역행열로부터 선택한 제1 행 벡터와, 상기 제2 비선형 변환 정보를 제3 열 벡터로 나타내는 동시에 상기 제2 선형 변환 정보를 제4열 벡터로 나타낸 경우에, 상기 제3 열 벡터를 상기 제4 열 벡터로 변환하는 제2 행렬의 역행열로부터 선택한 제2행 벡터는, 어느 행 벡터를 선택한 경우라도 서로 선형 독립인,

암호 처리 장치.

청구항 20

제19항에 있어서,

상기 암호 처리 장치는, 상기 배타적 논리합부에 있어서의 배타적 논리합 결과를 상기 제1 암호 처리부 또는 상기 제2 암호 처리부에 재차 입력하고, 상기 제1 암호 처리부 및 상기 제2 암호 처리부에 있어서의 암호 처리를 반복하여 실행하는 구성인, 암호 처리 장치.

청구항 21

제19항 또는 제20항에 있어서,

상기 암호 처리 장치는, 외부 기기와의 데이터 통신을 행하는 송수신부를 더 포함하는, 암호 처리 장치.

명 세 서

기 술 분 야

[0001] 본 발명은, 암호 처리 장치, 암호 처리 방법 및 컴퓨터 프로그램에 관한 것이다. 보다 상세하게는, 암호 해석 처리, 공격 처리로서 알려진 선형(線形) 해석, 차분(差分) 해석에 대한 내성(耐性)을 향상시킨 암호 처리 장치, 암호 처리 방법 및 컴퓨터 프로그램에 관한 것이다.

배 경 기 술

[0002] 최근, 네트워크 통신, 전자 상거래의 발전에 따라, 통신에 있어서의 시큐리티 확보가 중요한 문제로 되고 있다. 시큐리티 확보의 하나의 방법이 암호 기술이며, 현재 다양한 암호화 방법을 이용한 통신이 실제로 행해지고 있다.

[0003] 예를 들면, IC 카드 등의 소형의 장치 중에 암호 처리 모듈을 매립하고, IC 카드와, 데이터 판독 기록 장치로서의 리더 라이터 사이에서 데이터 송수신을 행하고, 인증 처리 또는 송수신 데이터의 암호화, 복호를 행하는 시스템이 실용화되어 있다.

[0004] 암호 처리 알고리즘에는 다양한 것이 있지만, 크게 분류하면, 암호화키와 복호키를 상이한 키, 예를 들면, 공개키와 비밀키로서 설정하는 공개키 암호 방식과, 암호화키와 복호키를 공통의 키로 하여 설정하는 공통키 암호 방식으로 분류된다.

[0005] 공통키 암호 방식에도 다양한 알고리즘이 있지만, 그 하나로 공통키를 베이스로 하여 복수개의 키를 생성하고, 생성한 복수개의 키를 사용하여 블록 단위 64비트, 128비트 등)의 데이터 변환 처리를 반복 실행하는 방식이 있다. 이와 같은 키 생성 방식과 데이터 변환 처리를 적용한 알고리즘의 대표적인 것이 공통키 블록 암호 방식이다.

[0006] 대표적인 공통키 블록 암호의 알고리즘으로서는, 예를 들면, 미국 표준 암호로서의 DES(Data Encryption Standard) 알고리즘이 있으므로, 다양한 분야에 있어서 널리 사용되고 있다.

[0007] DES로 대표되는 공통키 블록 암호의 알고리즘은, 주로, 입력 데이터의 변환을 실행하는 라운드(round)

함수부와, 라운드 함수(F함수)부의 각 라운드에서 적용하는 키를 생성하는 키 스케줄부로 나눌 수 있다. 라운드 함수부의 각 라운드에서 적용하는 라운드 키(부(副)키)는, 1개의 마스터 키(주(主)키)에 따라, 키 스케줄부에 입력되어 생성되고, 각 라운드 함수부에서 적용된다.

[0008] 그러나, 이와 같은 공통키 암호 처리에 있어서는, 암호 해석에 의한 키의 누출이 문제로 되어 있다. 암호 해석 또는 공격 방법의 대표적인 방법으로서, 어떤 차분을 가지는 입력 데이터(평문(平文))와 그 출력 데이터(암호문)를 다수 해석함으로써 각 라운드 함수에 있어서의 적용 키를 해석하는 차분 해석(차분 해독법 또는 차분 공격이라고도 함)이나, 평문과 대응 암호문에 따른 해석을 행하는 선형 해석(선형 해독법 또는 선형 공격이라고도 함)이 알려져 있다.

[0009] 암호 해석에 의한 키의 해석이 용이하다는 것은, 그 암호 처리의 안전성이 낮다는 것으로 된다. 종래의 DES 알고리즘에 있어서는, 라운드 함수(F함수)부의 선형 변환부에 있어서 적용하는 처리(변환 행렬)가, 각 단(段)의 라운드에 있어서 같은 것이었으므로 해석을 행하기 쉽고, 결과로서 키의 해석의 용이성을 초래한다는 문제가 있다.

[0010] 본 발명은, 상기한 바와 같은 문제점을 감안하여 이루어진 것이며, 선형 해석이나 차분 해석에 대한 내성이 높은 공통키 블록 암호 알고리즘을 실현하는 암호 처리 장치, 암호 처리 방법 및 컴퓨터 프로그램을 제공하는 것을 목적으로 한다.

발명의 상세한 설명

[0011] 본 발명의 제1 측면은,

[0012] 페이스텔(Feistel)형 공통키 블록 암호 처리를 실행하는 암호 처리 장치로서,

[0013] 비선형(非線形) 변환부 및 선형 변환부를 가지는 SPN형의 F함수를, 복수의 라운드로 반복 실행하는 구성을 가지고,

[0014] 상기 복수의 라운드 각각에 대응하는 F함수의 선형 변환부는, m개의 비선형 변환부 각각이 출력하는 n비트, 총계 mn비트의 입력에 대한 선형 변환 처리를, 정방(正方) MDS(Maximum Distance Separable) 행렬을 적용한 선형 변환 처리로서 실행하는 구성이며,

[0015] 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서는, 상이한 정방 MDS 행렬: La, Lb가 적용되고, 또한 상기 정방 MDS 행렬의 역행렬: La^{-1}, Lb^{-1}을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬이 선형 독립인 것을 특징으로 하는 암호 처리 장치에 있다.

[0016] 또한, 본 발명의 암호 처리 장치의 일실시예에 있어서, 또한 상기 역행렬: La^{-1}, Lb^{-1}을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬이 정방 MDS 행렬인 것을 특징으로 한다.

[0017] 또한, 본 발명의 암호 처리 장치의 일실시예에 있어서, 상기 페이스텔형 공통키 블록 암호 처리의 알고리즘은, 라운드의 수가 2r인 암호 처리 알고리즘이며, 상기 F함수의 선형 변환부는, r개의 모든 짝수 라운드 및 r개의 모든 홀수 라운드에 있어서 $2 \leq q < r$의 q 종류의 상이한 정방 MDS 행렬을 차례로 반복 적용한 선형 변환 처리를 실행하는 구성인 것을 특징으로 한다.

[0018] 또한, 본 발명의 암호 처리 장치의 일실시예에 있어서, 상기 F함수의 선형 변환부에 있어서 적용하는 상이한 복수개의 정방 MDS 행렬은 각각, 상기 복수개의 정방 MDS 행렬을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬이 선형 독립인 정방 MDS 행렬인 것을 특징으로 한다.

[0019] 또한, 본 발명의 암호 처리 장치의 일실시예에 있어서, 상기 F함수의 선형 변환부에 있어서 적용하는 상이한 복수개의 정방 MDS 행렬은 각각, 상기 복수개의 정방 MDS 행렬을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬도 정방 MDS 행렬로 되는 정방 MDS 행렬인 것을 특징으로 한다.

[0020] 또한, 본 발명의 암호 처리 장치의 일실시예에 있어서, 상기 F함수의 선형 변환부에 있어서 적용하는 상이한 복수개의 정방 MDS 행렬은 각각, 상기 복수개의 정방 MDS 행렬을 구성하는 요소를 모두 포함하는 정방 MDS 행렬 M로부터 선택된 행 벡터에 의해 구성되는 행렬 M'로부터 추출된 열 벡터에 의해 구성되는 행렬에 의해 구성되어 있는 것을 특징으로 한다.

[0021] 또한, 본 발명의 제2 측면은,

[0022] 페이스텔형 공통키 블록 암호 처리를 실행하는 암호 처리 방법으로서,

[0023] 비선형 변환 처리 및 선형 변환 처리를 실행하는 SPN형의 F함수를, 복수의 라운드로 반복 실행하고,

[0024] 상기 복수의 라운드 각각에 대응하는 F함수의 선형 변환 처리는, m개의 비선형 변환부가 각각 출력하는 n비트, 총계 mn비트의 입력에 대한 선형 변환 처리를, 정방 MDS 행렬을 적용한 선형 변환 처리로서 실행하고,

[0025] 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서는, 상이한 정방 MDS 행렬: La, Lb가 적용되고, 또한 상기 정방 MDS 행렬의 역행렬: La^{-1}, Lb^{-1}을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬이 선형 독립인 정방 MDS 행렬에 의한 선형 변환 처리를 실행하는 것을 특징으로 하는 암호 처리 방법에 있다.

[0026] 또한, 본 발명의 암호 처리 방법의 일실시예에 있어서, 또한 상기 역행렬: La^{-1}, Lb^{-1}을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬이 정방 MDS 행렬인 정방 MDS 행렬에 의한 선형 변환 처리를 실행하는 것을 특징으로 한다.

[0027] 또한, 본 발명의 암호 처리 방법의 일실시예에 있어서, 상기 페이스텔형 공통키 블록 암호 처리의 알고리즘은, 라운드의 수가 2r인 암호 처리 알고리즘이며, 상기 F함수의 선형 변환 처리는, r개의 모든 짝수 라운드 및 r개의 모든 홀수 라운드에 있어서 $2 \leq q < r$의 q 종류의 상이한 정방 MDS 행렬을 차례로 반복 적용한 선형 변환 처리를 실행하는 것을 특징으로 한다.

[0028] 또한, 본 발명의 암호 처리 방법의 일실시예에 있어서, 상기 F함수의 선형 변환 처리에 있어서 적용하는 상이한 복수개의 정방 MDS 행렬은 각각, 상기 복수개의 정방 MDS 행렬을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬이 선형 독립인 정방 MDS 행렬인 것을 특징으로 한다.

[0029] 또한, 본 발명의 암호 처리 방법의 일실시예에 있어서, 상기 F함수의 선형 변환 처리에 있어서 적용하는 상이한 복수개의 정방 MDS 행렬은 각각, 상기 복수개의 정방 MDS 행렬을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬도 정방 MDS 행렬로 되는 정방 MDS 행렬인 것을 특징으로 한다.

[0030] 또한, 본 발명의 암호 처리 방법의 일실시예에 있어서, 상기 F함수의 선형 변환 처리에 있어서 적용하는 상이한 복수개의 정방 MDS 행렬은 각각, 상기 복수개의 정방 MDS 행렬을 구성하는 요소를 모두 포함하는 정방 MDS 행렬 M으로부터 선택된 행 벡터에 의해 구성되는 행렬 M'로부터 추출된 열 벡터에 의해 구성되는 행렬에 의해 구성되어 있는 것을 특징으로 한다.

[0031] 또한, 본 발명의 제3 측면은,

[0032] 페이스텔형 공통키 블록 암호 처리를 실행하는 컴퓨터 프로그램으로서,

[0033] 비선형 변환 처리 및 선형 변환 처리를 실행하는 SPN형의 F함수를, 복수의 라운드로 반복 실행하는 스텝을 포함하고,

[0034] 상기 복수의 라운드 각각에 대응하는 F함수의 선형 변환 처리는, m개의 비선형 변환부가 각각 출력하는 n비트, 총계 mn비트의 입력에 대한 선형 변환 처리를, 정방 MDS 행렬을 적용한 선형 변환 처리로서 실행하는 선형 변환 스텝이며,

[0035] 상기 선형 변환 스텝에 있어서, 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에서는, 상이한 정방 MDS 행렬: La, Lb가 적용되고, 또한 상기 정방 MDS 행렬의 역행렬: La^{-1}, Lb^{-1}을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬이 선형 독립인 정방 MDS 행렬에 의한 선형 변환 처리를 실행하는 것을 특징으로 하는 컴퓨터 프로그램에 있다.

[0036] 그리고, 본 발명의 컴퓨터 프로그램은, 예를 들면, 다양한 프로그램·코드를 실행 가능한 컴퓨터·시스템에 대하여, 컴퓨터 판독 가능한 형식으로 제공하는 기억 매체, 통신 매체, 예를 들면, CD나 FD, MO 등의 기록 매체, 또는 네트워크 등의 통신 매체에 의해 제공 가능한 컴퓨터 프로그램이다. 이와 같은 프로그램을 컴퓨터 판독 가능한 형식으로 제공함으로써, 컴퓨터·시스템 상에서 프로그램에 따른 처리가 실현된다.

[0037] 본 발명의 또 다른 목적, 특징이나 이점은, 후술하는 본 발명의 실시예나 첨부하는 도면에 따른 보다 상세한 설명에 따라서, 명백해 질 것이다. 그리고, 본 명세서에 있어서 시스템이란, 복수개의 장치의 논리적 집합 구성이며, 각 구성의 장치가 동일 하우징 내에 있는 것에 한정되지는 않는다.

[0038] 본 발명의 구성에 의하면, 비선형 변환부 및 선형 변환부를 가지는 SPN형의 F함수를, 복수의 라운드로 반복 실행하는 페이스텔형 공통키 블록 암호 처리에 있어서, 복수의 라운드 각각에 대응하는 F함수의 선형 변환 처리를, 정방 MDS 행렬을 적용한 선형 변환 처리로서 실행하는 동시에, 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서 상이한 정방 MDS 행렬: La, Lb를 적용하고, 또한 상기 정방 MDS 행렬의 역행렬: La^{-1}, Lb^{-1}을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬이 선형 독립인지, 또는 정방 MDS 행렬을 구성하는 성질로 한 정방 MDS 행렬에 의한 선형 변환 처리를 실행하는 구성으로 하였으므로, 공통키 블록 암호에 있어서의 선형 공격에 대한 내성이 향상되고, 암호 키 등의 해석 곤란성이 높아지게 되어, 안정성이 높은 암호 처리가 실현된다.

[0039] 또한, 본 발명의 구성에 의하면, 비선형 변환부 및 선형 변환부를 가지는 SPN형의 F함수를, 복수의 라운드로 반복 실행하는 페이스텔형 공통키 블록 암호 처리에 있어서, 복수의 라운드 각각에 대응하는 F함수의 선형 변환 처리를, 정방 MDS 행렬을 적용한 선형 변환 처리로서 실행하는 동시에, 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서 상이한 정방 MDS 행렬을 적용하는 구성으로 하고, 이들 정방 MDS 행렬 자신이, 선형 독립성을 나타내거나, 또는 정방 MDS 행렬을 구성하는 구성으로 하였으므로, 액티브 S박스의 기여에 의한 동시 차분 캔슬의 발생하지 않는 것이 보증되어 공통키 블록 암호에 있어서의 차분 공격에 대한 강도 지표의 하나인 암호화 함수 전체에서의 액티브 S박스의 최소수를 크게 하는 것이 가능해진다. 본 구성에 의해, 선형 공격, 차분 공격의 양쪽에 대하여 내성이 향상되고, 보다 안정성이 높은 암호 처리가 실현된다.

실 시 예

[0058] 이하, 본 발명의 암호 처리 장치, 및 암호 처리 방법 및 컴퓨터 프로그램의 상세에 대하여 설명한다. 그리고, 설명은, 이하의 항목순으로 행한다.

[0059] 1. 공통키 블록 암호 알고리즘에 있어서의 차분 해석 처리

[0060] 2. 공통키 블록 암호 알고리즘에 있어서의 선형 해석 처리

[0061] 3. 본 발명에 따른 암호 처리 알고리즘

[0062] (3-a) 차분 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예

[0063] (3-b) 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예

[0064] (3-c) 차분 공격 및 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예

[0065] [1.공통키 블록 암호 알고리즘에 있어서의 차분 해석 처리]

[0066] 먼저, DES(Data Encryption Standard) 암호 처리에 대표되는 공통키 블록 암호 알고리즘에 있어서의 차분 해석 처리의 개요에 대하여, 일반화한 공통키 블록 암호 모델을 사용하여 설명한다.

[0067] 공통키 블록 암호의 알고리즘은, 주로, 입력 데이터의 변환을 실행하는 라운드 함수부와, 라운드 함수부의 각 라운드로 적용하는 키를 생성하는 키 스케줄부로 나눌 수가 있다. 라운드 함수부의 각 라운드로 적용하는 키(부키)는, 1개의 마스터 키(주키)에 따라, 키 스케줄부에 입력되어 생성되고, 각 라운드 함수부에서 적용된다. 이 공통키 암호 방식의 대표적인 방식으로 미국 연방 표준 암호 방식으로서의 DES(Data Encryption Standard)가 있다.

[0068] 페이스텔 구조라고 하는 대표적인 공통키 블록 암호의 구조에 대하여, 도 1을 참조하여 설명한다.

[0069] 페이스텔 구조는, 변환 함수의 단순한 반복에 의해, 평문을 암호문으로 변환하는 구조를 가진다. 평문의 길이를 2mn비트로 한다. 단, m, n는 함께 정수(整數)이다. 처음에, 2mn비트의 평문을, mn비트의 2개의 입력 데이터 PL(Plain-Left)(101), PR(Plain-Right)(102)로 분할하고, 이것을 입력값으로 한다.

[0070] 페이스텔 구조는 라운드 함수라는 기본 구조의 반복으로 표현되고, 각 라운드에 포함되는 데이터 변환 함수는 F함수(120)라고 한다. 도 1의 구성에서는, F함수(라운드 함수)(120)가 r단(段) 반복되는 구성예를 나타내고 있다.

[0071] 예를 들면, 제1 라운드에서는, mn비트의 입력 데이터 X와, 키 생성부(도시하지 않음)로부터 입력되는 mn비트의 라운드 키 K_1(103)가 F함수(120)에 입력되고, F함수(120)에 있어서의 데이터 변환 처리 후에 mn비트의 데이터 Y를 출력한다. 출력은 또다른 한쪽의 전단으로부터의 입력 데이터(제1 단의 경우에는 입력 데이터 P_L)와 배타적

논리합부(104)에 있어서, 배타적 논리합 연산이 행해지고, mn비트의 연산 결과가 다음의 라운드 함수로 출력된다. 이 처리, 즉 F함수를 정해진 라운드의 수 (r)만큼 반복 적용하여 암호화 처리가 완료되고, 암호문의 분할 데이터 CL(Cipher-Left), CR(Cipher-Right)가 출력된다. 이상의 구성으로부터, 페이스텔 구조의 복호 처리는 라운드 키를 삽입하는 순서를 역으로 하는 것만으로 되고, 역함수를 구성할 필요가 없는 것이 안내된다.

[0072] 각 라운드의 함수로서 설정되는 F함수(120)의 구성에 대하여, 도 2를 참조하여 설명한다. 도 2 (a)는, 1개의 라운드에 있어서의 F함수(120)에 대한 입력 및 출력을 나타낸 도면이며, 도 2 (b)는, F함수(120)의 구성의 상세를 나타낸 도면이다. F함수(120)는, 도 2 (b)에 나타낸 바와 같이, 비선형 변환층과 선형 변환층을 접속한 이른바 SPN형의 구성을 가진다.

[0073] SPN형의 F함수(120)는, 도 2 (b)에 나타낸 바와 같이, 비선형 변환 처리를 실행하는 복수개의 S박스(S-box)(121)를 가진다. 라운드 함수부의 전단(前段)으로부터의 mn비트의 입력값 X는, 키 스케줄부로부터 입력되는 라운드 키 K_i와 배타적 논리합이 실행되고, 그 출력이 n비트씩 비선형 변환 처리를 실행하는 복수개(m개)의 S박스(121)에 입력된다. 각 S박스에서는, 예를 들면, 변환 테이블을 적용한 비선형 변환 처리가 실행된다.

[0074] S박스(121)로부터의 출력 데이터인 mn비트의 출력값 Z는, 선형 변환 처리를 실행하는 선형 변환부(122)에 입력되어, 예를 들면, 비트 위치의 교체 처리 등의 선형 변환 처리가 실행되고, mn비트의 출력값 Y를 출력한다. 이 출력값 Y가 전단으로부터의 입력 데이터와 배타적 논리합되고, 다음의 라운드의 F함수의 입력값으로 된다.

[0075] 도 2에 나타낸 F함수(120)는, 입출력 비트 길이가 m×n(m, n: 정수)비트이며, 비선형 변환층은 n비트의 입출력을 가지는 비선형 변환층로서의 S박스(121)는, m개 병렬로 줄선 구성을 가지고, 선형 변환층으로서의 선형 변환부(122)는 n차의 기약(旣約) 다항식에 의해 정의되는 2의 확대체 GF(2n) 상의 원(元)을 요소로서 가지는 m차의 정방 행렬에 따른 선형 변환 처리를 실행한다.

[0076] 선형 변환부(122)에 있어서의 선형 변환 처리에 적용하는 정방 행렬의 예를 도 3에 나타낸다. 도 3에 나타낸 정방 행렬(125)은, n= 8, m= 8의 경우의 예이다. 비선형 변환부(S박스(121))로부터 출력된 m개의 n비트 데이터 Z[1], Z[2], . . ., Z [m] 에 대하여 미리 정해진 정방 행렬(125)을 적용한 연산에 의해 선형 변환이 행해지고, F함수(라운드 함수)출력으로서의, Y[1], Y[2], . . ., Y [m] 가 결정된다. 단, 이 때 각 데이터의 행렬의 요소에 대한 선형 연산은 미리 정해진 2의 확대체 GF(2n) 상에서 행해진다.

[0077] 지금까지의 페이스텔형 암호에서는, 모든 단의 F함수에 같은 선형 변환층을 사용하고 있으므로, 차분의 전파시에 동시에 복수개의 차분이 캔슬되어 버리는 성질이 존재했다. 배경 기술의 난에 있어서 설명한 바와 같이, 암호 해석 방법의 대표적인 방법으로서, 어떤 차분을 가지는 입력 데이터(평문)와 그 출력 데이터(암호문)를 다수 해석함으로써 각 라운드 함수에 있어서의 적용 키를 해석하는 차분 해석(또는 차분 해독법)이 알려져 있고, 종래의 DES 암호 알고리즘 등의 공통키 블록 암호에 있어서는, F함수(120)부의 선형 변환부(122)에 있어서 적용하는 처리(변환 행렬)를, 각 단의 라운드에 있어서 같은 것으로 설정하고 있으므로, 차분 해석을 행하기 쉽고, 결과로서 키의 해석의 용이성을 초래하고 있다.

[0078] 차분의 전파시에, 동시에 복수개의 차분이 캔슬되는 예에 대하여, 도 4를 참조하여 설명한다. 그리고, 본 명세서에 있어서는, 차분을 나타내는 경우에는Δ(델타) 기호를 부여하여 나타낸다.

[0079] 도 4는 m= 8, n= 8의 128bit 블록 암호에 있어서의 3단의 동시 차분 캔슬의 상태를 설명하는 도면이다. 단, 도면 중에는 64bit의 데이터를 바이트 단위로 구획하여 벡터로서 표현하고, 각각의 요소를 16진수로 표기하는 것으로 한다.

[0080] 3단 구성을 가지는 F함수에서의 동시 차분 캔슬은, 예를 들면, 이하의 데이터 상태 1 ~ 4의 설정 메카니즘에 따라 발생한다. 이하에 설명하는 메카니즘이 발생하는 데이터 상태는, 다수의 차분 입력 데이터를 설정함으로써 발생시킬 수 있는 데이터 상태이며, 이른바 차분 해석에 있어서의 키(라운드 키)의 해석에 있어서 발생할 수 있다.

[0081] (상태 1)

[0082] i라운드로의 입력 차분의 좌측 반은, 모두 제로인 입력 차분(ΔXi-1= (00, 00, 00, 00, 00, 00, 00, 00))이며, 우측 반의 입력 차분이 단 하나의 S-box로의 입력을 제외하고 제로인 입력 차분(ΔXi= (34, 00, 00, 00, 00, 00, 00, 00))인 것으로 한다. 이 데이터 상태는, 다수의 차분 입력 데이터가 설정됨으로써, i라운드에 있어서, 이와 같은 데이터 상태를 얻을 수 있다는 것이다.

[0083] 그리고, ΔXi= (34, 00, 00, 00, 00, 00, 00, 00)의 8개의 각 요소는, F함수 중에 구성되는 m개(m= 8)의 S박스 각각에 대한 입력 차분에 대응한다. 차분(34)이 제1 S박스(도 4 중 (S1))에 입력되고, (00)가, 제2 ~8 S박스에 대한 입력 차분이다.

[0084] 그리고, 제로(00)의 입력 차분을 가지는 S박스의 출력 차분은 제로(00)이며, 차분 데이터에 관한 한, 제로(00)의 입력 차분을 가지는 S박스는, 어떤 작용도 행하지 않는 것이며, 액티브하지 않은 즉 비액티브 S박스라고 한다. 한편, 비제로의 입력 차분(도 4의 예에서는 차분: 34)을 가지는 S박스는, 비제로의 입력 차분에 대응한 비선형 변환 결과를 출력 차분으로서 발생시키므로, 액티브 S박스(Active S-box)라고 한다.

[0085] 도 4의 예에서는, 비제로의 입력 차분(34)을 입력하는 1개의 액티브 S박스(S1)의 출력 차분(b7)을 발생시키고 있고, 그 외의 비액티브 S박스 S2~S8는 제로의 입력 차분(00)에 따라 출력 차분(00)을 발생시키고, 선형 변환부의 차분 입력으로 하고 있다.

[0086] (상태 2)

[0087] i라운드로의 비제로의 입력 차분(도 4의 예에서는 차분: 34)을 가지는 S박스(이하, 액티브 S박스(Active S-box)라고 함)로부터의 출력 차분은 선형 변환층에서 확산된 후 F함수로부터 출력(출력값= ΔYi)되고, 그대로 다음 라운드로의 입력 차분ΔXi+1로 된다.

[0088] 도 4의 예에서의 선형 변환은, 각 라운드의 F함수에 있어서 공통되는 예를 들면, 도 5에 나타낸 어느 특정의 정방 행렬(125)에 의한 선형 변환이 실행되고 i라운드의 F함수 출력 차분으로서의 ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)를 출력한다. 도 5에 나타낸 선형 변환 구성으로부터 이해할 수 있는 바와 같이, 출력 차분ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)는, 1개의 액티브 S박스(S1)로부터의 출력 요소 Z[1]= b7에만 의존한 값으로서 결정된다.

[0089] 이 i라운드의 F함수 출력 차분으로서의 ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)는, 도 4에 나타낸 배타적 논리합부(131)에 있어서, 모두 제로인 입력 차분(ΔXi-1= (00, 00, 00, 00, 00, 00, 00, 00)와 배타적 논리합(XOR) 연산이 실행되고, 연산 결과가, 다음의 라운드(i+1)로의 입력 차분ΔXi+1로 된다.

[0090] i라운드의 F함수 출력 차분으로서의 ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)와, 모두 제로인 입력 차분ΔXi-1= (00, 00, 00, 00, 00, 00, 00, 00)와의 배타적 논리합(XOR) 결과는, ΔYi이므로, 다음의 라운드(i+1)로의 입력 차분ΔXi+1= ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)로 된다.

[0091] (상태 3)

[0092] i+1 라운드의 F함수로부터의 출력 차분ΔYi+1이, i라운드에서의 Active S-box의 위치에만 비제로값을 가진다. 이 데이터 상태는, 다수의 차분 입력 데이터가 설정됨으로써, 이와 같은 데이터 상태를 얻을 수 있다는 것이다.

[0093] 즉 ΔYi+1= (ad, 00, 00, 00, 00, 00, 00, 00)이며, i라운드와 마찬가지로, 비제로의 차분값(도 4의 예에서는 차분: 34)을 가지는 S-box의 위치(제1 S박스(S1))에만 비제로값을 가진다. 그리고, 명백하게 ad≠00이다.

[0094] (상태 4)

[0095] i+2 라운드의 액티브 S박스(Active S-box)(S1)의 출력 차분이 i라운드에서의 액티브 S박스(Active S-box)(S1)의 출력 차분과 일치한 경우, 즉 도 4에 나타낸 바와 같이 i+2 라운드의 액티브 S박스(S1)의 출력 차분이 b7로 되고, i라운드에서의 액티브 S박스(S1)의 출력 차분(b7)과 일치한다. 이 데이터 상태는, 다수의 차분 입력 데이터가 설정됨으로써, 이와 같은 데이터 상태를 얻을 수 있다는 것이다.

[0096] 이 데이터 상태가 발생하면, i+2 라운드의 F함수의 출력 차분ΔYi+2= (98, c4, b4, d3, ac, 72, 0f, 32)가, 2개 전의 라운드인 i라운드의 F함수의 출력 차분ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)와 일치하게 된다.

[0097] 이 결과, 배타적 논리합부(133)에서는,

[0098] ΔXi+1= ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)와,

[0099] ΔYi+2= (98, c4, b4, d3, ac, 72, 0f, 32)

[0100] 의 동일한 값끼리의 배타적 논리합 연산이 실행되게 되고, 배타적 논리합 연산 결과로서 올 0의 값을 출력한다.

[0101] 그 결과, 다음의 단(라운드 i+3)로의 출력 차분의 전단(i+2 라운드)으로부터의 좌측의 입력 차분ΔXi+3= (00, 00, 00, 00, 00, 00, 00, 00)로 된다.

[0102] 이 라운드 i+3로의 좌측 입력ΔXi+3= (00, 00, 00, 00, 00, 00, 00, 00)는, 라운드 i로의 좌측 입력ΔXi-1= (00, 00, 00, 00, 00, 00, 00, 00)와 마찬가지로 올 제로이며, 라운드 i+3 이후의 라운드에 있어서도, 라운드 i~i+2와 마찬가지의 처리가 반복될 가능성이 있다.

[0103] 이 결과, 라운드의 수의 성장에 대하여 액티브 S박스의 수가 증대하지 않고, 차분 공격에 대한 강도가 그만큼 신장되지 않는 문제를 발생시킨다.

[0104] 공통키 블록 암호에 있어서, 차분 공격에 대한 강도 지표의 하나로서 암호화 함수 전체에서의 액티브 S박스의 최소수가 알려져 있다. 액티브 S박스수의 최소수가 클수록 차분 공격에 대한 내성이 높은 것으로 판단된다.

[0105] 전술한 바와 같이, 차분 해석(차분 공격)에 있어서는, 어떤 차분을 가지는 입력 데이터(평문)와 그 출력 데이터(암호문)를 다수 설정하여 이 대응을 해석함으로써 각 라운드 함수에 있어서의 적용 키를 해석하는 방법이며, 이 차분 해석에 있어서, 액티브 S박스의 수를 적게 할 수 있으면, 해석이 용이해지므로, 해석 프로세스수를 삭감할 수 있다.

[0106] 전술한 도 4를 참조한 예에서는, 제1 S박스(S1)만이 액티브 S박스인 패턴의 발생 상태를 제시했지만, 그 외의 S박스(S2~S8)에 대하여도, 차분 해석의 입력 데이터의 설정에 의해, 각 S박스만을 액티브 S박스로 한 설정이 가능하며, 이와 같은 차분 해석 프로세스를 실행함으로써, 각 S박스의 비선형 변환 처리의 해석, 또한 F함수에 대하여 입력되는 라운드 키의 해석이 가능해진다.

[0107] 이와 같은 차분 해석에 대한 내성을 향상시키기 위해서는, 액티브 S박스의 수가 항상 많은 상태를 유지하는 것, 즉 액티브 S박스의 최소수가 많은 것이 필요하다.

[0108] 도 4를 참조하여 설명한 예에 있어서, 우측으로부터 좌측으로 입력을 행하는 F함수, 즉 제i 라운드와 제i+2 라운드만을 액티브 S박스 산출 처리 대상 라운드로서 보았을 경우, 액티브 S박스수는 불과 2이며, 좌측으로부터 우측으로 입력을 행하는 F함수, 즉 제i+1 라운드에서는 액티브 S박스수가 8이지만, 동시 차분 캔슬에 의해 제i+3 라운드에서의 액티브 S박스수가 0으로 되어 버리므로, 차분 해석에 의한 각 S박스의 비선형 변환 처리의 해석 처리가 용이해진다.

[0109] 도 4에 나타낸 공통키 블록 암호 알고리즘은, 각 라운드에 있어서의 선형 변환부에 있어서 적용하는 선형 변환 행렬이 같은 것이며, 이 구성에 기인하여, 특히 우측으로부터 좌측으로 입력을 행하는 F함수에 있어서의 불과 2개의 액티브 S박스에 의해 동시 차분 캔슬의 발생 가능성이 생기고 있다. 따라서, 라운드의 수의 성장에 대하여 액티브 S박스의 최소수가 충분히 증대하지 않고, 차분 공격에 대한 강도가 그만큼 신장되지 않는 문제가 있다.

[0110] 다음에, 마찬가지로, 같은 선형 변환 행렬을 모든 단(라운드)의 F함수에 사용하는 구성에 있어서, 5라운드에 걸친 동시 차분 캔슬의 발생 메카니즘에 대하여, 도 6을 참조하여 설명한다.

[0111] 도 6은 m= 8, n= 8의 128bit 블록 암호에 있어서의 5단의 동시 차분 캔슬의 상태를 설명하는 도면이다. 단, 도면 중에는 64bit의 데이터를 바이트 단위로 구획하여 벡터로서 표현하고, 각각의 요소를 16진수로 표기하는 것으로 한다.

[0112] 5단 구성을 가지는 F함수에서의 동시 차분 캔슬은, 예를 들면, 이하의 데이터 상태 1 ~ 7의 설정 메카니즘에 따라 발생한다. 이하에 설명하는 메카니즘이 발생하는 데이터 상태는, 다수의 차분 입력 데이터가 설정됨으로써 발생시킬 수 있는 데이터 상태이므로, 이른바 차분 해석에 있어서의 키(라운드 키)의 해석에 있어서 발생할 수 있다.

[0113] (상태 1)

[0114] i라운드로의 입력 차분의 좌측 반은, 모두 제로인 입력 차분(ΔXi-1= (00, 00, 00, 00, 00, 00, 00, 00))이며, 우측 반의 입력 차분이 단 하나의 S-box로의 입력을 제외하고 제로인 입력 차분(ΔXi= (34, 00, 00, 00, 00, 00, 00, 00))인 것으로 한다. 이 데이터 상태는, 다수의 차분 입력 데이터가 설정됨으로써, i라운드에 있어서, 이와 같은 데이터 상태를 얻을 수 있다는 것이다.

[0115] 그리고, ΔXi= (34, 00, 00, 00, 00, 00, 00, 00)의 8개의 각 요소는, F함수 중에 구성되는 m개(m= 8)의 S박스 각각에 대한 입력 차분에 대응한다. (34)가 제1 S박스(도 6 중의 (S1))에 입력되고, (00)이, 제2 ~제8 S박스에 대한 입력 차분이다.

[0116] 그리고, 전술한 바와 같이, 제로(00)의 입력 차분을 가지는 S박스의 출력 차분은 제로(00)이며, 차분 데이터에

관한 한, 제로(00)의 입력 차분을 가지는 S박스는, 어떤 작용도 행하지 않는 것이며, 액티브하지 않은 즉 비액티브 S박스라고 한다. 한편, 비제로의 입력 차분(도 6의 예에서는 차분: 34)을 가지는 S박스(S1)만이, 비제로의 입력 차분에 대응한 비선형 변환 결과를 출력 차분으로서 발생시키므로, 액티브 S박스(Active S-box)이다.

[0117] 도 6의 예에서는, 비제로의 입력 차분(34)을 입력하는 1개의 액티브 S박스(S1)의 출력 차분(b7)을 발생시키고 있고, 그 외의 비액티브 S박스 S2~S8는 제로의 입력 차분(00)에 따라 출력 차분(00)을 발생시켜, 선형 변환부의 차분 입력으로 하고 있다.

[0118] (상태 2)

[0119] i라운드로의 비제로의 입력 차분(도 4의 예에서는 차분: 34)을 가지는 S박스(이하, 액티브 S박스(Active S-box)라고 함)로부터의 출력 차분은 선형 변환층에서 확산된 후 F함수로부터 출력(출력값= ΔYi)되고, 그대로 다음의 라운드로의 입력 차분ΔXi+1로 된다.

[0120] 도 6의 예에 있어서, 각 라운드에 공통의 예를 들면, 도 5에 나타낸 어느 특정한 정방 행렬(125)에 의한 선형 변환이 실행되고 i라운드의 F함수 출력 차분으로서의 ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)를 출력한다.

[0121] i라운드의 F함수 출력 차분으로서의 ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)는, 도 6에 나타낸 배타적 논리합부(141)에 있어서, 모두 제로인 입력 차분(ΔXi-1= (00, 00, 00, 00, 00, 00, 00, 00)와 배타적 논리합(XOR) 연산이 실행되고, 연산 결과가, 다음의 라운드(i+1)로의 입력 차분ΔXi+1로 된다.

[0122] i라운드의 F함수 출력 차분으로서의 ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)와, 모두 제로인 입력 차분(ΔXi-1= (00, 00, 00, 00, 00, 00, 00, 00))와의 배타적 논리합(XOR) 결과는, ΔYi이므로, 다음의 라운드(i+1)로의 입력 차분ΔXi+1= ΔYi= (98, c4, b4, d3, ac, 72, 0f, 32)로 된다.

[0123] (상태 3)

[0124] i+1 라운드의 F함수로부터의 출력 차분ΔYi+1이, i라운드에서의 Active S-box의 위치에만 비제로값을 가진다. 이 데이터 상태는, 다수의 차분 입력 데이터가 설정됨으로써, 이와 같은 데이터 상태를 얻을 수 있다는 것이다.

[0125] 즉 ΔYi+1= (34, 00, 00, 00, 00, 00, 00, 00)이며, i라운드와 마찬가지로, 비제로의 차분값(도 6의 예에서는 차분: 34)을 가지는 S-box의 위치(제1 S박스(S1)에만 비제로값을 가진다.

[0126] (상태 4)

[0127] i+2 라운드의 F함수에 대한 입력은, ΔXi= (34, 00, 00, 00, 00, 00, 00, 00)와, ΔYi+1= (34, 00, 00, 00, 00, 00, 00, 00)와의 배타적 논리합부(142)에 있어서의 배타적 논리합 결과, 즉 동일 데이터끼리의 배타적 논리합이며, 올 제로의 입력, ΔXi+2= (00, 00, 00, 00, 00, 00, 00, 00)로 되고, 그 결과, i+2 라운드의 F함수로부터의 출력 차분도, 올 제로의 출력 차분, ΔYi+2= (00, 00, 00, 00, 00, 00, 00, 00)로 된다.

[0128] (상태 5)

[0129] i+3 라운드의 F함수에 대한 입력은, ΔXi+1= (98, c4, b4, d3, ac, 72, 0f, 32)와, 올 제로의 i+2 라운드의 F함수 출력 차분ΔYi+2= (00, 00, 00, 00, 00, 00, 00, 00)와의 배타적 논리합부(143)에 있어서의 배타적 논리합 결과이며, i+3 라운드의 F함수에 대한 입력ΔXi+3= ΔXi+1= (98, c4, b4, d3, ac, 72, 0f, 32)로 된다.

[0130] (상태 6)

[0131] i+3 라운드의 F함수 출력 차분이, ΔYi+3= (43, 00, 00, 00, 00, 00, 00, 00)로 되고, 올 제로의 ΔXi+2= (00, 00, 00, 00, 00, 00, 00, 00)와의 배타적 논리합부(144)에 있어서의 배타적 논리합의 결과로서의 ΔXi+4= ΔYi+3= (43, 00, 00, 00, 00, 00, 00, 00)가 i+4 라운드의 F함수 입력 차분으로 된다.

[0132] (상태 7)

[0133] i+4 라운드의 액티브 S박스(Active S-box)(S1)의 출력 차분이 i라운드에서의 액티브 S박스(Active S-box)(S1)의 출력 차분과 일치한 경우, 즉 도 6에 나타낸 바와 같이 i+4 라운드의 액티브 S박스(S1)의 출력 차분이 b7로 되고, i라운드에서의 액티브 S박스(S1)의 출력 차분(b7)과 일치한다. 이 데이터 상태는, 다수의 차분 입력 데이터가 설정됨으로써, 이와 같은 데이터 상태를 얻을 수 있다는 것이다.

[0134] 이 데이터 상태가 발생하면, i+4 라운드의 F함수의 출력 차분ΔYi+4= (98, c4, b4, d3, ac, 72, 0f, 32)가, 2개 전의 라운드인 i+2 라운드의 배타적 논리합부(143)의 출력 차분ΔXi+3= (98, c4, b4, d3, ac, 72, 0f, 32)

와 일치하게 된다.

[0135] 이 결과, 배타적 논리합부(145)에서는,

[0136] ΔXi+3= (98, c4, b4, d3, ac, 72, 0f, 32)와,

[0137] ΔYi+4= (98, c4, b4, d3, ac, 72, 0f, 32)

[0138] 의 동일한 값끼리의 배타적 논리합 연산이 실행되게 되고, 배타적 논리합 연산 결과로서 올 0의 값을 출력한다.

[0139] 그 결과, 다음의 단(라운드 i+5)로의 입력 차분은, ΔXi+5= (00, 00, 00, 00, 00, 00, 00, 00)로서 설정된다.

[0140] 이 라운드 i+5로의 좌측 입력ΔXi+5= (00, 00, 00, 00, 00, 00, 00, 00)는, 라운드 i로의 좌측 입력ΔXi-1= (00, 00, 00, 00, 00, 00, 00, 00)와 마찬가지로 올 제로이며, 라운드 i+5 이후의 라운드에 있어서도, 라운드 i~i+4와 마찬가지의 처리가 반복될 가능성이 있다.

[0141] 이 결과, 라운드의 수의 성장에 대하여 액티브 S박스의 수가 증대하지 않고, 차분 공격에 대한 강도가 그만큼 신장되지 않는 문제를 발생시킨다.

[0142] 전술한 바와 같이, 차분 해석(차분 공격)에 있어서는, 어떤 차분을 가지는 입력 데이터(평문)와 그 출력 데이터(암호문)를 다수 설정하여 이 대응을 해석함으로써 각 라운드 함수에 있어서의 적용 키를 해석하는 방법이며, 이 차분 해석에 있어서, 액티브 S박스의 수를 적게 할 수 있으면, 해석이 용이해져, 해석 프로세스수를 삭감할 수 있다.

[0143] 전술한 도 6을 참조한 예에 있어서, 우측으로부터 좌측으로 입력을 행하는 F함수, 즉 제i 라운드와 제i+2 라운드, 제i+4 라운드만을 액티브 S박스 산출 처리 대상 라운드로서 보았을 경우, 액티브 S박스수는, 제i 라운드= 1, 제i+2 라운드= 0, 제i+4 라운드= 1의 합계 불과 2이며, 좌측으로부터 우측으로 입력을 행하는 F함수, 즉 제i+1 라운드 및 제i+3 라운드에서는 액티브 S박스수가 8이지만, 동시 차분 캔슬에 의해 제i+5 라운드에서의 액티브 S박스수가 0으로 되어 버리므로, 차분 해석에 의한 각 S박스의 비선형 변환 처리의 해석, 및 F함수에 대한 입력 라운드 키의 해석 처리가 비교적 용이해진다.

[0144] 도 6을 참조한 예에서는, 제1 S박스(S1)만이 액티브 S박스인 패턴의 발생 상태를 제시했지만, 그 외의 S박스(S2~S8)에 대하여도, 차분 해석의 입력 데이터의 설정에 의해, 각 S박스만을 액티브 S박스로 한 설정이 가능하며, 이와 같은 차분 해석 프로세스를 실행함으로써, 각 S박스의 비선형 변환 처리의 해석, 또한 F함수에 대하여 입력되는 라운드 키의 해석이 가능해진다.

[0145] 도 4 및 도 6을 참조하여, 3 및 5라운드의 경우의 동시 차분 캔슬의 발생예를 설명하였으나, 임의의 라운드의 수로 일반화하여 동시 차분 캔슬을 정의하면 다음과 같이 정의할 수 있다. 도 7을 참조하여, 임의의 라운드의 수에 있어서의 동시 차분 캔슬의 정의에 대하여 설명한다. 그리고, 도 7은, 페이스텔(Feistel) 구조의 공통키 블록 암호를 실행하는 페이스텔 구조의 하나 건너의 라운드(i, i+2, i+4, ···, i+2j)를 나타내고 있다.

[0146] 「정의」

[0147] 페이스텔 구조의 라운드 i에서의 입력 차분의 반(PL 또는 PR)이 0(도 7에 있어서, ΔXi= (00, 00, 00, 00, 00, 00, 00, 00))이며, 거기에 i+2j 라운드(j= 0, 1, 2, . . .)의 F함수의 출력 차분이 배타적 논리합부로 연산되어 가는 과정에 있어서, 어느 라운드 i+2k에 있어서, 배타적 논리합의 결과가 0(도 7에 있어서, ΔXi+2j+1= (00, 00, 00, 00, 00, 00, 00, 00))으로 된 경우를 "동시 차분 캔슬"이라고 한다.

[0148] 그 때, i, i+2, i+4, . ., i+2k 라운드의 F함수 중에 존재하는 액티브 S박스(Active S-box)를 "동시 차분 캔슬을 발생시킨 액티브 S박스"라고 하는 것으로 하고, 벡터 A의 비제로의 요소수를 허밍 웨이트 hw(A)라고 정의하면, 동시 차분 캔슬을 발생시키는 액티브 S박스의 수a는, 이하의 식으로서 나타낼 수 있다.

[0149] [수식 1]

[0150] $$a = \sum_{j=0}^{k} hw(\Delta X_{i+2j})$$

[0151] 전술한 3라운드, 5라운드에서의 예에서는 모두 동시 차분 캔슬을 발생시킨 액티브 S박스수는 2, 즉 a= 2이다.

[0152] 전술한 바와 같이, 공통키 블록 암호에 있어서의 차분 공격에 대한 강도 지표의 하나가 암호화 함수 전체에서의 액티브 S박스의 최소수이며, 액티브 S박스수의 최소수가 클수록 차분 공격에 대한 내성이 높은 것으로 판단된다.

[0153] 그러나, DES 알고리즘과 같이 같은 선형 변환 행렬을 모든 단의 F함수에 사용하는 구성에서는, 도 4, 도 6을 참조하여 설명한 바와 같이, 불과 2개의 액티브 S박스에 의해 동시 차분 캔슬이 발생해 버릴 가능성이 있었다. 그와 같은 성질이 있으므로 라운드의 수의 성장에 대하여 액티브 S박스의 최소수가 충분히 증대하지 않고, 차분 공격에 대한 강도가 그만큼 신장되지 않는 문제가 있었다.

[0154] [2.공통키 블록 암호 알고리즘에 있어서의 선형 해석 처리]

[0155] 차분 해석 처리는, 전술한 바와 같이, 해석의 실행자가 일정한 차분을 가지는 입력 데이터(평문)를 용이하게, 그 대응하는 출력 데이터(암호문)를 해석하는 것이 필요하다. 선형 해석 처리는, 일정한 차분을 가지는 입력 데이터(평문)를 준비할 필요는 없고, 소정량 이상의 입력 데이터(평문)와 대응하는 출력 데이터(암호문)에 따라 해석을 행한다.

[0156] 전술한 바와 같이, 공통키 블록 암호 알고리즘에서는 비선형 변환부로서의 S박스를 가지고, 입력 데이터(평문)와 대응하는 출력 데이터(암호문)와의 선형 관계는 없지만, 선형 해석에서는, 이 S박스의 입출력을 선형 근사(近似)시키고, 다수의 입력 데이터(평문)와 대응하는 출력 데이터(암호문)의 구성 비트값의 선형 관계를 해석하고, 후보로 되는 키를 좁히는 것에 의해 해석을 행한다. 선형 해석에 있어서는, 특정한 차분을 가지는 입력 데이터를 준비하는 것이 필요하지 않고, 다수의 평문과 대응 암호문을 용이하게 함으로써, 해석이 가능해진다.

[0157] [3.본 발명에 따른 암호 처리 알고리즘]

[0158] 이하, 본 발명의 암호 처리 알고리즘에 대하여 설명한다. 본 발명의 암호 처리 알고리즘은, 전술한 선형 해석, 차분 해석 등의 공격에 대한 내성을 향상시킨 구성, 즉 키 해석의 곤란성을 높여 안전성을 향상시킨 구성을 가진다.

[0159] 본 발명에 관한 암호 처리 알고리즘의 하나의 특징은, 종래의 DES 알고리즘과 같이 각 라운드의 F함수에 구성되는 선형 변환부에 공통의 처리(변환 행렬)를 적용한 구성으로 하지 못하고, 복수개의 상이한 정방 MDS 행렬을 설정한 구성으로 한 것이다. 구체적으로는, 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서 상이한 정방 MDS 행렬을 적용한 선형 변환 처리를 실행하는 구성을 가진다.

[0160] 본 발명에 관한 암호 처리 알고리즘은, 정방 MDS 행렬의 성질을 이용하고, 적은 액티브 S박스에 따른 동시 차분 캔슬이 일어나지 않는, 또는 쉽게 일어나지 않는 구조를 실현하고, 액티브 S박스의 최소수를 증대시켜, 차분 공격에 대하여 보다 강한 공통키 블록 암호 처리를 실현한다. 또는, 기존 평문 공격으로서 행해지는 선형 해석에 대한 곤란성도 높인 구성을 가진다.

[0161] 본 발명의 암호 처리 알고리즘은, 도 1, 2를 참조하여 설명한 SPN형의 F함수를 가지는 페이스텔 구조라는 대표적인 공통키 블록 암호의 구조, 즉 비선형 변환부 및 선형 변환부를 가지는 SPN형의 F함수의 복수의 라운드에 건너는 단순한 반복에 의해, 평문을 암호문으로 변환하거나, 또는 암호문을 평문 변환하는 구조를 적용하고 있다.

[0162] 예를 들면, 평문의 길이를 2mn비트(단, m, n는 모두 정수(整數))로서 2mn비트의 평문을, mn비트의 2개의 데이터 PL(Plain-Left), PR(Plain-Right)로 분할하고, 이것을 입력값으로서 각 라운드에 있어서, F함수를 실행시키는 것이며, F함수는, 도 2를 참조하여 설명한 바와 같이, S박스로 이루어지는 비선형 변환부와 선형 변환부를 접속한 SPN형을 가지는 F함수이다.

[0163] 본 발명의 구성에 있어서는, F함수 중의 선형 변환부에 있어서 적용하는 선형 변환 처리를 위한 행렬로서 복수개의 상이한 정방 MDS 행렬로부터 선택된 행렬을 각 라운드의 F함수의 선형 변환부에 있어서 적용하는 행렬로서 설정한다. 구체적으로는, 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서 상이한 정방 MDS 행렬을 적용한다.

[0164] 정방 MDS 행렬에 대하여 설명한다. 정방 MDS 행렬과는 이하의 (a),(b)의 성질을 만족시키는 행렬을 말한다.

[0165] (a) 정방 행렬이다

[0166] (b) 행렬에 포함되는 모든 부분 행렬(submatrix)의 행렬식(determinant)이 0이 아닌, 즉 det(submatrix)≠0

[0167] 상기 (a),(b)의 조건을 만족시키는 행렬을 정방 MDS 행렬이라고 한다.

[0168] 공통키 블록 암호의 각 라운드로 실행하는 F함수에 대한 입출력 비트 길이가 m×n(m, n: 정수)비트이며, F함수 내에 구성되는 비선형 변환부가 n비트의 입출력을 가지는 m개의 S박스에 의해 구성되며, 선형 변환부가 n차의 기약 다항식에 의해 정의되는 2의 확대체 GF(2n) 상의 원(元)을 요소로서 가지는 m차의 정방 행렬에 따른 선형 변환 처리를 실행하는 경우의, 정방 MDS 행렬의 일례를 도 8에 나타낸다. 도 8에 나타내는 정방 MDS 행렬의 예는, n= 8, m= 8의 정방 MDS 행렬의 예이다.

[0169] 상기 (a),(b)를 만족시키는 정방 MDS 행렬은, 벡터 A의 비제로의 요소수를 허밍 웨이트 hw(A)로 하고, M을 m차의 정방 MDS 행렬로 하고, x를 정방 MDS 행렬 M로의 입력 벡터로 한 경우, 이하의 부등식(식 1)을 만족시키는 것으로 된다.

[0170] $hw(x+hw(Mx)\geq m+1$ ·····(식 1)

[0171] 상기 식(식 1)은, 정방 MDS 행렬(M)에 의해 선형 변환되는 입력 데이터 x의 비제로의 요소수 hw(x)와 정방 MDS 행렬(M)에 의해 선형 변환된 출력 데이터 Mx의 비제로의 요소수 hw(Mx)의 총수가, 정방 MDS 행렬의 차수 m보다 커지는 것을 의미하고 있다.

[0172] 그리고, 정방 MDS 행렬 라는 이름은 정방 MDS-code(Maximum Distance Separable Code)의 생성 행렬의 표준형의 우측 반이 상기 조건을 만족시키고 있으므로 명명하고 있는 것이다.

[0173] 1개의 행렬을 모든 F함수에 내장한다는 종래의 구성이라도 선형 변환 행렬에 정방 MDS 행렬을 사용함으로써, 정방 MDS 행렬이 아닌 행렬을 사용하는 경우에 비해 액티브 S박스수의 최소수를 비교적 고수준으로 유지할 수 있다는 것은 알려져 있다.

[0174] 본 발명에서는, 각 라운드의 F함수에는 정방 MDS 행렬의 조건을 만족시키는 행렬을 이용하고, 또한 라운드마다 상이한 행렬을 설정하는 방법을 제안한다. 구체적으로는, 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서 상이한 정방 MDS 행렬을 적용한다.

[0175] 이하에, 단수(라운드의 수)가 2r(r은 정수(整數))의 페이스텔형 공통키 블록 암호에 있어서, 차분 공격에 대한 내성을 보다 높인 복수개의 구성예에 대하여 설명한다.

[0176] 그리고, 이하의 설명에 있어서, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성의 j단째의 F함수에 있어서의 선형 변환부에서 적용하는 선형 변환 행렬을 MLTj로서 나타내는 것으로 한다.

[0177] 본 발명의 구성에서는, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성에 있어서의 각 단의 F함수 중의 선형 변환부에 있어서 적용하는 선형 변환 처리를 위한 행렬로서 복수개의 상이한 정방 MDS 행렬로부터 선택된 행렬을 각 라운드의 F함수의 선형 변환부에 있어서 적용하는 행렬로서 설정한다. 구체적으로는, 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서 상이한 정방 MDS 행렬을 적용한다.

[0178] 구체적으로는, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성에 대응하여, r 이하의 q개의 정방 MDS 행렬: L1, L2, ··, Lq를 생성하고, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성에 있어서의 홀수단째의 F함수 중의 선형 변환부에 있어서 적용하는 선형 변환 처리를 위한 행렬로서 상위단의 F함수로부터 차례로 L1, L2, ··, Lq, L1, L2··로 하여, q개의 정방 MDS 행렬을 반복 설정한다. 또한, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, ··, Lq, L1, L2··로 하여 q개의 정방 MDS 행렬을 반복 설정한다.

[0179] 본 설정을 적용한 구성예를 도 9에 나타낸다. 도 9는, 단수(라운드의 수)가 2r= 12, 즉 r= 6의 페이스텔형 공통키 블록 암호 처리 구성으로 한 경우, q= 3, 즉 12단의 라운드의 수를 가지는 페이스텔형 공통키 블록 암호 처리 구성에 있어서 3종류의 상이한 정방 MDS 행렬을 배치한 구성예로서, 각 라운드의 F함수부의 선형 변환부로 설정하는 정방 MDS 행렬(L1, L2, L3)을 나타내고 있다.

[0180] 도 9의 구성은, 2mn비트의 평문을, mn비트의 2개의 데이터 PL(Plain-Left), PR(Plain-Right)로 분할하고, 이것을 입력값으로서 각 라운드에 있어서, F함수를 실행시키는 구성이며, 제1 라운드의 F함수(401) 및 그 외의 라운드의 F함수도, 모두 도 2를 참조하여 설명한 바와 같이, S박스로 이루어지는 비선형 변환부와 선형 변환부를 접속한 SPN형을 가지는 F함수이다.

[0181] 도 9의 설정예는 r= 6, q= 3이며, 각 F함수 내에 나타내는 기호 Ln은 정방 MDS 행렬(402)을 나타내고 있다. 즉 L1, L2, L3은, 각각 상이한 3종류의 정방 MDS 행렬을 나타내고, 각 F함수의 선형변부에 있어서 선형 변환 처리

에 적용하는 정방 MDS 행렬을 나타내고 있다.

[0182] 선형 변환 행렬 MLTj의 설정 처리 시켄스에 대하여, 도 10을 참조하여 설명한다.

[0183] [스텝 S21]

[0184] 라운드의 수가 2r인 반수 r에 대하여 r 이하의 수q, 즉

[0185] q≤r로 되는 수q를 선택한다. 단, q는 2 이상의 정수이다.

[0186] [스텝 S22]

[0187] q개의 GF(2^n) 상의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를 생성한다. q개의 GF(2^n) 상의 m차 정방 MDS 행렬 L1, L2, . . ., Lq의 생성 처리 방법에 대한 자세한 것은, 후단에서 설명한다.

[0188] 스텝 S22에 있어서, q개의 GF(2n) 상의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가 생성한 후, 다음에, 이하의 정방 MDS 행렬 설정 처리를 실행한다.

[0189] [스텝 S23]

[0190] (2i-1(1≤i≤r)단째의 선형 변환 행렬 $MLT2_{i-1}$에 $L_{(i-1 \bmod q)+1}$을 설정한다.

[0191] [스텝 S24]

[0192] (2i)(1≤i≤r)단째의 선형 변환 행렬에 MLT_{2i}에 $MLT_{2r-2i+1}$을 설정한다.

[0193] 예를 들면, 도 9에 나타낸 구성예, 즉 12단(r= 6)이며 q= 3으로 한 경우에는,

[0194] MLT1=L1, MLT2=L3

[0195] MLT3=L2, MLT4=L2

[0196] MLT5=L3, MLT6=L1

[0197] MLT7=L1, MLT8=L3

[0198] MLT9=L2, MLT10=L2

[0199] MLT11=L3, MLT12=L1

[0200] 의 설정으로 된다.

[0201] 이와 같이, 본 발명의 암호 처리 장치에 있어서는, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성에 대응하여, r 이하의 q개의 정방 MDS 행렬: L1, L2, ・・, Lq를 생성하고, 홀수단째에 대하여는 상위 단의 F함수로부터 차례로 L1, L2, ・・, Lq, L1, L2・・로 하여, q개의 정방 MDS 행렬을 반복 설정하고, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, ・・, Lq, L1, L2・・로 하여 q개의 정방 MDS 행렬을 반복 설정하는 구성으로 하고 있다.

[0202] 다음에, 도 10의 처리 플로에 있어서의 스텝 S22의 q개의 GF(2n) 상의 m차 정방 MDS 행렬 L1, L2, . . ., Lq의 생성 처리 및 F함수로의 설정 구성의 상세에 대하여 설명한다. 그리고, 설명은 이하의 항목에 따라 행한다.

[0203] (3-a) 차분 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예

[0204] (3-b) 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예

[0205] (3-c) 차분 공격 및 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예

[0206] [(3-a) 차분 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예]

[0207] 먼저, 차분 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예로서 3개의 처리예 a1, a2, a3에 대하여 설명한다.

[0208] (처리예 a1)

[0209] 차분 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예의 제1 예에 대하여 설명한다. 먼저, 도 11에 나타낸 플로차트를 참조하여 정방 MDS 행렬의 생성 처리에 대하여 설명한다.

[0210] [스텝 S101]

[0211] 입력: 필요한 정방 MDS의 개수 q, 확대 차수: n, 행렬의 사이즈: m로서

[0212] $GF(2^n)$ 상에서, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를 랜덤으로 생성한다. 그리고, 도 11에 나타낸 플로에서는, MDS의 개수 q= 6, 확대 차수: n= 8, 행렬의 사이즈: m= 8의 경우의 처리예로서 나타내고 있다.

[0213] [스텝 S102]

[0214] q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq에 포함되는 m개 열의 임의의 m개를 인출했을 때, 선형 독립으로 되어 있는지 여부를 체크한다. 체크에 통과하면 스텝 S103으로 진행하고, 그렇지 않은 경우에는 스텝 S101로 돌아온다.

[0215] [스텝 S103]

[0216] q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를, 라운드의 수가 2r인 페이스텔형 공통키 블록 암호에 적용하는 정방 MDS 행렬로서 출력한다.

[0217] 이상의 프로세스에 의해, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가 생성된다. 그리고, q≤r이다.

[0218] 이같이 하여 생성한 q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를, 먼저, 도 10을 참조하여 설명한, [스텝 S23], [스텝 S24]의 처리에 따라, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성의 각 단의 F함수부의 선형 변환부의 선형 변환 처리에 적용하는 행렬로서 설정한다. 즉 홀수단째에 대하여는 상위단으로부터 차례로 L1, L2, ・・, Lq, L1, L2・・로 하여, q개의 정방 MDS 행렬을 반복 설정하고, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, ・・, Lq, L1, L2・・로 하여, q개의 정방 MDS 행렬을 반복 설정한다.

[0219] 이와 같이, 짝수 라운드의 정방 MDS 행렬과 홀수 라운드의 정방 MDS 행렬을 서로 역순으로 배치함으로써, 암호화 처리와 복호 처리는 키의 순서를 교체하는 처리를 제외하고 동일한 것이 보증된다.

[0220] 본 구성에 의해,

[0221] (a) 각 F함수의 선형 변환 행렬은 정방 MDS인 것,

[0222] (b) 암호화 함수 내의 홀수 라운드 내의 적어도 연속되는 q개의 F함수에 포함되는 선형 변환 행렬의 임의의 m개의 열 벡터가 독립인 것,

[0223] (c) 짝수 라운드 내의 적어도 연속되는 q개의 F함수에 포함되는 선형 변환 행렬의 임의의 m개의 열 벡터가 독립인 것,

[0224] 이들 (a)~(c)가 보증되므로, 복수개 단의 라운드의 수를 가지는 페이스텔형 공통키 블록 암호 처리 구성에 있어서, 연속되는 2q-1 라운드에 있어서, m개 이하의 액티브 S박스의 기여에 의한 동시 차분 캔슬은 발생하지 않는 것이 보증된다. 따라서, 암호화 함수 전체의 액티브 S박스수의 최소값이 증대한다.

[0225] 이와 같이, 본 처리예에 의해, 공통키 블록 암호에 있어서의 차분 공격에 대한 강도 지표의 하나인 암호화 함수 전체에서의 액티브 S박스의 최소수를 크게 하는 것이 가능해지고, 결과로서, 차분 해석(차분 공격)을 행한 경우의 액티브 S박스의 수가 증대하고, 해석의 곤란성이 높아지게 된다. 따라서, 키의 해석이 곤란한, 안정성이 높은 암호 처리가 실현된다.

[0226] (처리예 a2)

[0227] 차분 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예의 제2 예에 대하여 설명한다. 도 12의 플로차트를 참조하여 정방 MDS 행렬의 생성 처리에 대하여 설명한다.

[0228] [스텝 S201]

[0229] 입력: 필요한 MDS의 개수 q, 확대 차수: n, 행렬의 사이즈: m로서

[0230] $GF(2^n)$ 상에서, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를 랜덤으로 생성한다. 그리고, 도 12에 나타낸 플로에서는, MDS의 개수 q= 6, 확대 차수: n= 8, 행렬의 사이즈: m= 8의 경우의 처리예로서 나타내고 있다.

[0231] [스텝 S202]

[0232] q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq에 포함되는 qm개 열의 임의의 m개를 인출했을 때, 정방 MDS 행렬로 되어 있는지 여부를 체크한다. 체크에 통과하면 스텝 S203으로 진행하고, 그렇지 않은 경우에는 스텝 S201로 돌아온다.

[0233] 그리고, 정방 MDS 행렬이란, 전술한 바와 같이 이하의 성질을 만족시키는 행렬을 말한다.

[0234] (a) 정방 행렬이다

[0235] (b) 행렬에 포함되는 모든 부분 행렬(submatrix)의 행렬식(determinant)이 0이 아닌, 즉 det(submatrix)≠0

[0236] [스텝 S203]

[0237] q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를, 라운드의 수가 2r인 페이스텔형 공통키 블록 암호에 적용하는 정방 MDS 행렬로서 출력한다.

[0238] 이상의 프로세스에 의해, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가 생성된다. 그리고, q≤r이다.

[0239] 전술한 처리예 a1에 있어서의 정방 MDS 행렬 생성 처리에 있어서는, 도 11의 처리 시켄스에 있어서 설명한 바와 같이, 스텝 S102에 있어서, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq에 포함되는 qm개 열의 임의의 m개를 인출했을 때의 선형 독립성을 판정하였으나, 이 처리예 a2에 있어서의 정방 MDS 행렬 생성 처리에 있어서는, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq에 포함되는 qm개 열의 임의의 m개를 인출했을 때 정방 MDS 행렬로 되어 있는지 여부를 체크한다. 즉, 보다 엄격한 체크가 실행되게 된다.

[0240] 이 도 12에 나타낸 처리 시켄스에 따른 정방 MDS 행렬 생성 처리에 의해 생성된 q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가, 먼저 설명한 처리예 a1와 마찬가지로, 먼저, 도 10을 참조하여 설명한, [스텝 S23], [스텝 S24] 의 처리에 따라, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성의 각 단의 F함수부의 선형 변환부의 선형 변환 처리에 적용하는 행렬로서 설정된다. 즉 홀수단째에 대하여는 상위단으로부터 차례로 L1, L2, ··, Lq, L1, L2··로 하여, q개의 정방 MDS 행렬을 반복 설정하고, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, ··, Lq, L1, L2··로 하여, q개의 정방 MDS 행렬을 반복 설정한다.

[0241] 이와 같이, 짝수 라운드의 정방 MDS 행렬과 홀수 라운드의 정방 MDS 행렬을 서로 역순으로 배치함으로써, 암호화 처리와 복호 처리는 키의 순서를 교체하는 처리를 제외하고 동일한 것이 보증된다.

[0242] 본 구성에 의해,

[0243] (a) 각 F함수의 선형 변환 행렬은 정방 MDS인 것,

[0244] (b) 암호화 함수 내의 홀수 라운드 내의 적어도 연속되는 q개의 F함수에 포함되는 선형 변환 행렬의 임의의 m개의 열 벡터가 정방 MDS 행렬인 것,

[0245] (c) 짝수 라운드 내의 적어도 연속되는 q개의 F함수에 포함되는 선형 변환 행렬의 임의의 m개의 열 벡터가 정방 MDS 행렬인 것,

[0246] 이들 (a)~(c)가 보증되므로, 복수개 단의 라운드의 수를 가지는 페이스텔형 공통키 블록 암호 처리 구성에 있어서, 연속되는 2q-1 라운드에 있어서, m개 이하의 액티브 S박스의 기여에 의한 동시 차분 캔슬은 발생하지 않는 것이 보증된다.

[0247] 또한,

[0248] (d) 정방 MDS의 성질로부터, a개(a≤m의 액티브 S박스의 기여에 의해 얻어지는 차분값에 있어서의 비제로의 요소수는 m+1-a개 이상으로 되는 것이 보증된다. 따라서, 암호화 함수 전체의 액티브 S박스수의 최소값이 증대한다.

[0249] 이와 같이, 본 처리예에 의해, 공통키 블록 암호에 있어서의 차분 공격에 대한 강도 지표의 하나인 암호화 함수 전체에서의 액티브 S박스의 최소수를 크게 하는 것이 가능해지고, 결과로서, 차분 해석(차분 공격)을 행한 경우의 액티브 S박스의 수가 증대하고, 해석의 곤란성이 높아지게 된다. 따라서, 키의 해석이 곤란한, 안정성이 높은 암호 처리가 실현된다.

[0250] (처리예 a3)

[0251] 차분 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예의 제3 예에 대하여 설명한다.

도 13의 플로차트를 참조하여 정방 MDS 행렬의 생성 처리에 대하여 설명한다.

[0252] [스텝 S301]

[0253] 입력: 필요한 MDS의 개수 q, 확대 차수: n, 행렬의 사이즈: m로서

[0254] $GF(2^n)$ 상에서, 1개의 qm차 정방 MDS 행렬 M를 생성한다. 그리고, 도 13에 나타낸 플로에서는, MDS의 개수 q= 6, 확대 차수: n= 8, 행렬의 사이즈: m= 8의 경우의 처리예로서 나타내고 있다.

[0255] [스텝 S302]

[0256] 1개의 qm차 정방 MDS 행렬 M로부터 m개의 행을 임의로 선택 추출하고, m행, qm열의 행렬 M'를 구성한다.

[0257] [스텝 S303]

[0258] m행, qm열의 행렬 M'에 포함되는 qm개의 열 벡터를 중복되지 않고 m개의 열 벡터로 이루어지는 q개의 그룹에 임의로 분할하고, 각각의 그룹에 포함되는 열 벡터로부터 m차의 정방 행렬 L1, L2, . . ., Lq를, 라운드의 수가 2r인 페이스텔형 공통키 블록 암호에 적용하는 정방 MDS 행렬로서 출력한다.

[0259] 이상의 프로세스에 의해, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가 생성된다. 그리고, q≤r이다.

[0260] 처리예 a3에 있어서의 정방 MDS 행렬 생성 방법 3에 대하여, 도 14를 참조하여, 보다 구체적으로 설명한다.

[0261] [스텝 S301]

[0262] $GF(2^n)$ 상에서, 1개의 qm차 정방 MDS 행렬 M를 생성한다. 도 14에 나타낸 바와 같이, qm×qm의 정방 MDS 행렬 M를 생성한다. 그리고, 이 스텝 S301에 있어서 생성하는 행렬 M의 차수는 qm 차보다 큰 것이라도 된다.

[0263] [스텝 S302]

[0264] 도 14에 나타낸 바와 같이, qm차 정방 MDS 행렬 M로부터 m개의 행을 임의로 선택 추출하고, m행, qm열의 행렬 M'를 구성한다. 그리고, 도면에 나타낸 예에서는, 연속되는 m개의 행을 선택 추출한 예로서 나타내고 있지만, m차 정방 MDS 행렬 M를 구성하는 임의의 이격된 행을 m개 선택 추출하여, m행, qm열의 행렬 M'를 구성해도 된다.

[0265] [스텝 S303]

[0266] m행, qm열의 행렬 M'에 포함되는 qm개의 열 벡터를 중복되지 않고 m개의 열 벡터로 이루어지는 x개의 그룹으로 임의로 분할하고, 각각의 그룹에 포함되는 열 벡터로부터 m차의 정방 행렬 L1, L2, . . ., Lx를 생성한다.

[0267] 도 13, 도 14를 참조하여 설명한 처리 시켄스에 따른 정방 MDS 행렬 생성 처리에 의해 생성된 q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가, 먼저 설명한 처리예 a1, a2와 마찬가지로, 먼저, 도 10을 참조하여 설명한, [스텝 S23] , [스텝 S24] 의 처리에 따라, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성의 각 단의 F함수부의 선형 변환부의 선형 변환 처리에 적용하는 행렬로서 설정된다. 즉 홀수단째에 대하여는 상위단으로부터 차례로 L1, L2, ・・, Lq, L1, L2・・로 하여, q개의 정방 MDS 행렬을 반복 설정하고, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, ・・, Lq, L1, L2・・로 하여, q개의 정방 MDS 행렬을 반복 설정한다.

[0268] 이와 같이, 짝수 라운드의 정방 MDS 행렬과 홀수 라운드의 정방 MDS 행렬을 서로 역순으로 배치함으로써, 암호화 처리와 복호 처리는 키의 순서를 교체하는 처리를 제외하고 동일한 것이 보증된다.

[0269] 본 구성에 의해,

[0270] (a) 각 F함수의 선형 변환 행렬은 정방 MDS인 것,

[0271] (b) 암호화 함수 내의 홀수 라운드 내의 적어도 연속되는 q개의 F함수에 포함되는 선형 변환 행렬의 임의의 m개의 열 벡터가 독립인 것,

[0272] (c) 짝수 라운드 내의 적어도 연속되는 q개의 F함수에 포함되는 선형 변환 행렬의 임의의 m개의 열 벡터가 독립인 것,

[0273] 이들 (a)~(c)가 보증되므로, 복수개 단의 라운드의 수를 가지는 페이스텔형 공통키 블록 암호 처리 구성에 있어서, 연속되는 2q-1 라운드에 있어서, m개 이하의 액티브 S박스의 기여에 의한 동시 차분 캔슬은 발생하지 않는

것이 보증된다.

[0274] 또한,

[0275] (d) 정방 MDS의 성질로부터, a개(a≤m의 액티브 S박스의 기여에 의해 얻어지는 차분값에 있어서의 비제로의 요소수는 m+1-a개 이상으로 되는 것이 보증된다. 따라서, 암호화 함수 전체의 액티브 S박스수의 최소값이 증대한다.

[0276] 그리고, 처리예 a3가 특히 효과를 발휘하는 것은, m, r,이 커지고, 전술한 처리예 a1, a2의 행렬 결정 처리 방식에 관한 시간 목표 비용이 막대하게 되고, 현실적인 시간 내에 행렬을 결정하는 것이 곤란한 경우이다. 그와 같은 경우라도 본 처리예 a3의 정방 MDS 행렬 생성 방법이라면 비교적 단시간에서의 행렬 생성 처리가 가능해진다.

[0277] 이것은, 처리예 a3에 있어서는, 큰 m, r에 대하여도 현실적인 시간에 충분히 처리 가능한 방식, 예를 들면, 리드 솔로몬(Reed-Solomon) 부호의 생성 행렬의 생성법을 적용하는 것이 가능해지기 때문이다.

[0278] 이 처리예 a3에 있어서도, 전술한 바와 같이, 공통키 블록 암호에 있어서의 차분 공격에 대한 강도 지표의 하나인 암호화 함수 전체에서의 액티브 S박스의 최소수를 크게 하는 것이 가능해져, 결과로서, 차분 해석(차분 공격)을 행한 경우의 액티브 S박스의 수가 증대하고, 해석의 곤란성이 높아지게 된다. 따라서, 키의 해석이 곤란한 안정성이 높은 암호 처리가 실현된다.

[0279] [(3-b) 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예]

[0280] 다음에, 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예로서 2개의 처리예 b1, b2에 대하여 설명한다.

[0281] (처리예 b1)

[0282] 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예의 제1 예에 대하여, 설명한다. 도 15에 나타낸 플로차트를 참조하여 정방 MDS 행렬의 생성 처리에 대하여 설명한다.

[0283] [스텝 S401]

[0284] 입력: 필요한 정방 MDS의 개수 q, 확대 차수: n, 행렬의 사이즈: m로서

[0285] $GF(2^n)$ 상에서, q개의 m차 정방 MDS 행렬 M1, M2, . . ., Mq를 랜덤으로 생성한다. 그리고, 도 14에 나타낸 플로에서는, 정방 MDS의 개수 q= 6, 확대 차수: n= 8, 행렬의 사이즈: m= 8의 경우의 처리예로서 나타내고 있다.

[0286] [스텝 S402]

[0287] q개의 m차 정방 MDS 행렬 M1, M2, . . ., Mq의 역행렬M1-1, M (2-1, . . ., Mq^{-1}을 산출하고, 인접하는 2개의 역행렬에 포함되는 2m의 행 벡터로부터 임의의 m개의 행 벡터를 인출했을 때, 선형 독립적으로 되어 있는지 여부를 체크한다. 도 15 중, ^tR은, 행 벡터의 전치(轉置) 벡터를 나타내는 것이다. 체크에 통과하면 스텝 S403으로 진행한다, 그렇지 않은 경우에는 스텝 S401로 돌아온다. 단, $M1^{-1}$과 Mq^{-1}은, 인접하는 행렬로 한다.

[0288] [스텝 S403]

[0289] q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를, 라운드의 수가 2r인 페이스텔형 공통키 블록 암호에 적용하는 정방 MDS 행렬로서 출력한다.

[0290] 이상의 프로세스에 의해, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가 생성된다. 그리고, q≤r이다.

[0291] 이같이 하여 생성한 q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를, 먼저, 도 10을 참조하여 설명한, [스텝 S23], [스텝 S24]의 처리에 따라, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성의 각 단의 F함수부의 선형 변환부의 선형 변환 처리에 적용하는 행렬로서 설정한다. 즉 홀수단째에 대하여는 상위단으로부터 차례로 L1, L2, · ·, Lq, L1, L2 · ·로 하여, q개의 정방 MDS 행렬을 반복 설정하고, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, · ·, Lq, L1, L2 · ·로 하여, q개의 정방 MDS 행렬을 반복 설정한다.

[0292] 이와 같이, 짝수 라운드의 정방 MDS 행렬과 홀수 라운드의 정방 MDS 행렬을 서로 역순으로 배치함으로써, 암호화 처리와 복호 처리는 키의 순서를 교체하는 처리를 제외하고 동일한 것이 보증된다.

[0293] 본 구성에 의해,

[0294] (a) 각 F함수의 선형 변환 행렬은 정방 MDS인 것,

[0295] (b) 암호화 함수 내의 홀수 라운드 내에 연속하여 포함되는 선형 변환 행렬, 및 짝수 라운드 내에 연속하여 포함되는 선형 변환 행렬의 역행렬의 임의의 m개의 열 벡터는 독립인 것

[0296] 이 보증된다. 이로써, 선형 공격에 있어서의 선형 근사에 의한 해석 곤란성을 높이는 것이 가능해지므로, 해석의 곤란성, 즉 키의 해석이 곤란한 안정성이 높은 암호 처리가 실현된다.

[0297] (처리예 b2)

[0298] 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예의 제2 예에 대하여, 설명한다. 도 16에 나타낸 플로차트를 참조하여 정방 MDS 행렬의 생성 처리에 대하여 설명한다.

[0299] [스텝 S501]

[0300] 입력: 필요한 정방 MDS의 개수 q, 확대 차수: n, 행렬의 사이즈: m으로 하여,

[0301] GF(2^n) 상에서, q개의 m차 정방 MDS 행렬 M1, M2, . . ., Mq를 랜덤으로 생성한다. 그리고, 도 16에 나타낸 플로에서는, 정방 MDS의 개수 q= 6, 확대 차수: n= 8, 행렬의 사이즈: m= 8의 경우의 처리예로서 나타내고 있다.

[0302] [스텝 S502]

[0303] q개의 m차 정방 MDS 행렬 M1, M2, . . ., Mq의 역행렬$M1^{-1}$, $M2^{-1}$, . . ., Mq^{-1}을 산출하고, 인접하는 2개의 역행렬에 포함되는 2m의 행 벡터로부터 임의의 m개의 행 벡터를 인출했을 때, 정방 MDS 행렬로 되어 있는지 여부를 체크한다. 도 16 중, ^tR은, 행 벡터의 전치 벡터를 나타내는 것이다. 체크에 통과하면 스텝 S503으로 진행한다, 그렇지 않은 경우에는 스텝 S401로 돌아온다. 단, M^{1-1}과 Mq^{-1}은, 인접하는 행렬로 한다.

[0304] 그리고, 정방 MDS 행렬이란, 전술한 바와 같이 이하의 성질을 만족시키는 행렬을 말한다.

[0305] (a) 정방 행렬이다

[0306] (b) 행렬에 포함되는 모든 부분 행렬(submatrix)의 행렬식(determinant)이 0이 아닌, 즉 det(submatrix)≠0

[0307] [스텝 S503]

[0308] q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를, 라운드의 수가 2r인 페이스텔형 공통키 블록 암호에 적용하는 정방 MDS 행렬로서 출력한다.

[0309] 이상의 프로세스에 의해, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가 생성된다. 그리고, q≤r이다.

[0310] 전술한 처리예 b1에 있어서의 정방 MDS 행렬 생성 처리에 있어서는, 도 15의 처리 시켄스에 있어서 설명한 바와 같이, 스텝 S402에 있어서, q개의 m차 정방 MDS 행렬의 M1, M2, . . ., Mq의 역행렬M1-1, M (2-1, . . ., Mq-1에 포함되는 qm개 열의 임의의 m개를 인출했을 때의 선형 독립성을 판정하였으나, 이 처리예 b2에 있어서의 정방 MDS 행렬 생성 처리에 있어서는, q개의 m차 정방 MDS 행렬의 M1, M2, . . ., Mq의 역행렬$M1^{-1}$, $M2^{-1}$, . . ., Mq^{-1}에 포함되는 qm개 열의 임의의 m개를 인출했을 때 정방 MDS 행렬로 되어 있는지 여부를 체크한다. 즉, 보다 어려운 체크가 실행되게 된다.

[0311] 이 도 16에 나타낸 처리 시켄스에 따른 정방 MDS 행렬 생성 처리에 의해 생성된 q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가, 먼저 설명한 처리예 b1와 마찬가지로, 먼저, 도 10을 참조하여 설명한, [스텝 S23], [스텝 S24] 의 처리에 따라, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성의 각 단의 F함수부의 선형 변환부의 선형 변환 처리에 적용하는 행렬로서 설정된다. 즉 홀수단째에 대하여는 상위단으로부터 차례로 L1, L2, ・・, Lq, L1, L2・・로 하여 q개의 정방 MDS 행렬을 반복 설정하고, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, ・・, Lq, L1, L2・・로 하여 q개의 정방 MDS 행렬을 반복 설정한다.

[0312] 이와 같이, 짝수 라운드의 정방 MDS 행렬과 홀수 라운드의 정방 MDS 행렬을 서로 역순으로 배치함으로써, 암호화 처리와 복호 처리는 키의 순서를 교체하는 처리를 제외하고 동일한 것이 보증된다.

[0313] 본 구성에 의해,

[0314] (a) 각 F함수의 선형 변환 행렬은 정방 MDS인 것,

[0315] (b) 암호화 함수 내의 홀수 라운드 내에 연속하여 포함되는 선형 변환 행렬, 및 짝수 라운드 내에 연속하여 포함되는 선형 변환 행렬의 역행렬의 임의의 m개의 열 벡터가 정방 MDS 행렬로 되는 것

[0316] 이 보증된다. 이로써, 선형 공격에 있어서의 선형 근사에 의한 해석 곤란성을 높이는 것이 가능해져, 해석의 곤란성, 즉 키의 해석이 곤란한 안정성이 높은 암호 처리가 실현된다.

[0317] [(3-c) 차분 공격 및 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예]

[0318] 다음에, 차분 공격 및 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬의 생성 및 F함수로의 설정예에 대하여 설명한다.

[0319] 차분 공격에 대한 내성을 가지는 암호 처리 알고리즘은, 먼저, 도 10~도 13을 참조하여 설명한 처리, 즉 F함수의 선형 처리부에 있어서의 선형 변환에 적용하는 정방 MDS 행렬을 전술한 처리예 a1(도 11)~a3(도 13) 중 어느 하나의 처리를 적용하여 설정됨으로써 실현된다. 또, 선형 공격에 대한 내성을 가지는 암호 처리 알고리즘은, 먼저, 도 10, 및 도 14, 도 15를 참조하여 설명한 처리, 즉 F함수의 선형 처리부에 있어서의 선형 변환에 적용하는 정방 MDS 행렬을 전술한 처리예 b1(도 14), b2(도 15) 중 어느 하나의 처리를 적용하여 설정됨으로써 실현된다.

[0320] 차분 공격 및 선형 공격에 대한 내성 향상을 실현한 정방 MDS 행렬은,

[0321] 처리예 a1(도 11)~a3(도 13) 중 어느 하나의 처리와,

[0322] 처리예 b1(도 14), b2(도 15) 중 어느 하나의 처리를

[0323] 병행하여 실행하여 생성한 정방 MDS 행렬을, 도 10에 있어서 설명한 [스텝 S23] , [스텝 S24] 의 처리에 따라, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성의 각 단의 F함수부의 선형 변환부의 선형 변환 처리에 적용하는 행렬로서 설정됨으로써 실현된다.

[0324] 즉,

[0325] 처리예 a1와 처리예 b1,

[0326] 처리예 a1와 처리예 b2,

[0327] 처리예 a2와 처리예 b1,

[0328] 처리예 a2와 처리예 b2,

[0329] 처리예 a3와 처리예 b1,

[0330] 처리예 a3와 처리예 b2,

[0331] 중 어느 하나의 조합에 의해, q개의 정방 MDS 행렬을 생성하고, 2r의 페이스텔형 공통키 블록 암호 처리 구성의 각 단의 F함수부의 선형 변환부의 선형 변환 처리에 적용하는 행렬로서 설정한다. 홀수단째에 대하여는 상위단으로부터 차례로 L1, L2, ・・, Lq, L1, L2・・로 하여 q개의 정방 MDS 행렬을 반복 설정하고, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, ・・, Lq, L1, L2・・로 하여 q개의 정방 MDS 행렬을 반복 설정한다. 이 설정에 의해, 차분 공격 및 선형 공격에 대한 내성 향상을 실현한 암호 처리가 가능해진다.

[0332] 도 17을 참조하여, 차분 공격 및 선형 공격에 대한 내성 향상을 실현한 암호 처리를 실현하기 위한 정방 MDS 행렬의 생성 처리의 일례에 대하여 설명한다. 이 처리는, 전술한 처리예 a2와 처리예 b2와의 조합이다.

[0333] [스텝 S601]

[0334] 입력: 필요한 정방 MDS의 개수 q, 확대 차수: n, 행렬의 사이즈: m로서

[0335] GF(2n) 상에서, q개의 m차 정방 MDS 행렬 M1, M2, . . ., Mq를 랜덤으로 생성한다. 그리고, 도 17에 나타낸 플로에서는, 정방 MDS의 개수 q= 6, 확대 차수: n= 8, 행렬의 사이즈: m= 8의 경우의 처리예로서 나타내고

있다.

[0336] [스텝 S602]

[0337] q개의 m차 정방 MDS 행렬 M1, M2, . . ., Mq에 포함되는 qm개 열의 임의의 m개를 인출했을 때, 정방 MDS 행렬로 되어 있는지 여부를 체크한다. 체크에 통과하면 스텝 S603으로 진행한다, 그렇지 않은 경우에는 스텝 S601로 돌아온다.

[0338] 그리고, 정방 MDS 행렬이란, 전술한 바와 같이 이하의 성질을 만족시키는 행렬을 말한다.

[0339] (a) 정방 행렬이다

[0340] (b) 행렬에 포함되는 모든 부분 행렬(submatrix)의 행렬식(determinant)이 0이 아닌, 즉 det(submatrix)≠0

[0341] [스텝 S603]

[0342] q개의 m차 정방 MDS 행렬 M1, M2, . . ., Mq의 역행렬M^{1-1}, M^{2-1}, . . ., M^{q-1}을 산출하고, 인접하는 2개의 역행렬에 포함되는 2m의 행 벡터로부터 임의의 m개의 행 벡터를 인출했을 때, 정방 MDS 행렬로 되어 있는지 여부를 체크한다. 도 17 중, tR은, 행 벡터의 전치 벡터를 나타내는 것이다. 체크에 통과하면 스텝 S604로 진행한다, 그렇지 않은 경우에는 스텝 S601로 돌아온다. 단, M^{1-1}과 Mq^{-1}은, 인접하는 행렬로 한다.

[0343] [스텝 S604]

[0344] q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq를, 라운드의 수가 2r인 페이스텔형 공통키 블록 암호에 적용하는 정방 MDS 행렬로서 출력한다.

[0345] 이상의 프로세스에 의해, q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가 생성된다. 그리고, q≤r이다.

[0346] 이 도 17에 나타낸 처리 시켄스에 따른 정방 MDS 행렬 생성 처리에 의해 생성된 q개의 m차 정방 MDS 행렬 L1, L2, . . ., Lq가, 먼저, 도 10을 참조하여 설명한, [스텝 S23], [스텝 S24] 의 처리에 따라, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성의 각 단의 F함수부의 선형 변환부의 선형 변환 처리에 적용하는 행렬로서 설정된다. 즉 홀수단째에 대하여는 상위단으로부터 차례로 L1, L2, ··, Lq, L1, L2··로 하여 q개의 정방 MDS 행렬을 반복 설정하고, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, ··, Lq, L1, L2··로 하여 q개의 정방 MDS 행렬을 반복 설정한다.

[0347] 이와 같이, 짝수 라운드의 정방 MDS 행렬과 홀수 라운드의 정방 MDS 행렬을 서로 역순으로 배치함으로써, 암호화 처리와 복호 처리는 키의 순서를 교체하는 처리를 제외하고 동일한 것이 보증된다.

[0348] 본 구성에 의해,

[0349] (a) 각 F함수의 선형 변환 행렬은 정방 MDS인 것,

[0350] (b) 암호화 함수 내의 홀수 라운드 내의 적어도 연속되는 q개의 F함수에 포함되는 선형 변환 행렬의 임의의 m개의 열 벡터가 정방 MDS 행렬인 것,

[0351] (c)짝수 라운드 내의 적어도 연속되는 q개의 F함수에 포함되는 선형 변환 행렬의 임의의 m개의 열 벡터가 정방 MDS 행렬인 것,

[0352] 이들 (a)~(c)가 보증되므로, 복수개 단의 라운드의 수를 가지는 페이스텔형 공통키 블록 암호 처리 구성에 있어서, 연속되는 $2q^{-1}$ 라운드에 있어서, m개 이하의 액티브 S박스의 기여에 의한 동시 차분 캔슬은 발생하지 않는 것이 보증된다.

[0353] 또한,

[0354] (d) 정방 MDS의 성질로부터, a개(a≤m의 액티브 S박스의 기여에 의해 얻어지는 차분값에 있어서의 비제로의 요소수는 m+1-a개 이상으로 되는 것이 보증된다. 따라서, 암호화 함수 전체의 액티브 S박스수의 최소값이 증대한다.

[0355] 또한,

[0356] (e) 암호화 함수 내의 홀수 라운드 내에 연속하여 포함되는 선형 변환 행렬, 및 짝수 라운드 내에 연속하여 포함되는 선형 변환 행렬의 역행렬의 임의의 m개의 열 벡터가 정방 MDS 행렬로 되는 것이 보증된다. 이로써, 선

형 공격에 있어서의 선형 근사에 의한 해석 곤란성을 높이는 것이 가능해져, 해석의 곤란성, 즉 키의 해석이 곤란한 안정성이 높은 암호 처리가 실현된다.

[0357] 이와 같이, 본 처리예에 의해, 차분 공격 및 선형 공격의 양쪽의 해석의 곤란성이 향상되고, 키의 해석이 곤란한 안정성이 높은 암호 처리가 실현된다. 그리고, 도 17에 나타낸 예는, 전술한 바와 같이, 먼저 설명한 처리예 a2와 처리예 b2의 조합에 의한 정방 MDS 행렬의 생성예이지만, 그 외에, 처리예 a1와 처리예 b1, 처리예 a1와 처리예 b2, 처리예 a2와 처리예 b1, 처리예 a3와 처리예 b1, 처리예 a3와 처리예 b2를 조합시켜 q개의 정방 MDS 행렬의 생성을 행하고, 단수(라운드의 수)가 2r의 페이스텔형 공통키 블록 암호 처리 구성의 각 단의 F함수부의 선형 변환부의 선형 변환 처리에 적용하는 행렬로서 홀수단째에 대하여는 상위단으로부터 차례로 L1, L2, ・・, Lq, L1, L2・・로 하여 q개의 정방 MDS 행렬을 반복 설정하고, 짝수단의 F함수에 대하여는, 하위 단의 F함수로부터 차례로, L1, L2, ・・, Lq, L1, L2・・로 하여 q개의 정방 MDS 행렬을 반복 설정함으로써도, 차분 공격 및 선형 공격의 양쪽의 해석의 곤란성이 높고, 키의 해석이 곤란한 안정성이 높은 암호 처리가 실현 가능하다.

[0358] 지금까지의 설명에서는, 알기 쉬움을 우선하여 선형 변환 행렬을 $GF(2^n)$ 상에서 정의되는 m×m의 행렬로서 mn비트로부터 mn비트로의 데이터 변환 연산으로서 왔지만, 차분 해석 및 선형 해석에 대한 마찬가지의 효과가 GF(2) 상에서 정의되는 mn×mn의 행렬을 사용한 경우라도 유효하다. 실제, 임의의 $GF(2^n)$ 상의 행렬은 같은 변환을 나타내는 GF(2) 상의 행렬에 1대1로 대응시킬 수 있다. 따라서, GF(2) 상의 행렬은 보다 일반적인 표현을 나타내고 있다고 말할 수 있다. GF(2) 상에서 는 행과 열의 개수가 mn씩이면, GF(2n)의 경우와 비교하여 n배로 된다. 그러므로 GF(2n) 상의 행렬의 1행째는, GF(2) 상의 행렬의 1로부터 n행째에 대응하고, 1열째는 1로부터 n열째에 대응하고 있다. 즉 i행째는(i－1)+1행째로부터(i-1)+n행째에 대응하고, i열째는(i－1)+1열째로부터 (i-1)+n열째에 대응하고 있다. 따라서, $GF(2^n)$ 상의 행이나 열을 인출해 오는 조작에는, GF(2) 상에서 정의되는 행렬을 사용하는 경우에는, 대응하는 n행분 또는 n열분을 인출한다는 조작을 대응시키면 된다. $GF(2^n)$ 상의 m개의 행이나 열을 인출하는 조작은, GF(2) 상에서는 n개의 행이나 열을 m회 인출하게 되어, 결과로서 mn×mn의 행렬을 얻을 수 있다. 이상의 대응부에 의해, GF(2) 상에서 정의되는 행렬로 용이하게 확장할 수 있다.

[0359] 마지막으로, 암호 처리를 실행하는 암호 처리 장치로서의 IC모듈(600)의 구성예를 도 18에 나타낸다. 전술한 처리는, 예를 들면, PC, IC 카드, 리더 라이터, 그 외에, 다양한 정보 처리 장치에 있어서 실행 가능하며, 도 18에 나타낸 IC모듈(600)은, 이들 다양한 기기로 구성할 수 있다.

[0360] 도 18에 나타낸 CPU(Central processing Unit)(601)는, 암호 처리의 개시나, 종료, 데이터의 송수신의 제어, 각 구성부 사이의 데이터 전송 제어, 그 외의 각종 프로그램을 실행하는 프로세서이다. 메모리(602)는, CPU(601)가 실행하는 프로그램, 또는 연산 파라미터로서의 고정 데이터를 저장하는 ROM(Read-Only-Memory), CPU(601)의 처리에 있어서 실행되는 프로그램, 및 프로그램 처리에 있어서 적당히 변화하는 파라미터의 저장 영역, 공작물 영역으로서 사용되는 RAM(Random Access Memory) 등으로 이루어진다. 또, 메모리(602)는 암호 처리에 필요한 키 데이터 등의 저장 영역으로서 사용 가능하다. 데이터 등의 저장 영역은, 내(耐)탬퍼 구조를 가지는 메모리로서 구성되는 것이 바람직하다.

[0361] 암호 처리부(603)는, 예를 들면, 전술한 페이스텔형 공통키 블록 암호 처리 알고리즘에 따른 암호 처리, 복호 처리 등을 실행한다. 그리고, 여기서는, 암호 처리 수단을 개별 모듈로 한 예를 나타냈으나, 이와 같은 독립된 암호 처리 모듈을 마련하지 않고, 예를 들면, 암호 처리 프로그램을 ROM에 저장하고, CPU(601)가 ROM 저장 프로그램을 판독하여 실행하도록 구성해도 된다.

[0362] 난수 발생기(604)는, 암호 처리에 필요한 키의 생성 등에 있어서 필요한 난수의 발생 처리를 실행한다.

[0363] 송수신부(605)는, 외부와의 데이터 통신을 실행하는 데이터 통신 처리부이며, 예를 들면, 리더 라이터 등, IC모듈과의 데이터 통신을 실행하고, IC모듈 내에서 생성한 암호문의 출력, 또는 외부의 리더 라이터 등의 기기로부터의 데이터 입력 등을 실행한다.

[0364] 이상, 특정한 실시예를 참조하면서, 본 발명에 대하여 상세히 설명하였다. 그러나, 본 발명의 요지를 벗어나지 않는 범위에서 당업자가 상기 실시예의 수정이나 변경을 행할 수 있는 것은 자명하다. 즉, 예시라는 형태로 본 발명을 개시한 것이며, 한정적으로 해석해서는 안된다. 본 발명의 요지를 판단하기 위해서는, 특허 청구의 범위의 란을 참작해야한다.

[0365] 그리고, 명세서 중에 있어서 설명한 일련의 처리는 하드웨어, 또는 소프트 웨어, 또는 양자의 복합 구성에 의해

실행할 수 있다. 소프트 웨어에 의한 처리를 실행하는 경우에는, 처리 시켄스를 기록한 프로그램을, 전용의 하드웨어에 내장된 컴퓨터 내의 메모리에 인스톨하여 실행시키거나, 또는 각종 처리가 실행 가능한 범용 컴퓨터에 프로그램을 인스톨하여 실행시키는 것이 가능하다.

[0366] 예를 들면, 프로그램은 기록 매체로서의 하드 디스크나 ROM(Read Only Memory)에 미리 기록하여 둘 수가 있다. 또는, 프로그램은 플렉시블 디스크, CD-ROM(Compact Disc Read Only Memory), MO(Magneto optical)디스크, DVD(Digital Versatile Disc), 자기 디스크, 반도체 메모리 등의 착탈 가능 기록 매체에, 일시 목표 또는 영속적으로 저장(기록)해 둘 수가 있다. 이와 같은 착탈 가능 기록 매체는, 이른바 패키지 소프트웨어로서 제공할 수 있다.

[0367] 그리고, 프로그램은, 전술한 바와 같은 착탈 가능 기록 매체로부터 컴퓨터에 인스톨하는 것 외에, 다운로드 사이트로부터, 컴퓨터에 무선 전송한, LAN(Local Area Network), 인터넷이라는 네트워크를 통하여, 컴퓨터에 유선으로 전송하고, 컴퓨터에서는, 그와 같이 하여 전송되어 오는 프로그램을 수신하고, 내장하는 하드 디스크 등의 기록 매체에 인스톨할 수 있다.

[0368] 그리고, 명세서에 기재된 각종의 처리는, 기재에 따라 시계열로 실행될 뿐아니라, 처리를 실행하는 장치의 처리 능력 또는 필요에 따라 병렬적으로 또는 개별적으로 실행되어도 된다. 또, 본 명세서에 있어서 시스템이란, 복수개의 장치의 논리적 집합 구성이며, 각 구성의 장치가 동일 하우징 내에 있는 것에는 한정되지 않는다.

산업상 이용 가능성

[0369] 전술한 바와 같이, 본 발명의 구성에 의하면, 비선형 변환부 및 선형 변환부를 가지는 SPN형의 F함수를, 복수의 라운드로 반복 실행하는 페이스텔형 공통키 블록 암호 처리에 있어서, 복수의 라운드 각각에 대응하는 F함수의 선형 변환 처리를, 정방 MDS 행렬을 적용한 선형 변환 처리로서 실행하는 동시에, 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서 상이한 정방 MDS 행렬: La, Lb를 적용하고, 또한 상기 정방 MDS 행렬의 역행렬: La^{-1}, Lb^{-1}을 구성하는 열 벡터로부터 임의로 선택한 m개의 열 벡터에 의해 구성하는 행렬이 선형 독립인지, 또는 정방 MDS 행렬을 구성하는 성질로 한 정방 MDS 행렬에 의한 선형 변환 처리를 실행하는 구성으로 하였으므로, 공통키 블록 암호에 있어서의 선형 공격에 대한 내성이 향상되고, 암호 키 등의 해석 곤란성이 높아지게 되어, 안정성이 높은 암호 처리가 실현된다. 따라서, 키 해석 곤란성을 높여 안전성이 요구되는 암호 처리 실행 장치에 있어서 적용가능하다.

[0370] 또한, 본 발명의 구성에 의하면, 비선형 변환부 및 선형 변환부를 가지는 SPN형의 F함수를, 복수의 라운드로 반복 실행하는 페이스텔형 공통키 블록 암호 처리에 있어서, 복수의 라운드 각각에 대응하는 F함수의 선형 변환 처리를, 정방 MDS 행렬을 적용한 선형 변환 처리로서 실행하는 동시에, 적어도 연속되는 짝수 라운드 및 연속되는 홀수 라운드의 각각에 있어서 상이한 정방 MDS 행렬을 적용하는 구성으로 하고, 이들 정방 MDS 행렬 자체가, 선형 독립성을 나타내거나, 또는 정방 MDS 행렬을 구성하는 구성으로 하였으므로, 액티브 S박스의 기여에 의한 동시 차분 캔슬의 발생하지 않는 것이 보증되어 공통키 블록 암호에 있어서의 차분 공격에 대한 강도 지표의 하나인 암호화 함수 전체에서의 액티브 S박스의 최소수를 크게 하는 것이 가능해진다. 본 구성에 의해, 선형 공격, 차분 공격의 양쪽에 대하여 내성이 향상되고, 보다 안정성이 높은 암호 처리가 실현된다. 따라서, 키 해석 곤란성을 높여 안전성이 요구되는 암호 처리 실행 장치에 있어서 적용가능하다.

도면의 간단한 설명

[0040] 도 1은 페이스텔 구조를 가지는 대표적인 공통키 블록 암호의 구성을 나타낸 도면이다.

[0041] 도 2는 라운드 함수부로서 설정되는 F함수의 구성에 대하여 설명하는 도면이다.

[0042] 도 3은 선형 변환부에 있어서, 선형 변환 처리에 적용하는 정방 행렬의 예를 나타낸 도면이다.

[0043] 도 4는 m= 8, n= 8의 128bit 블록 암호에 있어서의 3단의 동시 차분 캔슬의 상태를 설명하는 도면이다.

[0044] 도 5는 F함수의 선형 변환부에 있어서, 정방 행렬에 의한 선형 변환이 실행되어, F함수 출력 차분ΔYi를 생성하는 구체예를 설명하는 도면이다.

[0045] 도 6은 m= 8, n= 8의 128bit 블록 암호에 있어서의 5단의 동시 차분 캔슬의 상태를 설명하는 도면이다.

[0046] 도 7은 공통키 블록 암호에 있어서의 임의단의 동시 차분 캔슬의 정의를 설명하는 도면이다.

[0047] 도 8은 정방 MDS 행렬의 일례를 나타낸 도면이다.

[0048] 도 9는 본 발명에 관한 공통키 블록 암호 처리 알고리즘에 있어서의 각 라운드의 F함수의 선형 변환 행렬로서의 정방 MDS 행렬 설정예를 설명하는 도면이다.

[0049] 도 10은 본 발명에 관한 공통키 블록 암호 처리 알고리즘에 있어서의 각 라운드의 F함수의 선형 변환 행렬로서의 정방 MDS 행렬 설정 처리 시켄스를 설명하는 플로 차트이다.

[0050] 도 11은 각 라운드의 F함수로 설정하는 선형 변환 행렬인 정방 MDS 행렬의 생성 방법으로서 차분 공격에 대한 내성 향상을 실현하는 정방 MDS 행렬 생성 처리예 a1를 설명하는 플로 차트이다.

[0051] 도 12는 각 라운드의 F함수로 설정하는 선형 변환 행렬인 정방 MDS 행렬의 생성 방법으로서 차분 공격에 대한 내성 향상을 실현하는 정방 MDS 행렬 생성 처리예 a2를 설명하는 플로 차트이다.

[0052] 도 13은 각 라운드의 F함수로 설정하는 선형 변환 행렬인 정방 MDS 행렬의 생성 방법으로서 차분 공격에 대한 내성 향상을 실현하는 정방 MDS 행렬 생성 처리예 a3를 설명하는 플로 차트이다.

[0053] 도 14는 각 라운드의 F함수로 설정하는 선형 변환 행렬인 정방 MDS 행렬의 생성 처리예 a3의 구체적 방법을 설명하는 도면이다.

[0054] 도 15는 각 라운드의 F함수로 설정하는 선형 변환 행렬인 정방 MDS 행렬의 생성 방법으로서 선형 공격에 대한 내성 향상을 실현하는 정방 MDS 행렬 생성 처리예 b1를 설명하는 플로 차트이다.

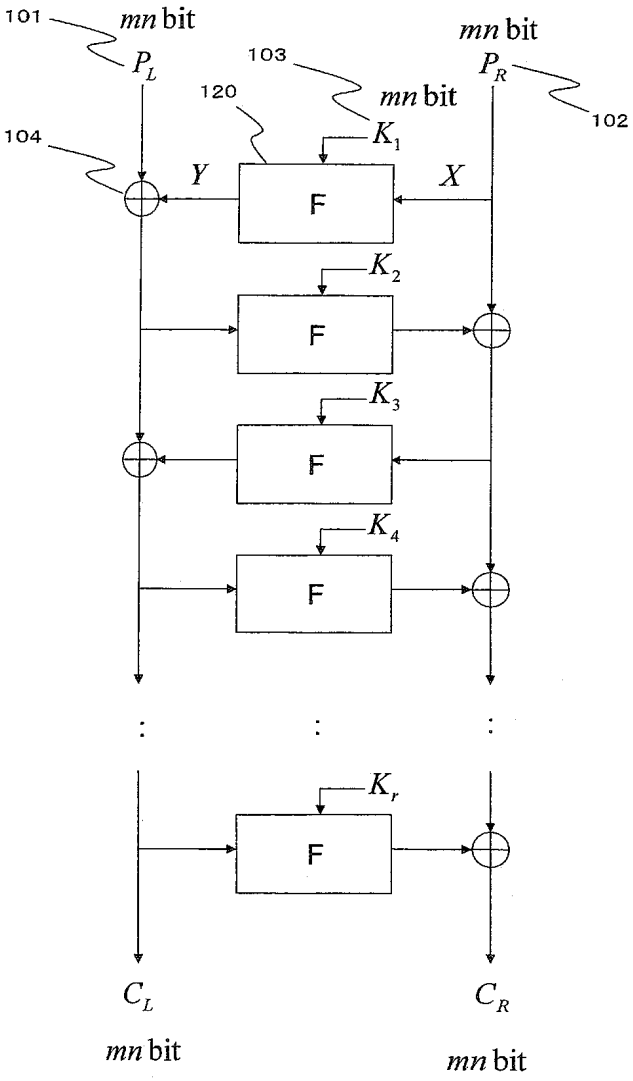
[0055] 도 16은 각 라운드의 F함수로 설정하는 선형 변환 행렬인 정방 MDS 행렬의 생성 방법으로서 선형 공격에 대한 내성 향상을 실현하는 정방 MDS 행렬 생성 처리예 b2를 설명하는 플로 차트이다.

[0056] 도 17은 각 라운드의 F함수로 설정하는 선형 변환 행렬인 정방 MDS 행렬의 생성 방법으로서 차분 공격 및 선형 공격에 대한 내성 향상을 실현하는 정방 MDS 행렬 생성 처리예를 설명하는 플로 차트이다.

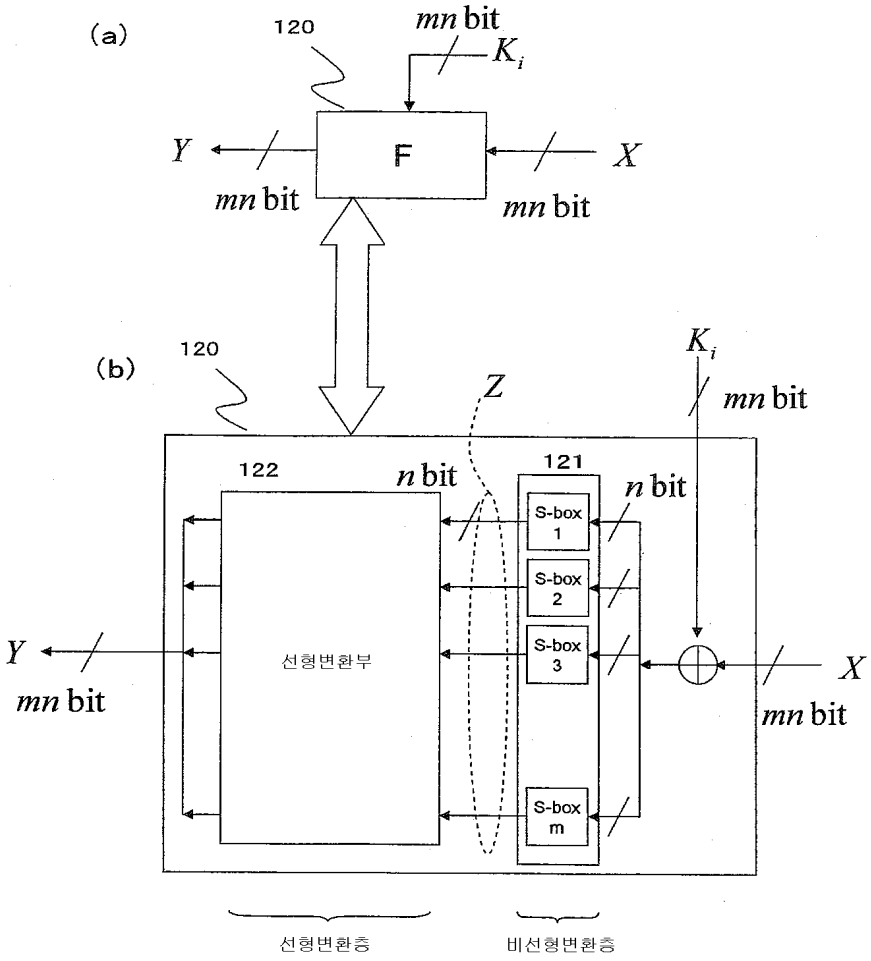
[0057] 도 18은 본 발명에 관한 암호 처리를 실행하는 암호 처리 장치로서의 IC모듈의 구성예를 나타낸 도면이다.

도면

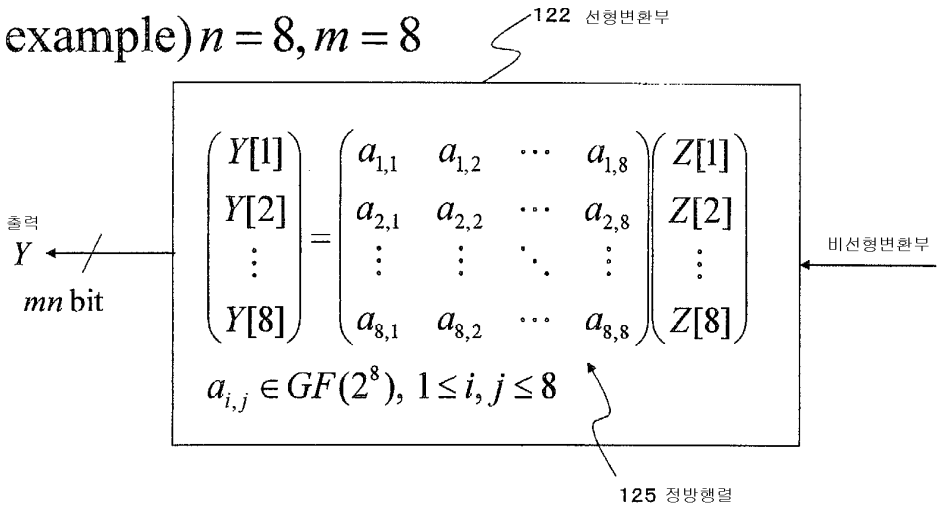
도면1


101
mn bit
P_L
103
mn bit
120
K_1
104
Y
F
X
mn bit
P_R
102
K_2
F
K_3
F
K_4
F
:
:
:
K_r
F
C_L
mn bit
C_R
mn bit

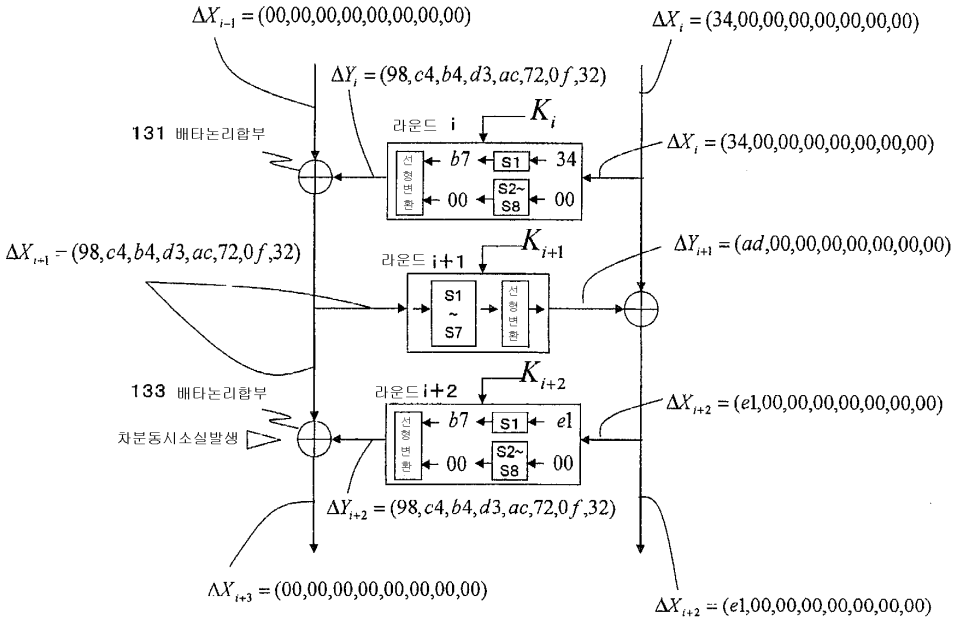
도면2


(a)
120
mn bit
K_i
Y
F
X
mn bit
mn bit
(b)
120
K_i
Z
mn bit
122
n bit
121
n bit
S-box 1
S-box 2
S-box 3
S-box m
Y
선형변환부
X
mn bit
mn bit
선형변환층
비선형변환층

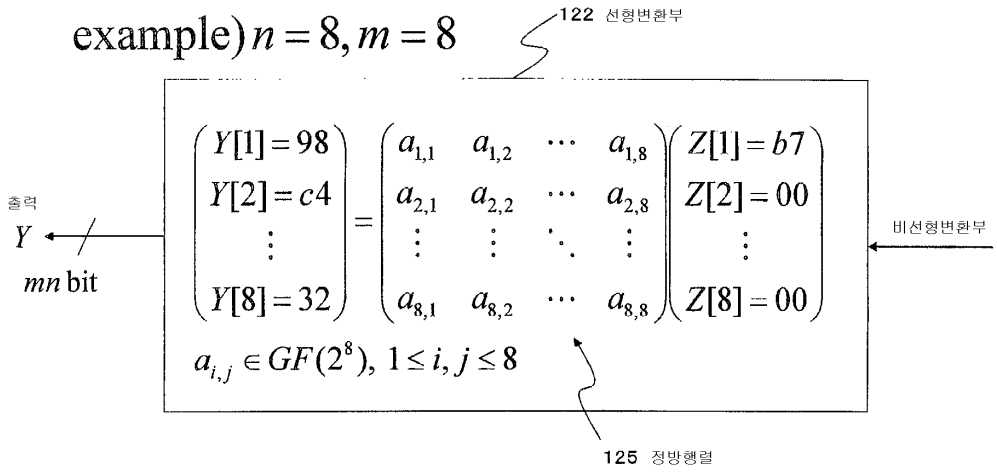
도면3


example) $n=8, m=8$
122 선형변환부
$\begin{pmatrix} Y[1] \\ Y[2] \\ \vdots \\ Y[8] \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,8} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,8} \\ \vdots & \vdots & \ddots & \vdots \\ a_{8,1} & a_{8,2} & \cdots & a_{8,8} \end{pmatrix} \begin{pmatrix} Z[1] \\ Z[2] \\ \vdots \\ Z[8] \end{pmatrix}$
$a_{i,j} \in GF(2^8),\ 1 \le i, j \le 8$
출력
Y
mn bit
비선형변환부
125 정방행렬

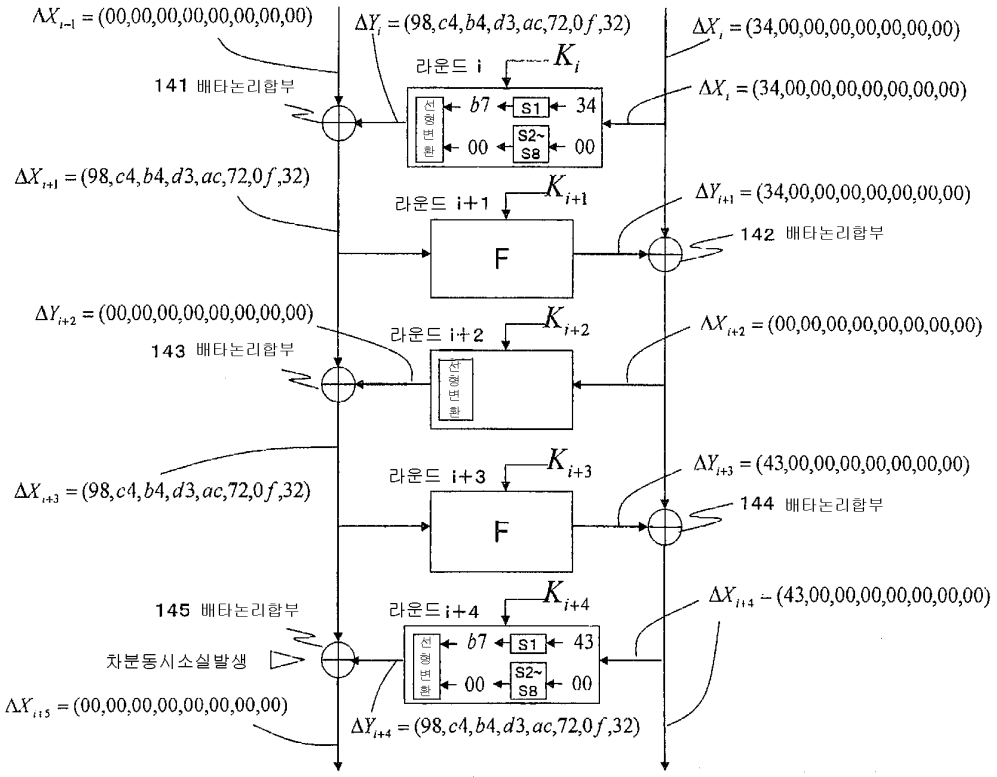
도면4


$\Delta X_{i-1} = (00,00,00,00,00,00,00,00)$
$\Delta Y_i = (98,c4,b4,d3,ac,72,0f,32)$
$\Delta X_i = (34,00,00,00,00,00,00,00)$
131 배타논리합부
라운드 i
K_i
선형변환
b7 S1 34
00 S2~S8 00
$\Delta X_i = (34,00,00,00,00,00,00,00)$
$\Delta X_{i+1} = (98,c4,b4,d3,ac,72,0f,32)$
라운드 i+1
K_{i+1}
S1 ~ S7
선형변환
$\Delta Y_{i+1} = (ad,00,00,00,00,00,00,00)$
133 배타논리합부
차분동시소실발생
라운드 i+2
K_{i+2}
선형변환
b7 S1 e1
00 S2~S8 00
$\Delta X_{i+2} = (e1,00,00,00,00,00,00,00)$
$\Delta Y_{i+2} = (98,c4,b4,d3,ac,72,0f,32)$
$\Delta X_{i+3} = (00,00,00,00,00,00,00,00)$
$\Delta X_{i+2} = (e1,00,00,00,00,00,00,00)$

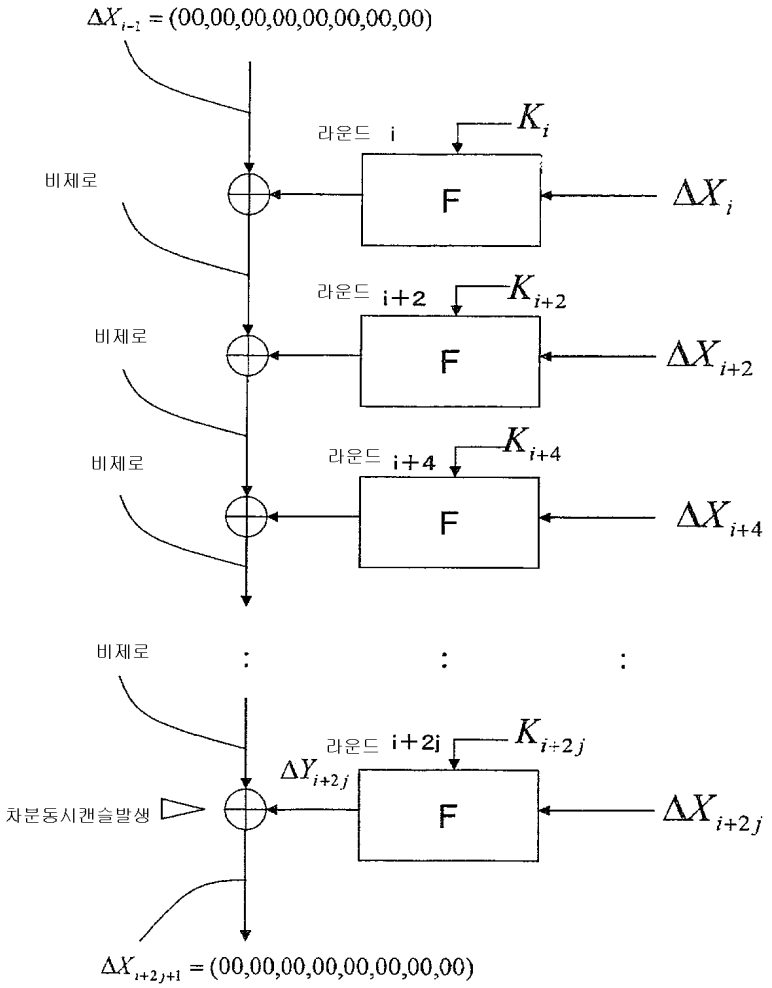
도면5


example) $n=8, m=8$
122 선형변환부
$\begin{pmatrix} Y[1]=98 \\ Y[2]=c4 \\ \vdots \\ Y[8]=32 \end{pmatrix} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,8} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,8} \\ \vdots & \vdots & \ddots & \vdots \\ a_{8,1} & a_{8,2} & \cdots & a_{8,8} \end{pmatrix} \begin{pmatrix} Z[1]=b7 \\ Z[2]=00 \\ \vdots \\ Z[8]=00 \end{pmatrix}$
$a_{i,j} \in GF(2^8),\ 1 \le i, j \le 8$
125 정방행렬
출력
Y
mn bit
비선형변환부

도면6


$\Delta X_{i-1} = (00,00,00,00,00,00,00,00)$
$\Delta Y_i = (98,c4,b4,d3,ac,72,0f,32)$
$\Delta X_i = (34,00,00,00,00,00,00,00)$
141 배타논리합부
라운드 i
K_i
선형변환
b7 S1 34
S2~S8 00 00
$\Delta X_{i+1} = (98,c4,b4,d3,ac,72,0f,32)$
$\Delta Y_{i+1} = (34,00,00,00,00,00,00,00)$
라운드 i+1
K_{i+1}
F
142 배타논리합부
$\Delta Y_{i+2} = (00,00,00,00,00,00,00,00)$
$\Delta X_{i+2} = (00,00,00,00,00,00,00,00)$
143 배타논리합부
라운드 i+2
K_{i+2}
$\Delta X_{i+3} = (98,c4,b4,d3,ac,72,0f,32)$
$\Delta Y_{i+3} = (43,00,00,00,00,00,00,00)$
라운드 i+3
K_{i+3}
F
144 배타논리합부
145 배타논리합부
라운드 i+4
K_{i+4}
$\Delta X_{i+4} = (43,00,00,00,00,00,00,00)$
차분동시소실발생
b7 S1 43
S2~S8 00 00
$\Delta X_{i+5} = (00,00,00,00,00,00,00,00)$
$\Delta Y_{i+4} = (98,c4,b4,d3,ac,72,0f,32)$

도면7

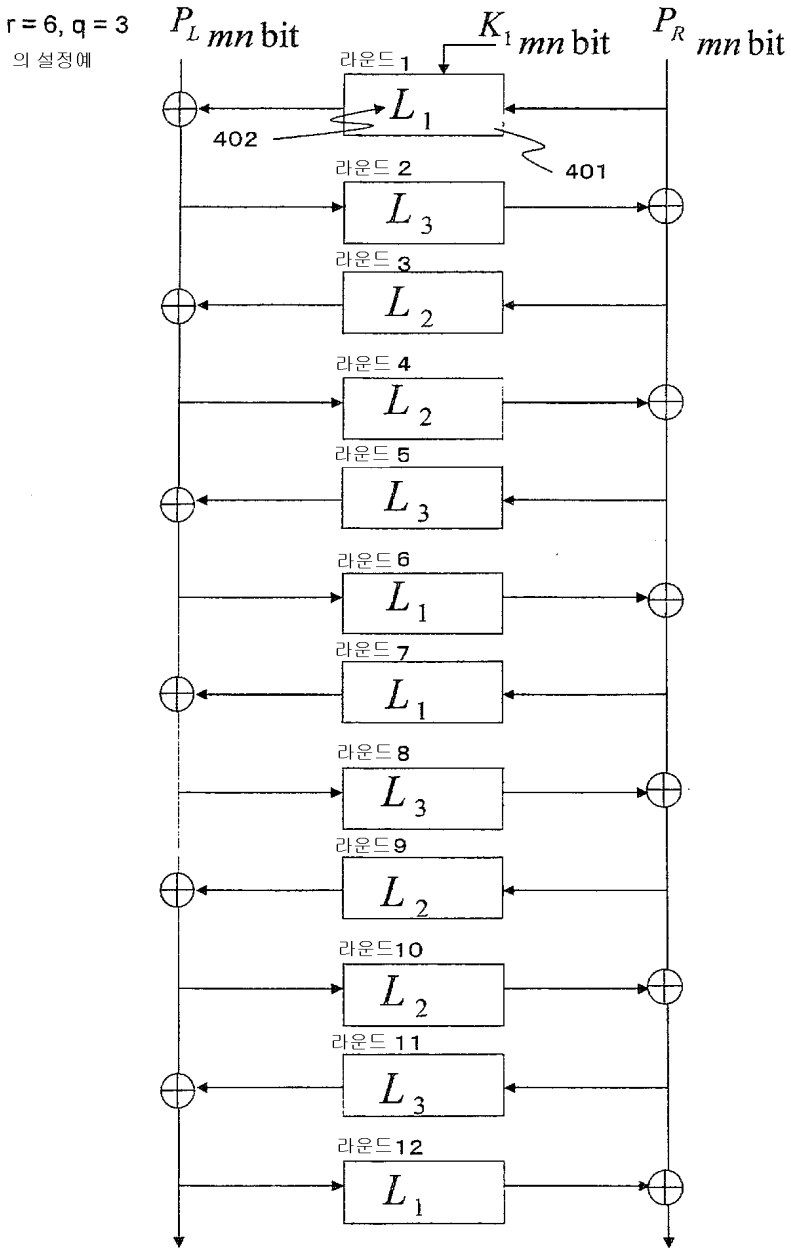

$\Delta X_{i-1} = (00,00,00,00,00,00,00,00)$
라운드 i
K_i
F
ΔX_i
비제로
라운드 i+2
K_{i+2}
F
ΔX_{i+2}
비제로
라운드 i+4
K_{i+4}
F
ΔX_{i+4}
비제로
:
:
:
라운드 i+2j
K_{i+2j}
ΔY_{i+2j}
F
ΔX_{i+2j}
차분동시캔슬발생
$\Delta X_{i+2j+1} = (00,00,00,00,00,00,00,00)$

도면8

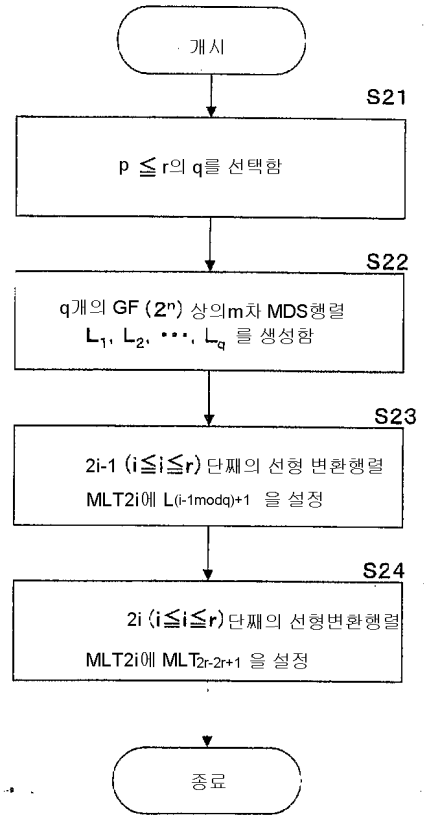
example) $n = 8, m = 8$

$$\begin{pmatrix} 9d & b4 & d3 & 5d & 84 & ae & ec & b9 \\ 29 & 34 & 39 & 60 & 5c & 81 & 25 & 13 \\ 67 & 6a & d2 & e3 & 4b & db & 9d & 4 \\ 8e & d7 & e6 & 1b & 8b & 9e & 3a & 91 \\ d9 & e5 & 4d & dd & c6 & 5 & f0 & ad \\ 2a & f7 & 67 & 72 & b1 & 7 & f2 & 27 \\ 42 & e6 & a0 & 4 & f1 & 4 & 7d & 8c \\ 55 & 63 & fa & 51 & c & d9 & 28 & d6 \end{pmatrix}$$

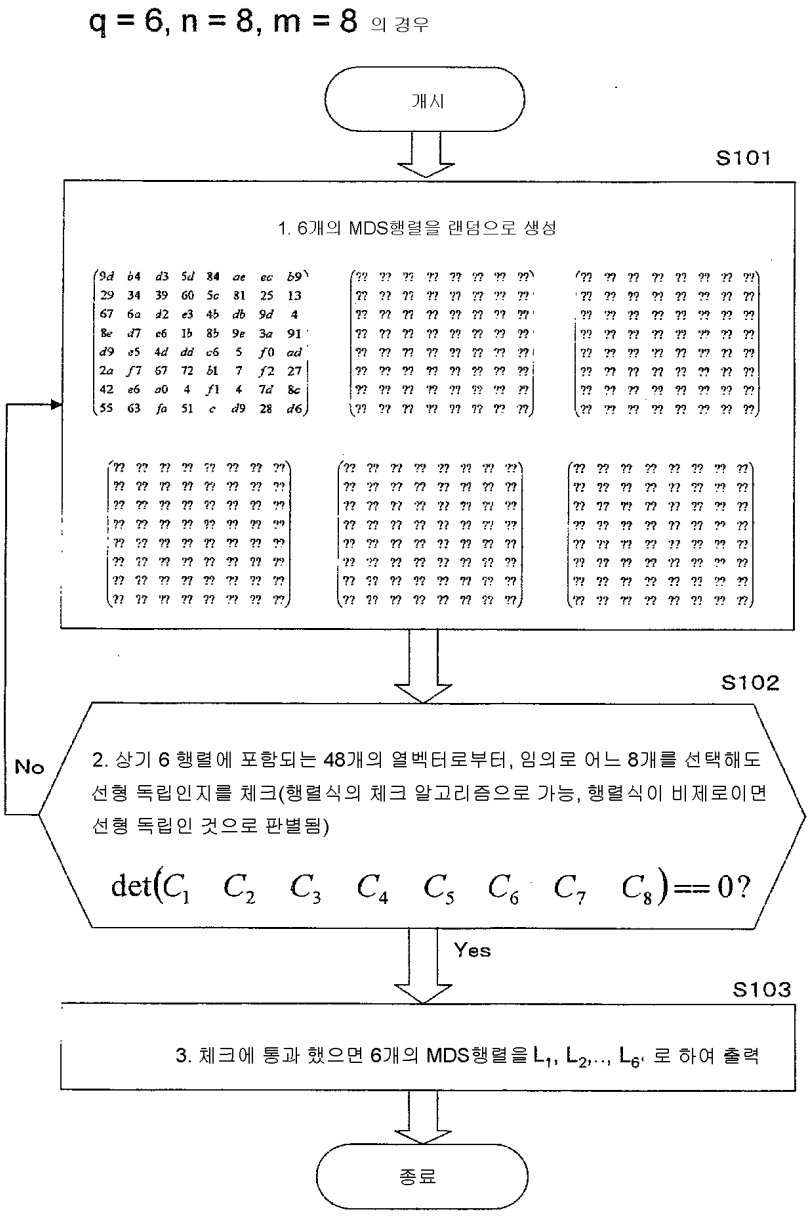
도면9


r = 6, q = 3
의 설정예
P_L mn bit
K_1 mn bit
P_R mn bit
라운드 1
L_1
402
401
라운드 2
L_3
라운드 3
L_2
라운드 4
L_2
라운드 5
L_3
라운드 6
L_1
라운드 7
L_1
라운드 8
L_3
라운드 9
L_2
라운드10
L_2
라운드 11
L_3
라운드12
L_1

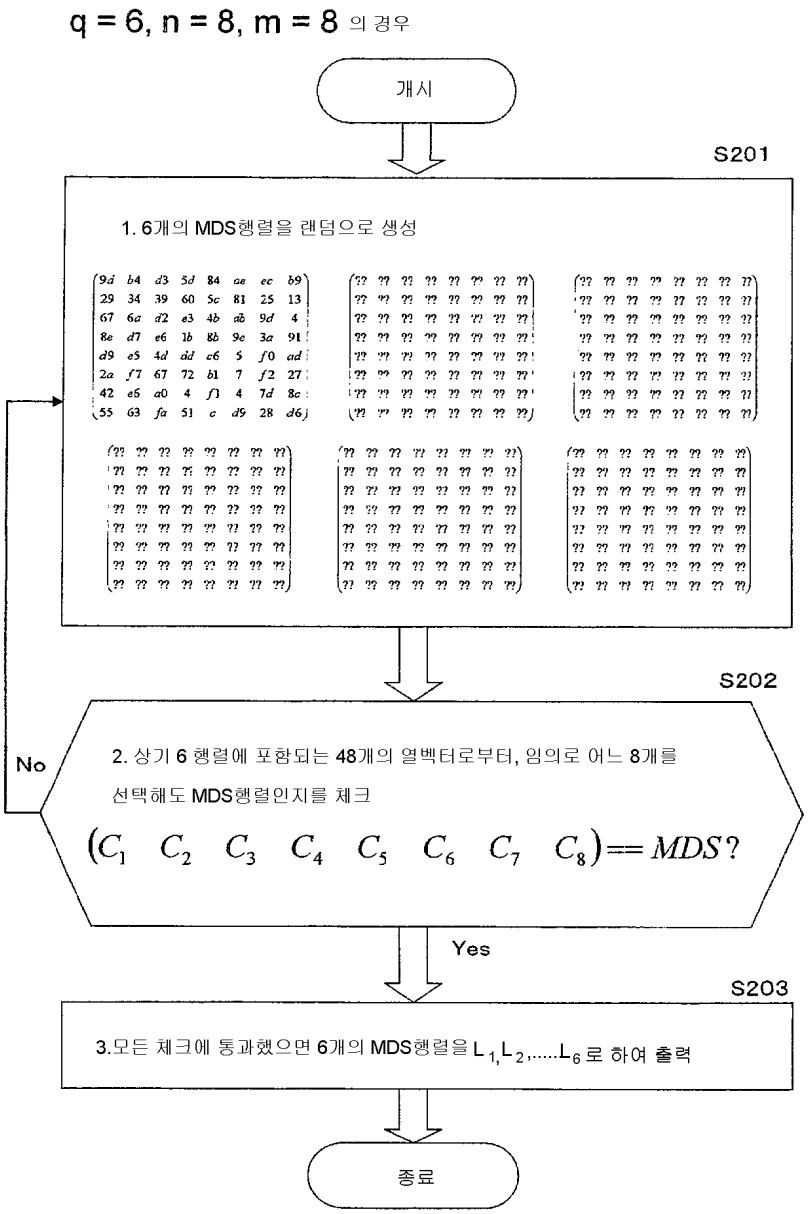
도면10


개시
S21
p ≦ r의 q를 선택함
S22
q개의 GF (2^n) 상의 m차 MDS행렬
L1, L2, ···, Lq 를 생성함
S23
2i-1 (i≦i≦r) 단째의 선형 변환행렬
MLT2i에 L(i-1modq)+1 을 설정
S24
2i (i≦i≦r) 단째의 선형변환행렬
MLT2i에 MLT2r-2r+1 을 설정
종료

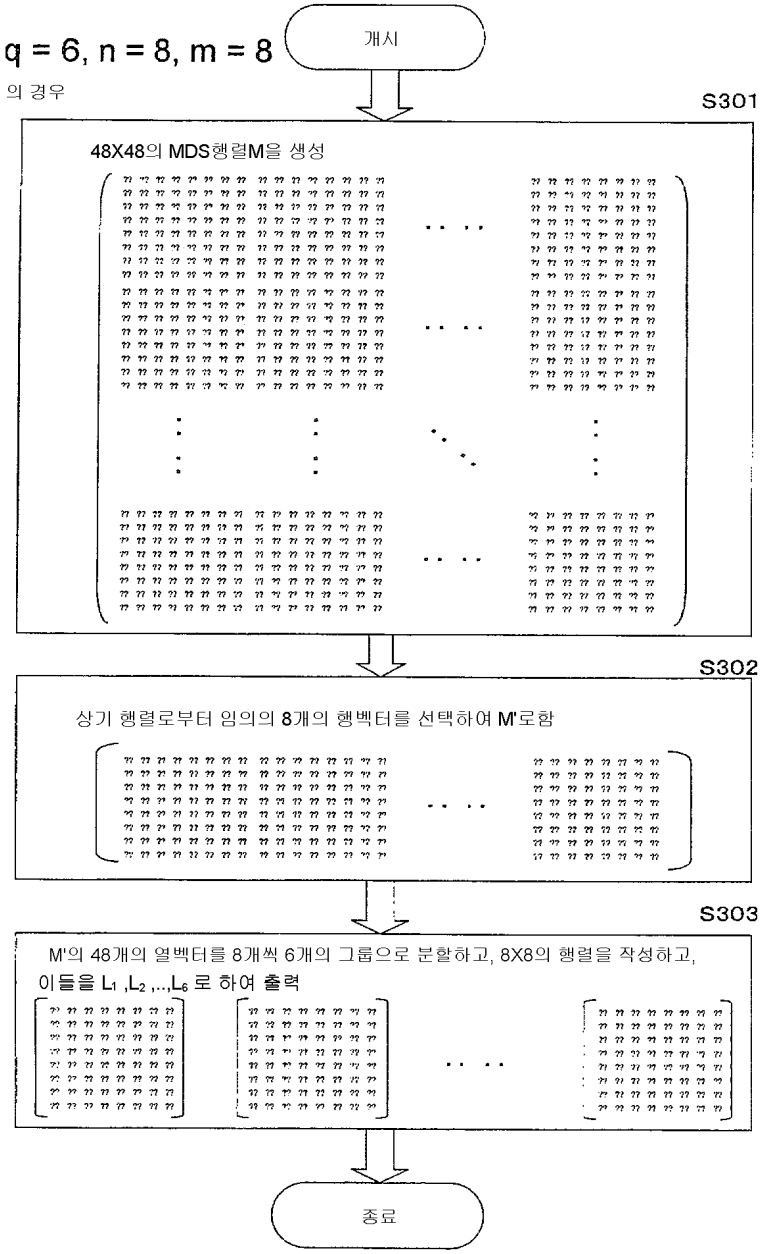
도면11


q = 6, n = 8, m = 8 의 경우
개시
S101
1. 6개의 MDS행렬을 랜덤으로 생성
S102
No
2. 상기 6 행렬에 포함되는 48개의 열벡터로부터, 임의로 어느 8개를 선택해도 선형 독립인지를 체크(행렬식의 체크 알고리즘으로 가능, 행렬식이 비제로이면 선형 독립인 것으로 판별됨)
$\det(C_1 \quad C_2 \quad C_3 \quad C_4 \quad C_5 \quad C_6 \quad C_7 \quad C_8) == 0?$
Yes
S103
3. 체크에 통과 했으면 6개의 MDS행렬을 $L_1, L_2, .., L_6$ 로 하여 출력
종료

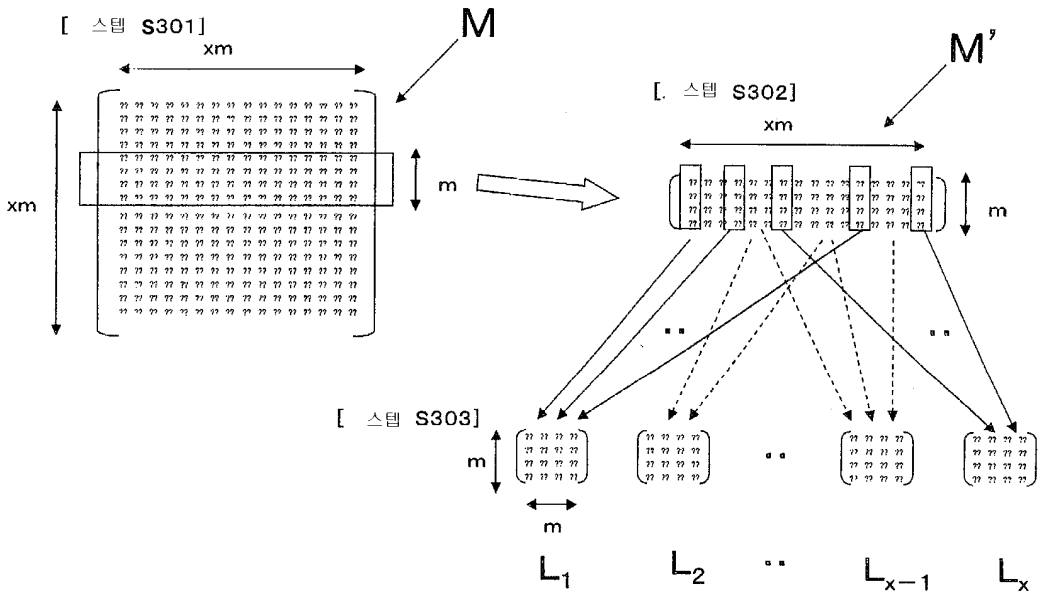
도면12


q = 6, n = 8, m = 8 의 경우
개시
S201
1. 6개의 MDS행렬을 랜덤으로 생성
S202
No
2. 상기 6 행렬에 포함되는 48개의 열벡터로부터, 임의로 어느 8개를 선택해도 MDS행렬인지를 체크
$(C_1 \quad C_2 \quad C_3 \quad C_4 \quad C_5 \quad C_6 \quad C_7 \quad C_8) == MDS?$
Yes
S203
3.모든 체크에 통과했으면 6개의 MDS행렬을 $L_1, L_2, \ldots L_6$로 하여 출력
종료

도면13

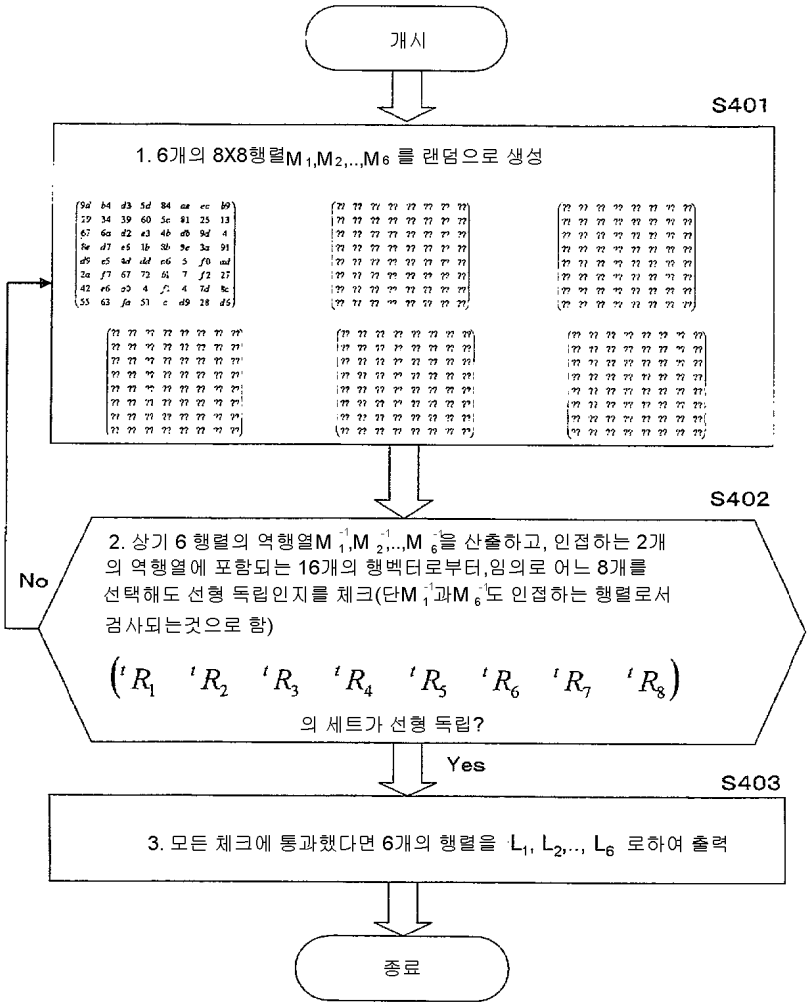

q = 6, n = 8, m = 8
의 경우
개시
S301
48X48의 MDS행렬M을 생성
S302
상기 행렬로부터 임의의 8개의 행벡터를 선택하여 M'로함
S303
M'의 48개의 열벡터를 8개씩 6개의 그룹으로 분할하고, 8X8의 행렬을 작성하고,
이들을 L1 ,L2 ,..,L6 로 하여 출력
종료

도면14



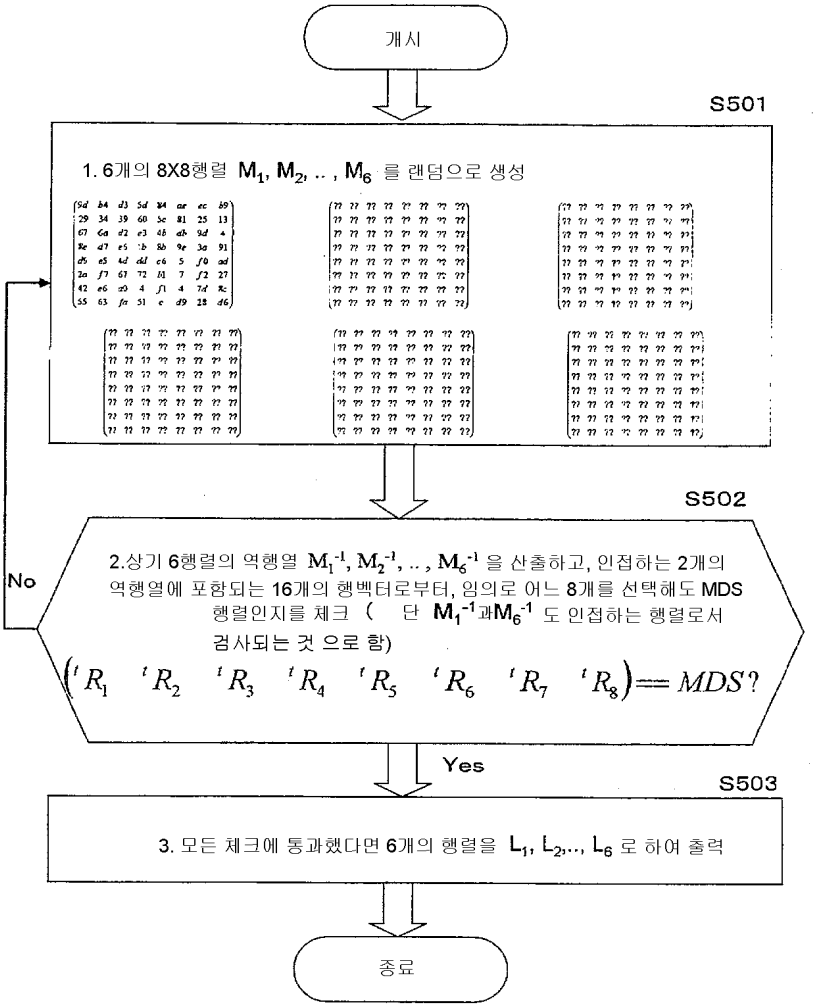
도면15

q = 6, n = 8, m = 8 의 경우

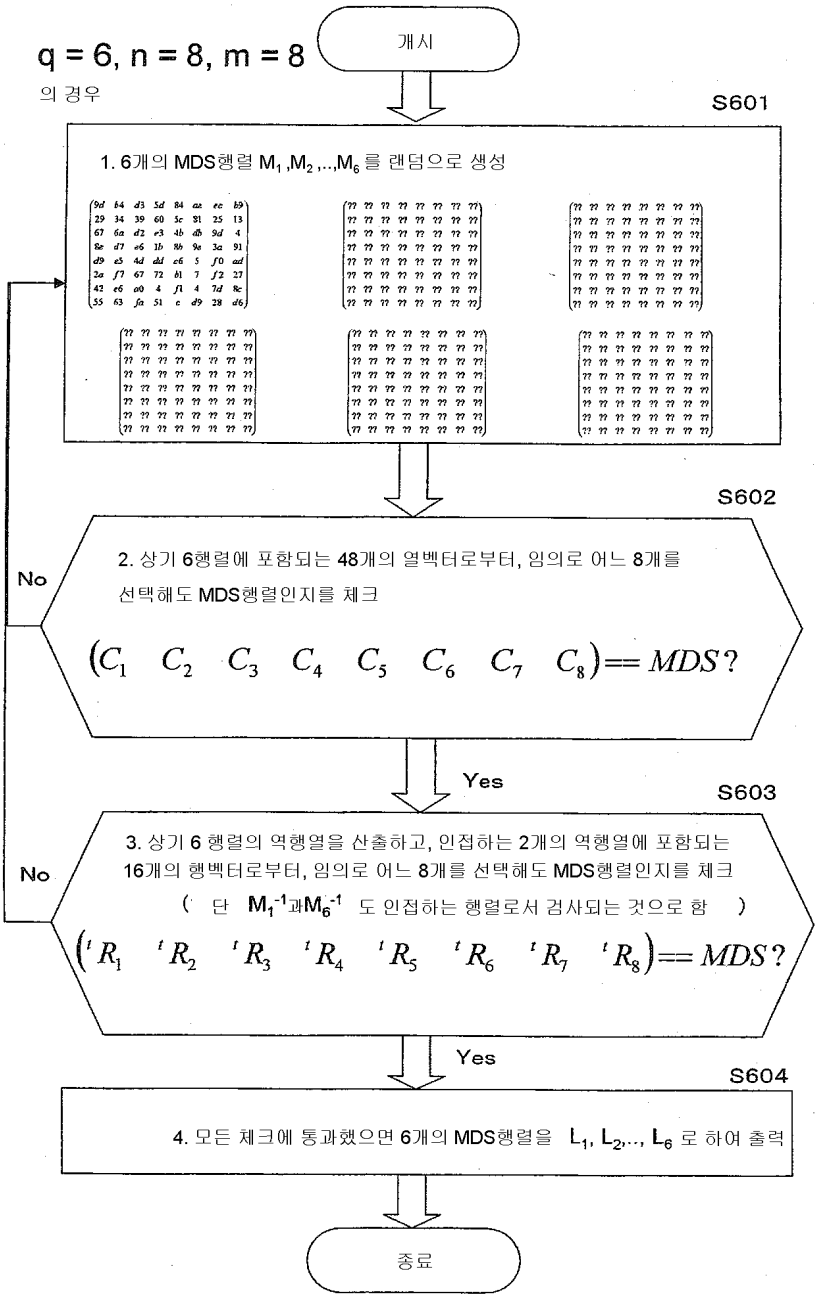


도면16

$q = 6, n = 8, m = 8$ 의 경우



도면17


q = 6, n = 8, m = 8
의 경우
개시
S601
1. 6개의 MDS행렬 $M_1, M_2, .., M_6$ 를 랜덤으로 생성
S602
No
2. 상기 6행렬에 포함되는 48개의 열벡터로부터, 임의로 어느 8개를 선택해도 MDS행렬인지를 체크
$(C_1 \quad C_2 \quad C_3 \quad C_4 \quad C_5 \quad C_6 \quad C_7 \quad C_8) == MDS?$
Yes
S603
No
3. 상기 6 행렬의 역행열을 산출하고, 인접하는 2개의 역행열에 포함되는 16개의 행벡터로부터, 임의로 어느 8개를 선택해도 MDS행렬인지를 체크
(단 M_1^{-1}과M_6^{-1} 도 인접하는 행렬로서 검사되는 것으로 함)
$({}^tR_1 \quad {}^tR_2 \quad {}^tR_3 \quad {}^tR_4 \quad {}^tR_5 \quad {}^tR_6 \quad {}^tR_7 \quad {}^tR_8) == MDS?$
Yes
S604
4. 모든 체크에 통과했으면 6개의 MDS행렬을 $L_1, L_2, .., L_6$ 로 하여 출력
종료

도면18