(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。Int. Cl. [7]
G09C 1/00

(11) 공개번호 특2002-0006475
(43) 공개일자 2002년01월19일

(21) 출원번호 10-2001-0041607
(22) 출원일자 2001년07월11일

(30) 우선권주장 JP-P-2000-0021 2000년07월12일 일본(JP)
1686

(71) 출원인 가부시끼가이샤 도시바
니시무로 타이죠
일본국 도꾜도 미나또꾸 시바우라 1쪼메 1방 1고

(72) 발명자 무라타니히로후미
일본국가나가와현가와사키시사이와이구고무가이도시바정1번지가부시끼가이샤도시바연구개발센터내
모토야마마사히코
일본국가나가와현가와사키시사이와이구고무가이도시바정1번지가부시끼가이샤도시바연구개발센터내
오쿠마겐지
일본국가나가와현가와사키시사이와이구고무가이도시바정1번지가부시끼가이샤도시바연구개발센터내
사노후미히코
일본국도쿄도후추시도시바정1번지가부시끼가이샤도시바후추공장내
가와무라신이치
일본국가나가와현가와사키시사이와이구고무가이도시바정1번지가부시끼가이샤도시바연구개발센터내

(74) 대리인 김윤배
이범일

심사청구 : 있음

(54) 암호화장치, 복호장치 및 확대키 생성장치, 확대키생성방법 및 기록매체

요약

암호화시와 복호(復號)시에서 확대키의 사용순서가 역으로 되는 공통키 블록 암호방식의 암호화장치의 확대키 생성부에 있어서, 초단으로부터의 단수와 최종단으로부터의 단수가 같은 2개의 라운드(round)함수 f_1과 f_{n+1} 을 서로 역함수로 되도록 설정한다. 이에 따라, 암호화시의 확대키 생성과 복호시의 확대키 생성이 기본적으로 동일하게 되므로, 암호화시에도 복호시에도 공통키를 입력으로 하여 직접 또한 사용순서로 확대키를 축차 생성해 가는 것이 가능하게 되어, 확대키 생성을 위한 지연시간의 발생을 회피하면서 쉴 사이 없는 키 생성을 가능하게 한다.

대표도

도 1

명세서

도면의 간단한 설명

도 1은 본 발명의 한 실시형태에 따른 암호화장치의 구성례를 나타낸 도면,

도 2는 동 실시형태에 따른 복호장치의 구성례를 나타낸 도면,

도 3은 라운드 트립 구성의 라운드함수의 계열의 구성에 대해 설명하기 위한 도면,

도 4는 루프 구성의 라운드함수의 계열의 구성에 대해 설명하기 위한 도면,

도 5는 라운드 트립 구성과 루프 구성의 복합형의 라운드함수의 계열의 구성에 대해 설명하기 위한 도면,

도 6은 라운드 트립 구성과 루프 구성의 복합형의 라운드함수의 계열의 구성에 대해 설명하기 위한 도면,

도 7은 라운드 트립 구성과 루프 구성의 복합형의 라운드함수의 계열의 구성에 대해 설명하기 위한 도면,

도 8은 라운드 트립 구성과 루프 구성의 복합형의 라운드함수의 계열의 구성에 대해 설명하기 위한 도면,

도 9는 라운드 트립 구성과 루프 구성의 복합형의 라운드함수의 계열의 구성에 대해 설명하기 위한 도면,

도 10은 동 실시형태에 따른 암호화장치의 다른 구성례를 나타낸 도면,

도 11은 동 실시형태에 따른 복호장치의 다른 구성례를 나타낸 도면,

도 12는 도 10, 도 11의 확대키와 교반(攪拌)처리의 접속관계예를 나타낸 도면,

도 13은 도 10, 도 11의 확대키와 교반처리의 접속관계예를 나타낸 도면,

도 14는 도 10, 도 11의 확대키와 교반처리의 접속관계예를 나타낸 도면,

도 15는 도 10, 도 11의 확대키와 교반처리의 접속관계예를 나타낸 도면,

도 16은 동 실시형태에 따른 암호화장치의 더욱 다른 구성례를 나타낸 도면,

도 17은 동 실시형태에 따른 복호장치의 더욱 다른 구성례를 나타낸 도면,

도 18은 동 실시형태에 따른 암호화장치의 더욱 다른 구성례를 나타낸 도면,

도 19는 동 실시형태에 따른 확대키 생성부의 구성례를 나타낸 도면,

도 20은 동 실시형태에 따른 라운드 처리부의 구성례를 나타낸 도면,

도 21은 동 실시형태에 따른 라운드 처리부의 다른 구성례를 나타낸 도면,

도 22는 도 21의 라운드 처리부의 비선형 함수 유닛(unit: 단위)의 구성례를 나타낸 도면,

도 23은 도 21의 라운드 처리부의 역함수를 갖는 라운드 처리부의 구성례를 나타낸 도면,

도 24는 동 실시형태에 따른 암호화장치의 더욱 다른 구성례를 나타낸 도면,

도 25는 도 24의 제1유닛 DU의 구성례를 나타낸 도면,

도 26은 도 24의 제2유닛 DD의 구성례를 나타낸 도면,

도 27은 도 24의 제3유닛 DD(woMDSH)의 구성례를 나타낸 도면,

도 28은 도 24의 확대키 생성부의 구성례를 나타낸 도면,

도 29는 도 24의 확대키 생성부의 다른 구성례를 나타낸 도면,

도 30은 도 28과 도 29의 비선형 함수 유닛을 설명하기 위한 도면,

도 31은 도 28과 도 29의 배타적 논리합에 의한 유닛을 설명하기 위한 도면,

도 32는 도 28과 도 29의 배타적 논리합에 의한 유닛을 설명하기 위한 도면,

도 33은 도 28과 도 29의 갈루아(Galois)체 상의 승산에 의한 유닛을 설명하기 위한 도면,

도 34a, 도 34b는 도 28과 도 29의 갈루아체 상의 승산에 의한 유닛을 설명하기 위한 도면,

도 35는 갈루아체 상의 승산을 실현하는 결선패턴도,

도 36은 동 실시형태에 따른 암호화장치의 확대키 생성부의 더욱 다른 구성례를 나타낸 도면,

도 37은 도 36의 비선형 함수 유닛 F의 구성례를 나타낸 도면,

도 38a, 도 38b는 도 36의 배타적 논리합에 의한 유닛을 설명하기 위한 도면,

도 39a, 도 39b는 패딩단의 구성을 나타낸 도면,

도 40은 더미단의 구성을 나타낸 도면,

도 41은 동 실시형태에 따른 암호화장치의 확대키 생성부의 더욱 다른 구성례를 나타낸 도면,

도 42는 도 41의 비선형 함수 유닛 F의 구성례를 나타낸 도면,

도 43a, 도 43b는 도 41의 배타적 논리합에 의한 유닛을 설명하기 위한 도면,

도 44는 더미단의 구성을 나타낸 도면,

도 45는 동 실시형태의 암호방식을 이용한 시스템의 일례를 나타낸 도면,

도 46은 동 실시형태의 암호방식을 이용한 시스템의 다른 예를 나타낸 도면,

도 47은 동 실시형태의 암호방식을 이용한 시스템의 더욱 다른 예를 나타낸 도면,

도 48은 종래의 확대키 생성장치에 대해 설명하기 위한 도면,

도 49는 종래의 확대키 생성장치에 대해 설명하기 위한 도면이다.

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은, 암호화(encryption)시와 복호(decryption)시에서 복수의 확대키를 역의 순번으로 이용하는 암호화장치, 복호장치 및 확대키 생성장치(expanded key generating device), 확대키 생성방법 및 기록매체에 관한 것이다.

전자화된 정보, 특히 저작권에 관계된 정보나 기밀정보나 프라이버시에 관계된 정보 등의 보안제어(security control)를 위해, 암호기술의 중요성이 대단히 높아지고 있다. 실제로 암호기술은 각종 분야에 있어서 여러 가지 형태로 이용되고 있다.

암호방식에는 여러 가지의 것이 있지만, 그 중의 하나로 공통키 암호방식이 있다. 공통키 암호방식은, 암호화시에 이용된 키와 동일한 키(공통키의 경우도 있고, 비밀키의 경우도 있음)를 이용하여 복호(復號)가 행하여지는 방식이다.

공통키 암호방식에도 여러 가지의 것이 있지만, 그 중의 하나로 확대키를 이용하는 방식이 있다. 이 방식에서는, 공통키를 기초로, 그것이 갖는 비트수보다 많은 총비트수의 복수의 확대키를 생성한다.

확대키의 생성방식의 하나로, 공통키에 대해 라운드함수[단(段: stage)함수]를 작용시키고, 그 출력값을 기초로 확대키를 생성함과 더불어, 더욱이 그 출력값에 라운드함수를 작용시키며, 그 출력값을 기초로 다음의 확대키를 생성함과 더불어, 더욱이 그 출력값에 라운드함수를 작용시키는 등과 같이, 잇달아 라운드함수를 작용시켜 확대키를 축차적으로 생성해 가는 것이 있다. 이러한 방식을 여기서는 라운드방식이라 부르기로 한다.

이러한 확대키 생성방식을 취하는 공통키 암호방식으로서는, 예컨대 공통키 블록 암호방식이 있다. 공통키 블록 암호방식은, 데이터 교반부(攪拌部)에 대해서도 처리단위로 되는 소정 비트길이의 블록 데이터에, 라운드함수를 잇달아 작용시켜 암호화 또는 복호를 행하는 구조를 갖는 것으로, 그 대표적인 기본구조로 SPN형과 페이스텔(Feistel)형 등이 있다.

그런데, 확대키의 생성에 라운드방식을 취하는 경우에는, 예컨대 블록암호와 같이 암호화시에 이용된 순번과는 역의 순번으로 확대키를 이용하는 것이 요구된다.

이하, 이러한 방식의 문제점에 대해 설명한다.

도 48에 종래의 암호화장치의 확대키 생성부의 구성례를 나타낸다. 생성부는 직렬로 접속된 라운드함수 처리부(1001_1 1001_n)와, 라운드함수 처리부(1001_1 1001_n)의 출력에 각각 접속된 확대키 변환부(1005_1 1005_n)를 구비한다.

데이터 교반부에서는 암호화처리에 확대키 (1)을 필요로 한다. 그래서, 공통키에 라운드함수 (1)을 작용시켜 그 출력값을 구하고, 이것에 확대키 변환 (1)을 작용시켜 확대키 (1)을 얻는다. 데이터 교반부는 이 확대키 (1)을 이용하여 암호화처리를 행한다.

데이터 교반부에서는 암호화처리에 확대키 (2)도 필요로 한다. 그래서, 라운드함수 (1)의 출력값에 라운드함수 (2)를 작용시켜 그 출력값을 구하고, 이것에 확대키 변환 (2)를 작용시켜 확대키 (2)를 얻는다. 데이터 교반부는 이 확대키 (2)를 이용하여 암호화처리를 행한다.

이후, 마찬가지로 하여 확대키 생성부에 의한 확대키의 생성과, 데이터 교반부에 의한 암호화처리가 행하여진다.

여기서, 복호시의 처리를 생각해 보자.

복호시에는, 암호화시와는 역의 순번, 즉 확대키 (n)→확대키 (1)의 순번으로, 확대키를 이용할 필요가 있다. 그런데, 도 48과 마찬가지의 구성의 확대키 생성부를 갖춘 종래의 복호장치에서는, 확대키는 확대키 (1)→확대키 (n)의 순번으로 생성되기 때문에, 예컨대 데이터 교반부의 처리에 앞서 모든 확대키를 생성하여 메모리에 기억해 둘 필요가 있었다.

그렇지만, 예컨대 IC카드와 같이 빈약한 하드웨어 환경밖에 없는 장치에서는, 복호에 필요한 확대키를 모두 격납할 만큼의 기억영역의 여유가 없다고 하는 문제점이 있다.

이 문제를 회피하기 위해, 도 49에 예시한 바와 같은 구성의 복호장치의 확대키 생성부를 생각할 수 있다. 생성부는 직렬로 접속된 라운드함수 처리부(1001_1 1001_n)와, 최종단의 라운드함수 처리부(1001_n)의 출력에 직렬로 더 접속되어 라운드함수 처리부(1001_n 1001_1)의 라운드함수 처리의 역의 처리를 행하는 라운드함수 처리부(1021_n 1021_2) 및, 라운드함수 처리부(1001_n, 1021_n 1021_2)의 출력에 각각 접속된 확대키 변환부(1005_n 1005_1)를 구비한다.

이 구성에서는, 일단 암호화시와 동일한 확대키 생성처리를 행하여 최종단의 라운드함수를 작용시켜 얻어지는 출력값(Rn)을 구한다. 그 후, 새삼 그 출력값(Rn)에 암호화시와는 역의 라운드방향으로 각 라운드함수의 역함수를 작용시켜, 확대키 (n)→확대키 (1)의 순번으로, 즉 쉴 사이 없이 확대키를 생성해 간다.

그렇지만, 최초로 암호화시와 동일한 확대키(Rn)의 생성처리를 행하는 불필요한 시간 때문에, 복호가 개시될 때까지의 지연시간이 발생한다고 하는 문제점이 있었다.

발명이 이루고자 하는 기술적 과제

이상 설명한 바와 같이, 종래의 기술에서는 확대키를 역순으로 생성할 수는 없으므로, 복호처리에 앞서 모든 확대키를 생성하여 기억해 둘 필요가 있지만, 예컨대 IC카드와 같이 빈약한 하드웨어 환경하에서는 복호에 필요한 확대키를 모두 격납할 만큼의 기억영역의 여유가 없다고 하는 문제점이 있었다.

쉴 사이 없는 키 생성에 의해 이 문제를 회피하기 위해서는, 일단 암호화시와 동일한 확대키 생성처리를 행하여 최종 라운드에서 라운드함수를 작용시켜 얻어지는 출력값을 구한 후에, 새삼 그 출력값에 역의 라운드방향으로 각 라운드함수의 역함수를 작용시켜 가는 것이 필요하게 되지만, 이 경우도 복호가 개시될 때까지의 지연시간을 피할 수 없다고 하는 문제점이 있었다.

본 발명은 상술한 사정에 대처하기 위해 이루어진 것으로, 그 목적은 확대키 생성을 위한 지연시간의 발생을 회피하거나 또는 작게 하고, 또한 쉴 사이 없는 키 생성을 가능하게 한 암호화장치, 복호장치 및 확대키 생성장치, 확대키 생성방법 및 기록매체를 제공하는 것을 목적으로 한다.

발명의 구성 및 작용

본 발명의 한 태양(態樣)에 의하면, 암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 암호화장치로,

초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성하는 복수단의 라운드 처리부와,

상기 복수단의 라운드 처리부의 적어도 일부에서 생성된 상기 서브키에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력하는 복수의 확대키 처리부를 구비하고,

상기 최종단의 출력이 초단에 입력된 공통키와 같은 것을 특징으로 한다.

본 발명의 한 태양에 의하면, 암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 복호장치로,

초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성하는 복수단의 라운드 처리부와,

상기 복수단의 라운드 처리부의 적어도 일부에서 생성된 상기 서브키에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력하는 복수의 확대키 처리부를 구비하고,

상기 최종단의 출력이 초단에 입력된 공통키와 같은 것을 특징으로 한다.

(실시형태)

이하, 도면을 참조하여 본 발명의 실시형태를 설명한다.

제1실시형태

본 발명은, 암호화시와 복호시에서 역의 순번으로 확대키를 이용하는 공통키 암호방식의 모두에 적용 가능하지만, 이하에서는 소정 비트길이의 블록 데이터에 대해 축차적으로 각 확대키를 이용한 데이터 교반처리를 행하여 가는 공통키 블록 암호방식을 예로 들어 설명한다.

이하에서 참조하는 각 도면에서는 (당해 암호에 주목하여 설명하기 위해) 암호대상으로 되는 데이터를 평문으로서 나타내고 있지만, 물론 암호대상으로 되는 데이터가 이전에 동일 또는 다른 암호방식에 의해 암호화된 것이어도 좋다. 본 암호방식은, 하드웨어에 의해서도 소프트웨어에 의해서도 실현가능하고, 이하에 나타낸 구성례는 암호화장치(복호장치)의 기능블록도로서도 성립하며, 또 암호알고리즘(복호알고리즘)의 기능모듈도 또는 플로우챠트도로서도 성립한다.

도 1에 본 발명의 한 실시형태에 따른 암호화장치의 구성례를 나타낸다. 본 암호화장치는, 데이터 교반부(1)와 확대키 생성부(3)를 갖추고 있다.

확대키 생성부(3)는, 복수의 라운드 처리부(31_1 31_n; 경우에 따라서는 31_{n+1} 도 갖춘다)를 갖추고 있다.

제1단의 라운드 처리부(31_1)는, 공통키(kc)에 제1단의 라운드함수(f_1)를 작용시켜 제1서브키 $kc_1 = f_1(kc)$를 출력한다.

제2단의 라운드 처리부(31_2)는, 전단(여기서는 제1단)의 라운드 처리부(31_1)가 출력한 서브키(kc_1)에 제2단의 라운드함수(f_2)를 작용시켜 서브키 $kc_2 = f_2(kc_1) = f_2(f_1(kc))$를 출력한다.

도시하지 않은 제3단으로부터 제(n-1)단의 라운드 처리부도 마찬가지이다.

제n단의 라운드 처리부(31_n)는, 전단(여기서는 제(n-1)단)의 라운드 처리부(31_{n-1} ; 도시하지 않음)가 출력한 서브키(kc_{n-1})에 제n단의 라운드함수(f_n)를 작용시켜 서브키 $kc_n = f_n(kc_{n-1}) = f_n(f_{n-1}(\cdots f_2(f_1(kc))\cdots))$를 출력한다.

본 실시형태에서는, 제(n+1)단의 라운드 처리부(31_{n+1})에서 제n단의 라운드 처리부(31_n)가 출력한 서브키(kc_n)에 라운드함수(f_{n+1})를 작용시키고, 이에 따라 얻어진 출력값 $kc_n = f_{n+1}(kc_{n-1}) = f_{n+1}(f_n(f_{n-1}(\cdots f_2(f_1(kc))\cdots)))$이 공통키(kc)와 같아지게 한다.

이 제n단의 라운드함수(f_{n+1})의 역함수(f_{n+1}^{-1})가 복호장치에서의 확대키 생성부의 초단의 라운드 처리부의 라운드함수로 된다.

암호화장치에서의 확대키 생성부에는 라운드함수(f_{n+1})의 라운드 처리부를 갖추고 있지 않아도 상관없지만, 갖추어 놓으면 암호화장치와 복호장치에서 키 생성부의 구성이 동일하게 되므로, 암호화기능과 복호기능의 양쪽의 기능을 겸비시킨 장치에 있어서는, 암호화시와 복호시에서 1개의 확대키 생성부를 겸용함으로써, 장치 규모를 삭감할 수 있는 이점이 있다.

확대키 생성부(3)는 복수의 확대키 변환부(33_1 33_n)를 갖추고 있다.

제1단의 확대키 변환부(33_1)는 제1단의 라운드 처리부(31_1)의 출력(kc_1)의 전부 또는 일부에 제1단의 확대키 변환함수(c_1)를 작용시켜 제1단의 확대키(k_1)를 생성한다.

제2단의 확대키 변환부(33_2)는 제2단의 라운드 처리부(31_2)의 출력(kc_2)의 전부 또는 일부에 제2단의 확대키 변환함수(c_2)를 작용시켜 제2단의 확대키(k_2)를 생성한다.

도시하지 않은 제3단으로부터 제(n-1)단의 확대키 변환부도 마찬가지이다.

제n단의 확대키 변환부(33_n)는 제n단의 라운드 처리부(31_n)의 출력(kc_n)의 전부 또는 일부에 제n단의 확대키 변환함수(c_n)를 작용시켜 제n단의 확대키(k_n)를 생성한다.

데이터 교반부(1)는 종속접속된 복수의 (예컨대 라운드함수에 의한) 교반처리부(11_1 11_n)를 갖춘다.

제1단의 교반처리부(11_1)는, 암호화의 대상으로 되는 블록 데이터(평문)를 입력하고, 확대키(k_1)를 이용하여 교반처리(R_1)를 행한다.

제2단의 교반처리부(11_2)는, 제1단의 교반처리부(11_1)로부터 출력된 블록 데이터를 입력하고, 확대키(k_2)를 이용하여 교반처리(R_2)를 행한다.

도시하지 않은 제3단으로부터 제(n-1)단의 교반처리부도 마찬가지이다.

제n단의 교반처리부(11_n)는, 제(n-1)단의 교반처리부로부터 출력된 블록 데이터를 입력하고, 확대키(k_n)를 이용하여 교반처리(R_n)를 행한다. 제n단의 교반처리부(11_n)의 출력이 구해야 할 암호문으로 된다.

복수의 교반처리에 이용되는 라운드함수는, 모두 다른 것이어도, 모두 동일한 것이어도, 다른 것과 동일한 것이 혼재하는 것이어도 좋다. 복수의 라운드함수를 다른 것으로 하는 경우에, 함수를 달리하는 방법 외에, 기본적으로는 동일한 함수이지만 단(段: stage)에 따라 다른 정수에 의존하는 것으로 하는 방법 등이 있다.

복수의 라운드함수는, 모두 선형 함수이어도 임의의 함수이어도 좋지만, 그들 중의 적어도 하나를 비선형 함수로 하는 것이 바람직하다. 2 이상의 라운드함수 혹은 모든 라운드함수를 비선형 함수로 해도 좋다.

라운드함수는, 변환테이블을 이용하여 실현하는 방법, 행렬연산이나 그 밖의 연산으로 실현하는 방법, 실질 회로에 의해 실현하는 방법 등, 여러 가지의 구성방법이 있다.

이들 점은 복수의 확대키 변환함수에 대해서도 마찬가지이다.

각 단의 확대키 생성부(33_1 33_n)의 변형례로서는, 입력된 서브키 또는 그 일부를 그대로 확대키로서 출력하는 구성으로 하는(혹은, 서브키를 데이터 교반부(1; 또는 후술하는 스위칭회로(15))에 직결하는) 것도 가능하다.

블록 데이터의 데이터길이와 공통키(kc)의 데이터는, 동일해도 좋고, 다른 것이어도 좋다. 확대키의 데이터길이와 블록 데이터의 데이터길이는, 동일해도 좋고, 다른 것이어도 좋다. 서브키의 데이터길이와 확대키의 데이터길이는, 동일해도 좋고, 다른 것이어도 좋다.

도 2에 본 발명의 한 실시형태에 따른 복호장치의 구성례를 나타낸다. 본 복호장치는, 데이터 교반부(2)와 확대키 생성부(4)를 갖추고 있다. 도 2의 복호장치는, 도 1의 암호화장치의 역변환을 행하는 기능을 갖춘 것이다.

확대키 생성부(4)는, 복수의 라운드 처리부(42_2 42_{n+1})(경우에 따라서는 42_1도 갖춘다)를 갖추고 있고, 도 1의 암호화장치의 확대키 생성부(2)의 복수의 라운드함수의 각각의 역함수를 역의 순번으로 작용시키는 것이다.

라운드 처리부(42_{n+1})는 공통키 $kc = f_{n+1}(kc_n) = f_{n+1}(f_n(f_{n-1}(\cdots f_2(f_1(kc))\cdots)))$에 라운드함수($f_{n+1}^{-1}$)을 작용시켜 서브키 $kc_n = f_{n+1}^{-1}(kc) = f_{n+1}^{-1}(f_{n+1}(f_n(f_{n-1}(\cdots f_2(f_1(kc))\cdots)))) = f_n(f_{n-1}(\cdots f_2(f_1(kc))\cdots))$을 출력한다.

라운드 처리부(42_n)는 전단의 라운드 처리부(42_{n+1})가 출력한 서브키(kc_n)에 라운드함수(f_n^{-1})을 작용시켜 $kc_{n-1} = f_n^{-1}(\cdots f_2(f_1(kc))\cdots)$을 출력한다.

도시하지 않은 라운드 처리부 42_{n-1} 로부터 42_3도 마찬가지이다.

라운드 처리부(42_2)는, 전단의 라운드 처리부(42_3)가 출력한 서브키 $kc_2 = f_2(f_1(kc))$에 라운드함수(f_2^{-1})를 작용시켜 $kc_1 = f_1(kc)$를 출력한다.

라운드 처리부(42_1)에서 전단의 라운드 처리부(42_2)가 출력한 서브키(kc_1)에 라운드함수(f_1^{-1})를 작용시키고, 이로써 얻어진 출력값은 공통키(kc)와 같아진다.

라운드함수(f_1^{-1})의 역함수(f_1)가 암호화장치에서의 확대키 생성부의 초단의 라운드 처리부의 라운드함수로 된다. 도 2의 구성례에서는, 확대키 생성부에서는 라운드함수(f_1^{-1})의 라운드 처리부(42_1)를 갖추지 않아도 상관없지만, 갖추어 놓으면 암호화장치와 복호장치에서 키 생성부의 구성이 동일하게 되므로, 암호화기능과 복호기능의 양쪽의 기능을 겸비시킨 장치에 있어서는, 암호화시와 복호시에서 1개의 확대키 생성부를 겸용함으로써, 장치 규모를 삭감할 수 있는 이점이 있다.

확대키 생성부(4)는 복수의 확대키 변환부(44_1 44_n)를 갖추고 있다. 이 부분은 도 1의 암호화장치의 대응하는 부분과 동일한 처리내용으로 된다.

데이터 교반부(2)는 종속접속된 복수의 (예컨대 라운드함수에 의한) 교반처리부(22_1 22_n)를 갖춘다.

최종단의 교반처리부(22_n)는, 복호의 대상으로 되는 블록 데이터(암호문)를 입력하고, 확대키(k_n)를 이용하여 암호화장치의 교반처리(R_n)의 역변환으로 되는 교반처리(R_n^{-1})를 행한다.

마찬가지로, 제(n-1)단으로부터 제2단의 교반처리부(22_{n-1} 22_2)도 순차, 후단의 교반처리부로부터 출력되는 블록 데이터를 입력하고, 확대키(k_{n-1}, …, k_2)를 이용하여 교반처리(R_{n-1}^{-1}, …, R_2^{-1})를 행한다.

제1단의 교반처리부(22_1)는, 제2단의 교반처리부(22_2)로부터 출력된 블록 데이터를 입력하고, 확대키(k_1)를 이용하여 암호화장치의 교반처리(R_1)의 역변환으로 되는 교반처리(R_1^{-1})를 행한다. 제1단의 교반처리부(22_1)의 출력이 구해야 할 복호결과(블록 데이터(평문))로 된다.

즉, 도 2에 나타낸 바와 같이, 복호시에 있어서 곧바로 암호화시와는 역의 순번으로의 확대키의 생성에 들어가 확대키를 잇달아 생성해 갈 수 있다.

이상과 같이, 암호화시의 확대키 생성을 위한 라운드함수의 계열(다만, 최종단은 갖추지 않은 것도 있음)과, 그 역함수로 되는 복호시의 확대키 생성을 위한 라운드함수의 계열(다만, 최종단은 갖추고 있지 않은 것도 있음)에 대해, 암호화시의 최종단의 출력에 상당하는 값이 근본의 공통키와 일치하도록 라운드함수의 계열을 설정함으로써, 암호화시와 복호시의 양쪽에 있어서 종래와 같은 불필요한 지연시간이나 기억용량의 소비 없이 공통키로부터 쉴 사이 없이 확대키를 생성하는 것이 가능하게 된다.

다음으로, 도 1의 암호화장치, 도 2의 복호장치의 확대키 생성부의 복수의 라운드 처리부에 의한 라운드함수의 계열에 대해 설명한다. 암호화장치에서의 라운드함수의 계열과 복호장치에서의 라운드함수의 계열은 역함수의 관계로 되므로, 한쪽이 결정되면, 다른쪽도 결정되게 된다. 여기서는, 암호화장치 쪽을 예로 들어 설명한다.

라운드함수의 계열($f_1, f_2, \cdots, f_{n+1}$)에 대해, 그 라운드함수의 계열의 내용, 혹은 각각의 순번의 라운드함수의 내용은 당해 라운드함수의 계열이 전체로서 공통키를 입력하고 공통키와 동일한 값을 출력하는 조건을 만족하는 범위 내에 있어서 적절히 설정가능하고, 각종의 변형이 있다. 이하, 라운드함수의 계열의 변형의 몇가지를 예시열거적으로 설명한다.

라운드 트립 구성

여기서는, 라운드함수의 계열의 단수를 2r단으로 한다(전술한 바와 같이, 2r단째의 라운드함수는 갖추지 않은 것이 있음).

라운드함수의 계열을 구성하는 하나의 방법은, $0 \leq i \leq r$을 만족하는 모든 i에 대해 제(r+i)단째의 단함수를, 제(r-i+1)단째의 단함수의 역함수로 되어 있다고 하는 관계를 만족하도록 구성하는 방법이다.

예컨대, 라운드함수의 계열을 8단, 즉

$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8$

로 하고, f_1 f_4를 임의의 라운드함수로 하며, $f_5 = f_4^{-1}$, $f_6 = f_3^{-1}$, $f_7 = f_2^{-1}$, $f_8 = f_1^{-1}$ 으로 하면,

$f_1, f_2, f_3, f_4, f_4^{-1}, f_3^{-1}, f_2^{-1}, f_1^{-1}$

이라고 하는 순번의 계열로 된다. 즉, 공통키를 입력으로 하여, f_1, f_2, f_3, f_4, f_4 - 1, $f_3^{-1}, f_2^{-1}, f_1^{-1}$ 을 잇달아 작용시킴으로써, 최종단의 출력이 공통키와 일치하게 된다.

이러한 구성을 라운드 트립 구성이라 부르기로 한다. 이 상태를 도 3에 개념적으로 나타낸다.

라운드 트립 구성을 채용한 경우, 암호화장치에서의 라운드함수의 계열과 복호장치에서의 라운드함수의 계열이 동일하게 된다.

상기 예의 경우, 암호화장치에서의 8단의 라운드함수를,

$f_1, f_2, f_3, f_4, f_4^{-1}, f_3^{-1}, f_2^{-1}, f_1^{-1}$

으로 하면, 복호장치에서의 8단의 라운드함수는, 이 역함수로 되므로,

$(f_1^{-1})^{-1}, (f_2^{-1})^{-1}, (f_3^{-1})^{-1}, (f_4^{-1})^{-1}, (f_4)^{-1}, (f_3)^{-1}, (f_2)^{-1}, (f_1)^{-1}$

이고, 따라서

$f_1, f_2, f_3, f_4, f_4^{-1}, f_3^{-1}, f_2^{-1}, f_1^{-1}$

으로 되어, 양자가 일치함을 알 수 있다.

암호화장치에 있어서도 최종단의 라운드함수(상기의 예에서는, f_1^{-1})을 갖추지 않아도 좋고, 복호장치에 있어서도 최종단의 라운드함수(상기의 예에서는, f_1)를 갖추지 않아도 좋지만, 모두 최종단의 라운드함수를 갖추면 구성이 동일하게 되므로, 암호화기능과 복호기능의 양쪽의 기능을 겸비시키는 장치에 있어서는, 암호화시와 복호시에서 1개의 확대키 생성부를 겸용함으로써, 장치 규모를 삭감하는 것도 가능하다.

이 구성에 있어서, 라운드함수의 계열의 전반의 각 라운드함수는, 모두 다른 것이어도, 모두 동일한 것이어도, 다른 것과 동일한 것이 혼재하는 것이어도 좋다.

예컨대, 라운드함수의 계열의 전반의 각 라운드함수가 모두 동일한 것인 경우, 단수를 8단으로 하면, 암호 측과 복호 측의 어느 것에 있어서도,

$f_1, f_1, f_1, f_1, f_1^{-1}, f_1^{-1}, f_1^{-1}, f_1^{-1}$

이라고 하는 계열로 된다.

그런데, 라운드 트립 구성을 채용한 경우, 라운드함수의 계열에 있어서 역함수의 관계에 있는 대응부분의 서브키가 동일하게 된다. 따라서, 동일한 서브키에 동일한 확대키 변환함수를 작용시키면, 동일한 확대키가 생성된다. 그래서, 이것을 피하기 위해, 라운드함수의 계열에 있어서 역함수의 관계에 있는 대응부분에 대한 2개의 확대키 변환부의 확대키 변환함수로서 상이한 것을 이용하도록 해도 좋다.

예컨대, 8단의 라운드함수의 계열을,

$f_1, f_2, f_3, f_4, f_4^{-1}, f_3^{-1}, f_2^{-1}, f_1^{-1}$

로 하고, f_1의 출력을 이용하는 확대키 변환함수를 c_1, …, f_2^{-1}의 출력을 이용하는 확대키 변환함수를 c_7로 하면, 확대키 변환함수 c_1과 확대키 변환함수 c_7을 다르게 하도록 해도 좋다. c_2와 c_6, c_3와 c_5에 대해서도 마찬가지이다.

루프 구성

라운드 트립 구성에서는 라운드함수의 계열의 후반을 그 전반의 역함수로 했지만, 라운드함수의 계열 중의 부분계열로서 라운드 트립 구성에 상당하는 부분을 전혀 갖추지 않는 구성도 가능하다.

이러한 구성을 루프 구성이라 부르기로 한다. 이 상태를 도 4에 개념적으로 나타낸다.

라운드 트립 구성에서는 라운드함수의 계열의 단수를 우수단으로 했지만, 루프 구성에서는 라운드함수의 계열의 단수는 우수단이어도, 기수단이어도 좋다.

예컨대, 라운드함수의 계열을 8단으로 한 경우에, 공통키를 입력으로 하여

$f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8$

을 잇달아 작용시킴으로써, 최종단의 출력이 공통키와 일치하게 된다. 이 경우, 그 역함수는,

$f_8^{-1}, f_7^{-1}, f_6^{-1}, f_5^{-1}, f_4^{-1}, f_3^{-1}, f_2^{-1}, f_1^{-1}$

이고, 공통키를 입력하면, 최종단의 출력이 공통키와 일치하게 된다.

예컨대, 라운드함수의 계열의 전반의 각 라운드함수가 모두 동일한 것인 경우, 단수를 8단으로 하면, 암호 측에서는,

$f_1, f_1, f_1, f_1, f_1, f_1, f_1, f_1$

이라고 하는 계열로 되고,

복호 측에서는,

$f_1^{-1}, f_1^{-1}, f_1^{-1}, f_1^{-1}, f_1^{-1}, f_1^{-1}, f_1^{-1}, f_1^{-1}$

이라고 하는 계열로 된다.

이러한 조건을 만족하는 함수는, 시프트연산, 행렬연산, 갈루아체 연산 등, 여러 가지의 것이 있다.

라운드 트립/루프 복합구성

라운드함수의 계열로서는, 그 부분계열로서 라운드 트립 구성에 상당하는 부분과, 루프 구성에 상당하는 부분을 복합적으로 갖춘 구성도 가능하다.

이하, 라운드 트립 구성 부분을 도 3의 표기방법으로 나타내고, 루프 구성 부분을 도 4의 표기방법으로 나타내는 것으로 하여, 도 5 도 9에 몇가지의 변형을 예시한다.

도 5의 예는, 라운드 트립 구성 부분의 도중에 라운드 트립 구성 부분을 포함하는 네스팅(nesting: 내포)구조로 되어 있는 것이다. 도 5의 경우의 라운드함수의 계열을 예시하면,

$a_1 a_2 a_3 b_1 b_2 b_2^{-1} b_1^{-1} a_4 a_5 a_6 a_6^{-1} a_5^{-1} a_4^{-1} a_3^{-1} c_1 c_2 c_2^{-1} d_1 d_1^{-1} c_1^{-1} a_2^{-1} a_1^{-1}$

로 된다.

이 예의 경우, $a_1 a_2 a_3 a_4 a_5 a_6 a_6^{-1} a_5^{-1} a_4^{-1} a_3^{-1} a_2^{-1} a_1^{-1}$ 이라고 하는 라운드 트립 구성 중에, $b_1 b_2 b_2^{-1} b_1^{-1}$ 이라고 하는 라운드 트립 구성과, $c_1 c_2 c_2^{-1} c_1^{-1}$ 이라고 하는 라운드 트립 구성이 들어가 있고, 더욱이 $c_1 c_2 c_2^{-1} c_1^{-1}$ 이라고 하는 라운드 트립 구성 중에 $d_1 d_1^{-1}$ 이라고 하는 라운드 트립 구성이 들어가 있다.

도 6의 예는, 루프 구성 부분의 도중에 루프 구성 부분을 포함하는 네스팅구조로 되어 있는 것이다. 도 6의 경우의 라운드함수의 계열을 예시하면,

$s_1 s_2 s_3 s_4 t_1 t_2 t_3 s_5 s_6 s_7 s_8$

로 된다.

이 예의 경우, $s_1 s_2 s_3 s_4 s_5 s_6 s_7 s_8$ 이라고 하는 루프 구성 중에, $t_1 t_2 t_3$ 라고 하는 루프 구성이 들어가 있다.

도 7의 예는, 루프 구성 부분의 도중에 라운드 트립 구성 부분을 포함하는 구조로 되어 있는 것이다. 도 7의 경우의 라운드함수의 계열을 예시하면,

$s_1 s_2 s_3 s_4 s_5 a_1 a_2 a_3 a_3^{-1} a_2^{-1} a_1^{-1} s_6 s_7 s_8 s_9$

로 된다.

이 예의 경우, $s_1 s_2 s_3 s_4 s_5 s_6 s_7 s_8 s_9$ 라고 하는 루프 구성 중에, $a_1 a_2 a_3 a_3^{-1} a_2^{-1} a_1^{-1}$ 이라고 하는 라운드 트립 구성이 들어가 있다.

도 8의 예는, 라운드 트립 구성 부분의 도중에 루프 구성 부분을 포함하는 구조로 되어 있는 것이다. 도 8의 경우의 라운드함수의 계열을 예시하면,

$a_1\ a_2\ a_3\ a_4\ a_5\ a_6\ s_1\ s_2\ s_3\ s_4\ a_6^{-1}\ a_5^{-1}\ a_4^{-1}\ a_3^{-1}\ a_2^{-1}\ a_1^{-1}$

로 된다.

이 예의 경우, $a_1\ a_2\ a_3\ a_4\ a_5\ a_6\ a_6^{-1}\ a_5^{-1}\ a_4^{-1}\ a_3^{-1}\ a_2^{-1}\ a_1^{-1}$ 이라고 하는 라운드 트립 구성 중에, $s_1\ s_2\ s_3\ s_4$ 라고 하는 루프 구성이 들어가 있다.

도 9의 예는 4개의 라운드 트립 구성 부분과 2개의 루프 구성 부분을 지닌 것이다.

물론, 이들 이외에도 라운드 트립 구성 부분과 루프 구성 부분의 조합방법, 혹은 계층구조의 취득방법 등 여러 가지의 변형이 가능하다.

그런데, 도 1, 도 2의 구성례는 데이터 교반부에서 필요로 하는 수만큼 확대키를 생성하는 것을 상정(想定)한 것이었지만, 데이터 교반부에서 필요로 하는 수를 넘는 수만큼의 확대키를 생성가능하게 하는 라운드함수의 단수를 갖추고, 생성가능한 확대키의 일부를 데이터 교반부에서 사용하는 구성도 가능하다.

이 경우의 도 1, 도 2의 암호화장치/복호장치에 대응하는 구성례를 도 10, 도 11에 각각 나타낸다.

여기서는, 도 1, 도 2와 상위한 점을 중심으로 설명한다. 물론, 라운드함수의 계열은, 상기한 라운드 트립 구성 등을 채용한 것이어도 좋다.

도 10의 5와 도 11의 6은, 확대키(ki)와 교반처리(Rj)의 접속관계를 나타내는 선택부로, 그 몇가지의 구체예를 도 12 도 15에 예시하고 있다. 본 실시형태에서는, 도 10의 5의 선택부와 도 11의 6의 선택부는 동일하게 된다.

초단의 라운드함수로의 입력으로서의 공통키와, 최종단의 라운드함수로부터 출력되는 것으로 되는 공통키의 양쪽 또는 한쪽을, 서브키로서 확대키 생성에 이용해도 좋다. 후자를 이용하는 경우에, 실제로 최종단의 라운드함수의 출력을 이용하도록 해도 좋고, 공통키를 기억해 두고 이것을 이용하도록 해도 좋다.

생성가능한 확대키의 수가 교반처리에서 필요한 확대키의 수보다 많아지도록 구성하고, 확대키(ki)와 교반처리(Rj)를 적절히 대응지운다. 동일한 확대키를 복수의 교반처리에 사용가능하다고 하는 방법과, 1개의 확대키를 교반처리에 대해 배타적으로 사용가능하게 하는 방법이 있다.

사용하지 않는 것으로 되는 확대키는 생성하지 않도록 해도 상관없다. 이 경우에는, 대응하는 확대키 변환부를 갖추지 않아도 상관없다.

이와 같이 생성할 수 있는 확대키의 일부만을 데이터 교반에 사용하는 구성은 공격에 대한 안전성의 면에서 효과적이다.

이하, 여러 가지의 변형에 대해 설명한다.

먼저, 데이터 교반부의 교반처리의 단수가 n이고, (확대키 변환부를 갖춘 것으로 하여) 생성가능한 확대키의 개수가 m ($m > n$)인 경우에, 기본적으로는 확대키의 중복사용을 허가하지 않는 구성에서는, m개의 확대키 중에서 n개의 확대키를 임의로 선택한 모든 조합이 가능하다. 여기서는, 생성되는 순번에 확대키를 사용하는 것으로 한다.

확대키의 중복사용을 허가하는 구성에서는, 기본적으로는 n^m 가지의 조합이 가능하다.

어느 확대키를 선택할 것인지에 대해서는, 랜덤하게 선택하는 방법 외에, 소정의 기준에 따라 선택하는 방법이 있다.

SQUARE 공격이라고 불리는 특수한 공격에서는, 종래의 암호방식에 대해 초단(혹은 최초로부터의 연속하는 수개의 단) 혹은 최종단(혹은 최종단까지의 연속하는 수개의 단)의 확대키 중 일부의 비트에 대해, 전수탐색(全數探索)이 행하여진다. 이 경우, 초단과 최종단의 확대키가 동일한 경우에는, 탐색의 공간이 작아져 버려 해독될 가능성이 높아진다.

초단의 확대키 변환부에 의해 얻어지는 확대키(이하, 초단의 확대키라 부른다)와 최종단의 확대키 변환부에 의해 얻어지는 확대키(이하, 최종단의 확대키라 부른다)에 대해서는, 고작 한쪽만을 데이터 교반에 사용하도록 해도 좋다(어느 한쪽만을 데이터 교반에 사용하는 방법과, 모두 데이터 교반에 사용하지 않는 방법이 있다).

마찬가지로, 초단으로부터의 연속하는 수개의 단의 확대키와 최종단까지의 연속하는 수개의 단의 확대키의 범위에 있어서, 초단 또는 최종단으로부터의 단수를 동일하게 하는 2개의 단의 확대키의 조(組)에 대해, 어느 조에 있어서도 고작 1개만을 데이터 교반에 사용하도록 해도 좋다. 이 경우에, 사용하는 것 혹은 사용하지 않는 것에 대한 선택방법으로는 각종의 변형을 생각할 수 있다. 예컨대, 어느 조에 있어서도, 어느 한쪽을 사용하는 것으로 하는 경우에, 각 조마다 사용하는 쪽(또는 사용하지 않는 쪽)을 랜덤하게 선택해도 좋고, 예컨대 전반과 후반에서 교대로 되도록 선택하는 등 일정의 기준에 따라 선택하도록 해도 좋다. 예컨대, 각 조마다 단의 순번이 앞쪽의 것을 사용할 것인지, 단의 순번이 뒤쪽의 것을 사용할 것인지, 모두 사용하지 않을 것인지를 랜덤하게 선택해도 좋고, 일정의 기준에 따라 선택하도록 해도 좋다.

예컨대, 도 12에 나타낸 바와 같이, 확대키가 15단분 생성가능하고, 교반처리가 9단 있는 경우에, 초단의 확대키(k_1)와 최종단의 확대키(k_{15}) 중에서는 k_{15} 를 사용하고, 그 1개 내측의 단인 k_2와 k_{14} 중에서는 k_2를 사용하며, 마찬가지로 k_3과 k_{13} 중에서는 k_{13} 을 사용하고, k_4와 k_{12} 중에서는 k_4를 사용하며, k_5와 k_{11} 중에서는 k_{11} 을 사용하고, k_6과 k_{10} 중에서는 k_6을 사용하도록 해도 좋다. 이 경우에 있어서도, 생성되는 순번에 확대키를 사용하도록 하고 있다.

초단으로부터의 연속하는 수개의 단의 확대키와 최종단까지의 연속하는 수개의 단의 확대키의 범위에 대해서는 사용하지 않도록 해도 좋다. 도 13에 이 때의 상태를 나타낸다.

초단과 최종단, 또는 초단으로부터의 연속하는 수개의 단의 확대키와 최종단까지의 연속하는 수개의 단의 확대키의 범위에 대해서는 사용하지 않고, 그 내측의 연속하는 수개의 단의 확대키의 범위에 대해서는 전술한 것과 마찬가지로 대응하는 1조 중의 한쪽만을 사용하도록 해도 좋다. 도 14에 이 때의 상태를 나타낸다.

물론, 이상의 예 외에도, 각종의 변형이 있다.

그런데, 지금까지는 확대키를 생성가능한 순번으로 데이터 교반에 사용하는 것으로서 설명했지만, 메모리 등의 하드웨어 또는 계산시간에 어느 정도의 여유가 있으면, 그 여유에 따라 확대키가 생성되는 순번과 확대키를 데이터 교반에 사용하는 순번을 교체하도록 해도 좋다. 이 순번의 교체는, 도 1이나 도 2의 구성에 있어서도 마찬가지이다. 이 순번의 교체도, 공격에 대한 안전성의 면에서 효과적이다.

도 15에 확대키가 생성되는 순번과 확대키를 데이터 교반에 사용하는 순번을 교체한 예를 나타낸다.

순번을 교체하는 경우에는, 예컨대 먼저 생성된 확대키를 나중에 생성된 확대키보다도 뒤에 사용하기 위해, 일단 메모리에 격납해 두도록 하면 좋다. 1개의 확대키의 순번을 교체할 뿐이라면, 1개의 확대키를 일시 보존하는데 필요한 메모리용량이 증가할 뿐이다.

메모리를 증가시키지 않기 위해서는, 서브키에 라운드함수의 역함수를 작용시켜 필요한 서브키를 복원시키면 좋다. 예컨대, 초단의 라운드함수(f_1)에 의한 서브키(kc_1)로부터 얻어지는 확대키(k_1)를, 2단째의 라운드함수(f_2)에 의한 서브키(kc_2)로부터 얻어지는 확대키(k_2)를 사용한 후에 사용하는 경우, 일단 서브키(kc_2)를 구한 후에 kc_2에 2단째의 라운드함수(f_2)의 역함수(f_2^{-1})를 작용시켜 서브키(kc_1)를 구하고(이에 따라 확대키(k_1)가 얻어짐), 더욱이 서브키(kc_1)에 2단째의 라운드함수($f2_2$)를 작용시켜 서브키(kc_2)를 구하며, 이것에 3단째의 라운드함수(f_3)를 작용시키는 등과 같이 함으로써, 사용순서로 확대키를 생성할 수 있다. 라운드함수 계열이 라운드 트립 구성을 갖는 경우에는, 라운드함수(f_2)의 역함수(f_2^{-1})도 동시에 갖추고 있으므로, 이것을 상기의 처리에 이용하도록 해도 좋다.

그런데, 이상의 확대키의 선택이나 순서의 교체는 고정적인 것이었지만, 이것을 가변으로 하도록 해도 좋다.

이 경우의 도 10, 도 11의 암호화장치/복호장치에 대응하는 구성례를 도 16, 도 17에 각각 나타낸다. 도 중의 7과 8은 디코더이고, 15와 16은 스위칭회로이다.

이 경우, 미리 각각의 교반처리(R_j)로 확대키(k_i)를 대응짓는 접속패턴(도 12 등 참조)을 복수 종류 준비해 두고, 각 패턴을 코드화하여 이것을 확장공통키(kc')로서 근본의 공통키(kc)에 부가한다.

암호화시에는, 확장공통키(kc')는 디코더(7)에 인가되고, 디코더(7)는 확장공통키(kc')를 해독하며, 그 확장공통키(kc')가 나타내는 접속패턴(도 12 등의 패턴)을 실현하도록 스위칭회로(15)에 대한 스위칭 제어를 행한다.

이들의 동작은 복호시도 마찬가지이다.

상기에서는 패턴을 코드화하도록 했지만, 그 대신에 사용하지 않는 확대키의 단수를 나타내는 정보 등, 다른 형태의 정보를 사용하는 것도 가능하다.

이러한 구성도, 공격에 대한 안전성의 면에서 효과적이다.

이상의 각 구성례에 있어서, 도 18에 나타낸 바와 같이 초단 및 최종단에 모의 아다마르(Hadamard)변환과 같은 보조함수(13)를 삽입하도록 해도 좋다. 이 경우에, 초단의 보조함수와 최종단의 보조함수에 동일한 확대키(예컨대, 초단의 확대키)를 이용하도록 해도 좋다. 모의 아다마르변환은, 예컨대 블록 데이터의 왼쪽 반분과 오른쪽 반분의 산술가산을 취한 것을 새로운 왼쪽 반분으로 하고, 이 새로운 왼쪽 반분과 오른쪽 반분의 산술가산을 취한 것을 새로운 오른쪽 반분으로 하는 처리가 이것에 해당한다. 도 18은 암호화장치에 보조함수를 삽입한 예이지만, 복호장치에도 마찬가지로 보조함수를 삽입할 수 있다.

이하에서는, 암호화장치나 복호장치의 확대키 생성부의 라운드 처리부의 변형에 대해 설명한다.

도 19에는 31_1 31_n의 1개의 라운드 처리부의 구성례를 나타낸다. 도 19에 있어서, 101은 8비트의 비선형 변환모듈인 S-박스(S-box)이고, 103은 MDS(Maximum Distance Separable; 최대거리분리)행렬에 기초한 32비트의 교반부이다. 이 예에서는, 서브키의 전부 또는 일부로서 32×k비트의 데이터를 입력하고, 32×k비트의 확대키를 출력하는 것으로, 4병렬의 S-박스(101)에 교반부(103)를 접속한 것을 1단위(102)로 하여, 이것을 k개분 병렬로 설치한 것이다.

물론, 전술한 바와 같이 확대키 생성부는 여러 가지의 구성이 가능하다.

그런데, 어떤 종류의 암호해독에 의해 어떤 단(일반적으로는 최종단)의 확대키(의 일부)가 알려질 우려는 완전히는 부정할 수 없다. 만일 어떤 단의 확대키가 알려져 버린 경우, 확대키 변환부의 역변환을 행함으로써, 그 단의 (라운드함수에 대한) 서브키를 알아 버리고, 그 결과로서 다른 단의 서브키 모두가 알려지는 곳으로 되어, 결국 모든 확대키가 알려지는 곳으로 될 우려가 있다.

그래서, 일부(예컨대 최종단을 포함하는 1단 또는 수개의 단)의 확대키 변환부에 있어서는, 역변환이 용이하지 않은 함수(예컨대, 누승(累乘)함수)나, 역이 일의적(一意的)으로 정해지는 함수(예컨대, 다대일 함수)를 이용하도록 해도 좋다. 이에 따라, 다른 단의 확대키를 용이하게는 알려지지 않도록 하는 것이 가능하게 되어 안전성을 유지할 수 있다. 물론, 전부의 확대키 변환부에 대해 역변환이 용이하지 않은 함수나 역이 일의적으로 정해지는 함수를 이용하도록 해도 좋다.

확대키 변환부에는, 대응하는 단의 서브키의 모든 데이터를 인가해도 좋지만, 그 대신에 대응하는 단의 서브키의 일부만을 건네주는 구성으로 하여 서브키의 모든 데이터를 알 수 없도록 함으로써, 안전성을 유지할 수도 있다.

사이드 채널(side channel) 해석이라 불리는 특수한 공격에서는, 하드웨어로 구성된 암호화장치에 대해 전력, 전자파 등, IC카드 등의 장치로부터 누설되는 정보를 기초로 키의 추측을 행한다. 특히, 데이터 교반처리에서의 어떤 회로에 있어서, 동일한 구성을 갖는 복수의 회로부분이 있고, 그들 회로로의 입력 비트열과 그 회로에서 사용되는 키 비트열(확대키 자체 또는 확대키의 일부의 데이터)이 동일하다면, 사이드 채널 정보(예컨대, 소비전류의 변화)의 동일성으로부터, 그들 회로에 대해 입력된 비트열이 동일했던 것이 추측되어 버린다. 따라서, 확대키 생성에 있어서는, IC카드 등에서 문제로 되는 사이드 채널 해석이 용이하게 되는 키를 생성하지 않는 것이 바람직하다.

그래서, 적어도 처리요소(회로부분)의 입출력의 일부가 집적 관측 혹은 추측이 가능한 상이한 처리요소(회로부분)에 있어서, 동일한 확대키가 사용되지 않는 확대키 생성방법이 유효하다.

모든 확대키가 항시 일치하지 않도록, 확대키 변환부, 또는 확대키 변환부 및 라운드 처리부, 또는 라운드 처리부를 설계하고, 우연히 일치하는 것은 허용하도록 해도 좋다.

모든 확대키가 항시 일치하지 않도록, 확대키 변환부, 또는 확대키 변환부 및 라운드 처리부, 또는 라운드 처리부를 설계함과 더불어, 공통키의 생성시에 모든 확대키가 다른지 어떤지를 체크하여, 모든 확대키가 다르다고 판정된 경우에만 그 공통키를 사용하도록 해도 좋다.

여기서, 확대키의 일치에 대해서는, 여러 가지 레벨을 생각할 수 있다. 예컨대, 2개의 확대키의 모든 비트가 동일한 때에, 일치한다고 판단하도록 해도 좋다. 2개의 확대키의 특정의 바이트 위치의 데이터가 동일한 때에는, 일치한다고 판단하도록 해도 좋다. 2개의 확대키의 모든 비트, 또는 2개의 확대키의 특정의 바이트 위치의 데이터간에 일정의 관계가 있을 때에는, 일치한다고 판단하도록 해도 좋다. 이들 외에도, 여러 가지의 일치판정방법이 가능하다.

이하에서는, 암호화장치나 복호장치의 확대키 생성부의 복수의 라운드 처리부의 변형에 대해 설명한다.

도 20에는 라운드 처리부의 계열의 구성례를 나타낸다. 도 20의 예에서는 3단분을 나타내고 있지만, 각 단의 구성이 소정 단수 종속접속되는 것으로 된다. 도 20에서는, 공통키가 128비트이고, 각 단의 확대키는 64비트인 경우를 예시하고 있다. 도 중, 105는 비선형 사상(F)이고, 107의 기호는 배타적 논리합을 나타내며, 이 105와 107은 도 1의 31_1 31_n의 1개이다.

비선형 사상(F)은 모든 단에서 동일해도 좋고, 단마다 달라도 좋다. 후자의 경우, 기본적으로는 동일한 구성을 갖지만, 단에 따라 다른 정수에 의존하는 것이어도 좋다.

도 20과 도 20의 역함수의 어느 하나를 암호화 측(또는 복호 측)에 이용하는 것도 가능하다.

일반적으로, 차분해독법이나 선형 해독법과 같은 강력한 해독법을 이용해도, 최종단의 확대키 중 고작 수비트를 특정하는 것이 한계이기 때문에, 라운드함수의 계열은 도 20에 나타낸 바와 같은 단순한 페이스텔(Feistel)구조라도 안전성에 큰 문제는 없다고 생각되지만, 만일 보다 강력한 해독법의 출현에 대비하여 보다 안전한 구조를 원한다면, 예컨대 도 21에 나타낸 바와 같은 라운드함수의 계열을 이용해도 좋다.

도 21의 예에서는 2단분을 나타내고 있지만, 각 단의 구성이 소정 단수 종속접속되는 것으로 된다. 도 21에서는, 공통키가 128비트이고, 각 단의 확대키는 64비트인 경우를 예시하고 있다. 도 중, 109와 111과 113은 각각 비선형 함수 f와 g와 h를 나타내고 있고, 115의 기호는 배타적 논리합을 나타내고 있다. 비선형 함수 f와 g와 h는 모두 동일해도, 모두 달라도, 그들 중 일부의 것끼리만 동일해도 좋다.

도 22에 도 21의 비선형 함수 f와 g와 h의 구성례를 나타낸다. 도 22에 있어서, 119는 8비트의 S-박스이고, 121은 MDS행렬에 기초한 32비트의 교반부이다.

도 21에서는, 도 20과 비교하여, 출력되는 128비트로부터 서브키를 일의적으로 결정하는 것이 보다 곤란하게 되어 있다.

도 23에 도 21의 역함수를 나타낸다. 도 21과 도 23의 어느 하나를 암호화 측(또는 복호 측)에 이용하는 것도 가능하다.

이하에서는, 본 발명을 적용한 암호화장치의 한 구체예를 나타낸다.

도 24에 본 암호화장치의 구성례를 나타낸다.

이 암호화장치는, 128비트(혹은 46비트)의 블록암호이고, 공통키는 256비트(혹은 128비트)인 경우를 예로 들고 있다. 라운드함수의 계열은 라운드 트립 구성을 갖는 경우를 예로 들고 있다. 통상의 SPN구조의 S-박스의 부분에 소형의 SPN구조를 재귀적으로 매립한 네스팅형의 SPN구조인 경우를 예로 들고 있다.

도 24에 있어서, 데이터 교반부(202)에서는 라운드함수(DU) 처리부(201)와 라운드함수(DD) 처리부(203)의 반복구조를 취한 후, 라운드함수(DU) 처리부(201)와 라운드함수(DD(woMDSH : without MDSH); 205)와, 라운드함수(EXOR; 207)가 접속되어 있다.

확대키 생성부(204)에서는, 유닛(unit: 단위)(209)과 유닛(211)의 쌍이 1단분의 라운드함수에 상당한다. 다만, 도 24의 예에서는, 유닛(209)과 유닛(209)의 사이 및 유닛(211)과 유닛(211)의 사이가 도 1의 서브키가 아니고, 서브키는 유닛(209)이나 유닛(211)의 내부에 나타나는 구조로 되어 있다.

도 25에 128비트의 블록암호의 경우에서의 도 24의 유닛(201)의 구성례를 나타낸다. 도 25에 있어서, 215는 키 가산을 위한 8비트의 배타적 논리합이고, 217은 8비트의 S-박스이며, 219는 MDS행렬에 기초한 32비트의 교반부이다. 유닛(213)이 4개 병렬로 설치된다.

64비트의 블록암호의 경우에는, 유닛(213)이 2개 병렬로 설치된다.

도 26에 128비트의 블록암호의 경우에서의 도 24의 유닛(203)의 구성례를 나타낸다. 도 26에 있어서, 221은 키 가산을 위한 8비트의 배타적 논리합이고, 223은 16병렬의 8비트의 S-박스이며, 225는 MDS행렬에 기초한 32비트의 교반부이다.

64비트의 블록암호의 경우에는, S-박스(223)가 8개 병렬로 설치된다.

도 27에 128비트의 블록암호의 경우에서의 도 24의 유닛(205)의 구성례를 나타낸다. 도 27에 있어서, 227은 키 가산을 위한 8비트의 배타적 논리합이고, 229는 16병렬의 8비트의 S-박스이다.

64비트의 블록암호의 경우에는, 229의 S-박스가 8개 병렬로 설치된다.

유닛(207)은 128비트의 블록암호의 경우, 유닛(205)으로부터 출력되는 128비트의 블록 데이터에 128비트의 확대키를 가산하기 위한 배타적 논리합이다.

유닛(207)은 64비트의 블록암호의 경우, 유닛(205)으로부터 출력되는 64비트의 블록 데이터에 64비트의 확대키를 가산하기 위한 배타적 논리합이다.

도 28에 공통키의 비트길이가 256비트의 경우에서의 도 24의 확대키 생성부(204)의 구성례를 나타낸다. 도 28은 도 3에 나타낸 라운드 트립 구성에 기초한 것으로 하고, 초단부분과 반환부분에 대해서만 나타내고 있다. 도 중, 231은 비선형 함수(F)이고, 233은 배타적 논리합이며, 235는 단에 따라 다른 정수 const(r)과의 배타적 논리합이다. 231, 237, 239, 241, 243의 유닛에 대해서는 후술한다.

도 29에 공통키의 비트길이가 128비트의 경우에서의 도 24의 확대키 생성부(204)의 구성례를 나타낸다. 도 29도 도 3에 나타낸 라운드 트립 구성에 기초한 것으로 하고, 초단부분과 반환부분에 대해서만 나타내고 있다. 도 중, 251은 비선형 함수(F)이고, 253은 배타적 논리합이며, 255는 단에 따라 다른 정수 const(r)과의 배타적 논리합이다. 251, 257, 259, 261, 263의 유닛에 대해서는 후술한다.

도 30에 도 28의 비선형 함수(231), 도 29의 비선형 함수(251)의 구성례를 나타낸다. 도 중, 2311은 배타적 논리합이고, 2313은 S-박스이며, 2315, 2317에 대해서는 후술한다.

다음에는, 도 28의 $P^{(32)}$ 로 나타낸 유닛(237), 도 29의 $P^{(16)}$ 으로 나타낸 유닛(257), 도 30의 $P^{(16)}$ 으로 나타낸 유닛(2315), 도 30의 $P^{(8)}$ 로 나타낸 유닛(2317)에 대해 설명한다. 도 31에 이들에 공통하는 일반적인 구성례를 나타낸다. 도 중, 265는 배타적 논리합이고, 어떤 i비트와 다른 i비트와의 배타적 논리합을 취하는 조작이 4회 행하여진다. 이 구성을 $P^{(i)}$ 로 표현한 것이 각 도의 $P^{(8)}$, $P^{(16)}$, $P^{(32)}$ 이다. 즉, 도 28의 유닛(237)은 도 31의 구성에서 i = 32로 한 것이고, 도 29의 유닛(257)은 도 31의 구성에서 i = 16으로 한 것이며, 도 30의 유닛(2315)은 도 31의 구성에서 i = 16으로 한 것이고, 도 30의 유닛(2317)은 도 31의 구성에서 i = 8로 한 것이다.

도 32에 도 31의 $P^{(i)}$ 의 역함수인 $(P^{(i)})^{-1}$ 의 구성례를 나타낸다. 도 중, 267은 배타적 논리합이다. 도 28의 유닛(243)은 도 31의 구성에서 i = 32로 한 것이고, 도 29의 유닛(263)은 도 31의 구성에서 i = 16으로 한 것이다.

도 30은 128비트의 블록암호의 경우였지만, 64비트의 블록암호의 경우, 즉 도 29의 비선형 함수(251)의 경우에는 도 30에 있어서 $P^{(8)}$ 을 $P^{(4)}$ 로 하고, $P^{(16)}$ 을 $P^{(8)}$ 로 하면 좋다.

다음에, 도 28, 도 29, 도 30의 "5"로 나타낸 빗금이 쳐진 유닛(239, 259, 2313), 도 28, 도 29의 "B"로 나타낸 빗금이 쳐진 유닛(241, 261)에 대해 설명한다.

도 33에 "5"로 나타낸 유닛 및 "B"로 나타낸 유닛의 구성례를 나타낸다. 양자의 차이는, 도 33의 유닛(269)에서의 함수의 내용이다.

도 33의 구성은, 입력에 갈루아체 $GF(2^4)$의 근본 5 또는 B를 곱하는 것이다.

즉, 입력으로 되는 32비트를, 4조의 8비트로 나누고, 8비트 데이터의 동일한 위치(예컨대 도 33에서는 최상위비트의 최하위비트를 예로 들어 나타내고 있음)의 1비트를 모아 이것을 1조 4비트의 데이터로 하며, 8조의 4비트 데이터의 각각을 $GF(2^4)$의 근본으로 간주한다. 그리고, 각각의 4비트 데이터에 대해 각각의 유닛(269)에 의해 (갈루아체 상의 승산에 따라) 5 또는 B를 곱한 후에, 각각의 비트를 각각 대응하는 것의 위치로 돌려 보낸다.

상기에서는, 동일한 위치의 비트를 취출하여 처리를 행하는 것으로서 설명했지만, 다른 위치의 비트를 (배타적으로) 취출하여 처리를 행하는 것도 가능하다.

갈루아체 상의 승산은, 색인에 의해서도, 연산에 의해서도, 실질 회로에 의해서도 좋다.

도 34a에 도 33의 유닛(269)의 부분의 구성례, 즉 $GF(2^4)$상의 승산의 결선표현(결선패턴)을, 근본 5에 대해 나타낸다. 도 34b에 도 33의 유닛(269)의 부분의 구성례, 즉 $GF(2^4)$상의 승산의 결선표현(결선패턴)을 근본 B에 대해 나타낸다. 전술한 바와 같이, 결합부분(271)에서는 배타적 논리합이 이루어진다. 즉, 이 경우 도 28, 도 29의 5로 나타낸 유닛(239, 259)에 대해서는 도 33 및 도 34a에 의해 구성가능하고, 도 29, 도 30의 B로 나타낸 유닛(241, 261)에 대해서는 도 34a 및 도 34b에 의해 구성가능하다.

도 35에 갈루아체 $GF(2^4)$상의 승산의 결선표현(결선패턴)을, $GF(2^4)$의 1 F의 근본의 각각에 대해 나타낸다. 한편, 결합부분에서는 배타적 논리합이 이루어진다.

그런데, 데이터 교반부의 확대키를 작용시키는 대상이 기지(旣知) 혹은 비교적 용이하게 추측가능한 부분에서 사용되는 확대키, 예컨대 데이터 교반부의 최초의 배타적 논리합 연산이나, 출력과 키의 추측으로부터 데이터의 추측이 가능하게 되는 최후의 키 가산 전의 배타적 논리합 연산부로의 확대키에 있어서, 다른 위치의 연산요소단위(이 예의 경우, 8비트 단위)로 확대키가 항상 일치하는 것이나 항상 일정의 관계를 갖는 것을 방지하도록 하면 바람직하다.

한 구성례로서는, 상기의 데이터 교반부의 최초의 배타적 논리합 연산에서 사용되는 확대키와, 최후의 키 가산 전의 배타적 논리합 연산부로의 확대키에 요소단위(이 예의 경우, 8비트 단위)에서 항상 성립하는 일치가 발생하지 않도록, 확대키를 생성한다(또는 공통키를 선택한다). 이에 따라, 사이드 채널 해석을 용이하게 하는 확대키의 일치나 일정의 관계의 성립을 방해할 수 있다.

도 24에 대응하는 복호장치의 구성은, 데이터 교반부에 대해서는 도 24의 데이터 교반부의 역함수로 된다. 확대키 생성부에 대해서는, 암호 측과 복호 측 모두 최종단의 라운드함수를 갖춘 경우에는, 도 24의 확대키 생성부와 마찬가지의 구성으로 된다. 물론, 암호 측도 복호 측도 각각 최종단의 라운드함수를 갖추지 않아도 좋다.

도 1 도 34a, 도 34b를 참조하면서 설명해 온 이상과 같은 실시형태에 있어서, 128비트 등의 특정의 비트길이를 예로 들었지만, 물론 어떠한 비트길이의 블록 데이터라도 적용가능하다.

데이터 교반부는 어떠한 구성이어도 적용가능하다.

다음에, 도 24의 암호화장치의 확대키 생성부(204)의 다른 구성례를 설명한다. 도 36에 공통키의 비트길이가 256비트인 경우에서의 도 24의 암호화장치의 확대키 생성부(204A)의 구성례를 나타낸다. 도 36은 도 3에 나타낸 라운드 트립 구성에 기초한 것으로 하고, 초단부분과 반환부분에 대해서만 나타내고 있다. 도 중, 231A는 비선형 함수(F)이고, 233은 배타적 논리합이며, 235는 단에 따라 다른 정수 const(r)과의 배타적 논리합이다. 231A, 237A, 239A, 240A, 241A, 242A, 243A의 유닛에 대해서는 후술한다. 한편, 도 중의 0.5L은 0.5단째의 왼쪽 반분을 의미하고, 0.5R은 0.5단째의 오른쪽 반분을 의미한다. 마찬가지로, 1.0L은 1.0단째의 왼쪽 반분을 의미하고, 1.0R은 1.0단째의 오른쪽 반분을 의미한다. 여기서 하이어로크립트(Hierocrypt)의 네스팅으로 되어 있는 확대 S-박스의 초단에 들어가는 키가 다음 단으로 들어가는 키인지를 구별하고 있다.

도 37에 도 36의 비선형 함수(231A)의 구성례를 나타낸다. 도 중, 2311은 배타적 논리합이고, 2313은 S-박스이며, 2315는 $P^{(16)}$ 이다.

도 38a, 도 38b는 도 36의 $P^{(32)}$ 로 나타낸 유닛(237A), $(p^{(32)})^{-1}$ 로 나타낸 유닛(243A)의 상세(詳細)이다. 어떤 32비트와 다른 32비트의 배타적 논리합을 취하는 조작이 4회 행하여진다.

도 36의 "5", "E", "B", "3"으로 나타낸 빗금이 쳐진 유닛(239A, 240A, 241A, 242A)은 입력에 갈루아체 GF(2^4)의 근본 5, E, B, 3을 곱한 것으로, 도 35에 나타낸 결선표현(결선패턴)을 갖는다.

각 단의 배타적 논리합(235)에 입력되는 단에 따라 다른 정수 const(r)는 표1에 나타낸다. 초단에 공급되는 키의 비트수가 192비트, 128비트인 경우를 표2, 표3에 나타낸다. 키의 단수는 근본의 키로부터 보아 키의 단함수의 몇 단째를 이용하고 있는지를 나타내는 것이다. 한편, 키의 비트수가 192비트, 128비트인 경우는 초단에 비트수를 256비트로 부풀리기 위해 패딩(padding)이 행하여진다. 192비트를 256비트로 부풀리는 패딩단의 구성을 도 39a에, 128비트를 256비트로 부풀리는 패딩단의 구성을 도 38b에 나타낸다. 더미단의 키 생성부를 도 40에 나타낸다. 정수 G(0) G(5)를 표4에 나타낸다.

표1

256비트(8R)

확대키	함수	정수	단(stage)
패딩을 행하지 않음			
-	KEp	G(5)	더미
K1	KEp	G(4)	1
K2	KEp	G(0)	2
K3	KEp	G(2)	3
K4	KEp	G(1)	4
K5	KEp	G(3)	5
K6	KEc	G(3)	5
K7	KEc	G(1)	4
K8	KEc	G(2)	3
K9	KEc	G(0)	2

표2

192비트(7R)

확대키	함수	정수	단(stage)
-	H_2 ‖ H_3		패딩을 행함
-	KEp	G(5)	더미
K1	KEp	G(1)	1
K2	KEP	G(0)	2
K3	KEp	G(3)	3
K4	KEp	G(2)	4
K5	KEc	G(2)	4
K6	KEc	G(3)	3
K7	KEc	G(0)	2
K8	KEc	G(1)	1

표3

128비트(6R)

확대키	함수	정수	단(stage)
-	H_3 ‖ H_2		패딩을 행함
-	KEp	G(5)	더미
K1	KEp	G(0)	1
K2	KEP	G(1)	2
K3	KEp	G(2)	3
K4	KEp	G(3)	4
K5	KEc	G(3)	4
K6	KEc	G(2)	3
K7	KEc	G(1)	2

표4

G(0)	G(1)	G(2)	G(3)	G(4)	G(5)
H_3 ‖ H_0	H_2 ‖ H_1	H_1 ‖ H_3	H_0 ‖ H_2	H_2 ‖ H_3	H_1 ‖ H_0

H_0 = 0x5A827999

H_1 = 0x6ED9EBA1

H_2 = 0x8F1BBCDC

H_3 = 0xCA62C1D6

H_4 = 0xF7DEF58A

표5

128비트(6R)

확대키	함수	정수	단(stage)
-	KEp	H_0	더미
K1	KEp	H_1	1
K2	KEP	H_2	2
K3	KEp	H_3	3
K4	KEp	H_4	4
K5	KEc	H_4	4
K6	KEc	H_3	3
K7	KEc	H_2	2

다음에, 도 24의 암호화장치의 확대키 생성부(204)의 더욱 다른 구성례를 설명한다. 도 41에 공통키의 비트길이가 256비트인 경우에서의 도 24의 암호화장치의 확대키 생성부(204B)의 구성례를 나타낸다. 도 41도 도 3에 나타낸 라운드 트립 구성에 기초한 것으로 하고, 초단부분과 반환부분에 대해서만 나타내고 있다. 도 중, 231B는 비선형 함수(F)이고, 233은 배타적 논리합이며, 235는 단에 따라 다른 정수 const(r)과의 배타적 논리합이다. 231B, 237B, 239B, 240B, 241B, 242B, 243B의 유닛에 대해서는 후술한다.

도 42에 도 41의 비선형 함수(231B)의 구성례를 나타낸다. 도 중, 2311은 배타적 논리합이고, 2313은 S-박스이며, 2315는 $P^{(8)}$ 이다.

도 43a, 도 43b는 도 41의 $P^{(16)}$ 으로 나타낸 유닛(237B), $(p^{(16)})^{-1}$ 으로 나타낸 유닛(243B)의 상세이다. 어떤 16비트와 다른 16비트의 배타적 논리합을 취하는 조작이 4회 행하여진다.

도 41의 " 5" , " B" 로 나타낸 빗금이 쳐진 유닛은 입력에 갈루아체 $GF(2^4)$의 근본 5, B를 곱한 것으로, 도 35에 나타낸 결선표현(결선패턴)을 갖는다.

각 단의 배타적 논리합(235)에 입력되는 단에 따라 다른 정수 const(r)는 표5에 나타낸다. 더미단의 키 생성부를 도 44에 나타낸다. 정수 H_0 H_4를 표4에 나타내고 있는 것이다.

이하에서는, 본 실시형태의 하드웨어 구성, 소프트웨어 구성에 대해 설명한다.

본 실시형태의 암호화장치나 복호장치는 하드웨어로서도, 소프트웨어로서도 실현가능하다.

본 실시형태는, 소프트웨어로 표현하는 경우에, 암호화장치나 복호장치를 실현하는 프로그램으로, 컴퓨터에 소정의 수단을 실행시키기 위한(혹은 컴퓨터를 소정의 수단으로서 기능시키기 위한, 혹은 컴퓨터에 소정의 기능을 실현시키기 위한) 프로그램을 기록한 컴퓨터독출가능한 기록매체로서도 실시할 수 있다.

소프트웨어로서 구성하는 경우, 반도체장치로서 형성할 수 있다.

본 발명을 적용한 암호화장치나 복호장치를 구성하는 경우, 혹은 암호화 프로그램이나 복호 프로그램을 작성하는 경우에, 블록 혹은 모듈을 모두 개별로 작성하는 것도 가능하지만, 동일 구성을 갖는 블록 또는 모듈에 대해서는 1 또는 적당 수만 준비해 두고, 그것을 알고리즘의 각 부분에서 공유하는(돌려 사용하는) 것도 가능하다.

소프트웨어의 경우에는, 멀티프로세서를 이용해 병렬처리를 행하여 처리를 고속화하는 것도 가능하다.

암호화기능을 갖고 복호기능을 갖지 않는 장치로서 구성하는 것도, 복호기능을 갖고 암호화기능을 갖지 않는 장치로서 구성하는 것도, 암호화기능과 복호기능의 양쪽을 갖는 장치로서 구성하는 것도 가능하다. 마찬가지로, 암호화기능을 갖고 복호기능을 갖지 않는 프로그램으로서 구성하는 것도, 복호기능을 갖고 암호화기능을 갖지 않는 프로그램으로서 구성하는 것도, 암호화기능과 복호기능의 양쪽을 갖는 프로그램으로서 구성하는 것도 가능하다.

다음에는 본 실시형태의 시스템으로의 응용에 대해 설명한다.

본 실시형태의 암호방식은, 기본적으로는 어떠한 시스템에도 적용가능하다.

예컨대, 도 45에 나타낸 바와 같이, 송신측 장치(301)와, 수신측 장치(303)의 사이에서, 소정의 방법 또는 수속에 의해 키를 안전하게 공유해 두고, 송신측 장치(301)는 송신데이터를 블록길이마다 본 실시형태의 암호방식으로 암호화하며, 소정의 프로토콜에 따라 통신네트워크(302)를 매개로 하여 암호문을 수신측 장치(303)로 송신하고, 암호문을 수신한 수신측 장치(303)에서는 수신한 암호문을 블록길이마다 본 실시형태의 암호방식으로 복호하여 근본의 평문을 얻을 수 있다. 각각의 장치가 암호화기능과 복호기능을 양쪽 갖고 있으면, 쌍방향으로 암호통신을 행할 수 있다.

예컨대, 도 46에 나타낸 바와 같이, 계산기(311)에서는 소정의 방법으로 키를 생성하고, 보존하고 싶은 데이터를 블록길이마다 본 실시형태의 암호방식으로 암호화하며, 소정의 네트워크(예컨대, LAN, 인터넷 등; 314)을 매개로 하여 암호화 데이터로서 데이터 서버(313)에 보존해 둔다. 계산기(311)에서는, 이 데이터를 독출하고 싶을 때는, 데이터 서버(313)로부터 소망하는 암호화 데이터를 읽어들이고, 이것을 블록길이마다 본 실시형태의 암호방식으로 복호하여 근본의 평문을 얻을 수 있다. 다른 계산기(312)가 이 키를 알고 있으면, 마찬가지로 복호하여 근본의 평문을 얻을 수 있지만, 키를 알지 못한 다른 계산기는 그 암호데이터를 복호할 수 없어 정보의 보안제어가 가능하게 된다.

예컨대, 도 47에 나타낸 바와 같이, 콘텐츠 제공 측에서는 암호화장치(321)에 의해 어떤 콘텐츠를 어떤 키로 블록길이마다 본 실시형태의 암호방식으로 암호화하고, 이것을 암호화 콘텐츠로서 기록매체(322)에 기록하며, 이것을 반포 등 한다. 기록매체(322)를 취득한 사용자 측에서는, 소정의 방법으로 그 어떤 키를 입수함으로써, 복호장치(323)에 의해 그 콘텐츠를 블록길이마다 본 실시형태의 암호방식으로 복호하여 콘텐츠의 열람 또는 재생 등을 행할 수 있다.

물론, 상기 이외에도 여러 가지의 시스템에 적용가능하다.

본 실시형태는, 암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 암호화장치 또는 복호장치로, 복수단의 라운드함수에 대해 초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성하는 복수단의 라운드 처리부와, 상기 라운드 처리부의 전부 또는 일부의 단에

서 생성된 상기 서브키의 각각에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 상기 확대키로서 출력하기 위한 출력부를 구비하고, 상기 라운드 처리부는 복수의 라운드함수를 종속접속한 라운드함수 계열에 있어서 상기 공통키를 그 초단으로 입력한 경우에 그 최종단이 그 공통키와 동일한 값을 생성하도록 설정된 라운드함수 계열에서의 모든 단 또는 그 중의 일부에서 초단으로부터 연속한 복수단에 대한 라운드함수를 그 라운드함수 계열의 단의 순번에 따라 이용하는 것이다.

상기 라운드함수 계열은, 초단으로부터의 단수와 최종단으로부터의 단수가 일치하는 2개의 라운드함수를 서로 역함수로 되도록 설정한 것이도록 해도 좋다.

상기 라운드함수 계열은, 적어도 제1의 특정 단과 제2의 특정 단의 사이의 연속하는 복수단에 대해 그 제1의 특정 단으로부터의 단수와 그 제2의 특정 단으로부터의 단수가 일치하는 2개의 라운드함수를 서로 역함수로 되도록 설정한 부분계열을 포함하는 것이도록 해도 좋다.

상기 라운드함수 계열은, 적어도 제1의 특정 단으로부터 단수 증가방향으로 특정 단수 떨어진 단까지의 연속하는 범위와 제2의 특정 단으로부터 단수 감소방향으로 특정 단수 떨어진 단까지의 연속하는 범위에 대해, 그 제1의 특정 단으로부터의 단수와 그 제2의 특정 단으로부터의 단수가 일치하는 2개의 라운드함수를 서로 역함수로 되도록 설정한 부분계열을 포함하는 것이도록 해도 좋다.

상기 출력부는, 상기 확대키의 출력을 위해 상기 서브키를 이용할 때에, 그 서브키에 대해서는 그 모든 비트 중에서 선택한 당해 서브키를 일의적으로 결정하기에는 충분하지 않은 부분만을 이용하도록 해도 좋다.

상기 라운드함수 계열에 속하는 라운드함수 중, 적어도 그 초단 또는 초단으로부터 단수 증가방향으로 소정 단수 떨어진 단까지의 연속하는 범위 및/또는 최종단 또는 최종단으로부터 단수 감소방향으로 소정 단수 떨어진 단까지의 연속하는 범위에 속하는 라운드함수에 대응하는 서브키는, 상기 확대키로서 또는 그 근본으로 되는 데이터로서 이용하지 않도록 해도 좋다.

상기 라운드함수 계열에 속하는 라운드함수 중, 적어도 그 초단으로부터의 단수와 최종단으로부터의 단수가 일치하는 2개의 라운드함수에 대응하는 서브키의 어느 한쪽 또는 양쪽은, 상기 확대키로서 또는 그 근본으로 되는 데이터로서 이용하지 않도록 해도 좋다.

상기 라운드함수 계열에 속하는 라운드함수 중, 적어도 그 초단 또는 초단으로부터 단수 증가방향으로 소정 단수 떨어진 단까지의 연속하는 범위 및/또는 최종단 또는 최종단으로부터 단수 감소방향으로 소정 단수 떨어진 단까지의 연속하는 범위에 속하는 라운드함수에 대응하는 서브키 중, 초단으로부터의 단수와 최종단으로부터의 단수가 일치하는 2개의 라운드함수의 어느 한쪽 또는 양쪽에 대응하는 서브키는, 상기 확대키로서 또는 그 근본으로 되는 데이터로서 이용하지 않도록 해도 좋다.

복수의 상기 확대키 중의 임의의 것이 항시는 일치하지 않도록 해도 좋다.

복수의 상기 확대키 중의 임의의 것이 그들 확대키의 모든 비트 중의 임의의 비트군에 대해서도 항시는 일치하지 않도록 해도 좋다.

상기 라운드 처리부 및 상기 출력부는, 상기 데이터 교반처리에 필요한 확대키 수를 넘는 수의 확대키를 그 데이터 교반처리에 제공가능하고, 상기 제공가능한 확대키 중 실제로 상기 데이터 교반처리에 제공해야 할 확대키를 나타내는 정보, 또는 상기 데이터 교반처리에 제공해야 할 확대키 및 그 제공하는 순번을 나타내는 정보를 확장공통키로 하며, 상기 출력부는 상기 확장공통키에 따라 상기 확대키를 출력하도록 해도 좋다.

본 실시형태는, 암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 역의 순번으로 복수의 확대키를 사용하는 공통키 암호방식에 의한 암호화장치 또는 복호장치에 이용되는 확대키 생성장치로, 복수단의 라운드함수에 대해 초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성하는 라운드 처리부와, 상기 라운드 처리부의 전부 또는 일부의 단에서 생성된 상기 서브키의 각각에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 상기 확대키로서 출력하기 위한 출력부를 구비하고, 상기 라운드 처리부는 복수의 라운드함수를 종속접속한 라운드함수 계열에 있어서 상기 공통키를 그 초단으로 입력한 경우에 그 최종단이 그 공통키와 동일한 값을 생성하도록 설정된 라운드함수 계열에서의 모든 단 또는 그 중의 일부에서 초단으로부터 연속한 복수단에 대한 라운드함수를 그 라운드함수 계열의 단의 순번에 따라 이용하는 것이다.

본 실시형태에 의하면, 확대키 생성을 위한 라운드함수의 계열을, 공통키를 입력하고 공통키와 동일한 값을 출력하도록 설정함으로써, 암호화시와 복호시의 양쪽에 있어서, 종래와 같은 불필요한 지연시간이나 기억용량의 소비 없이 공통키로부터 쉴 사이 없이 확대키를 생성하는 것이 가능하게 된다.

본 실시형태에서 나타낸 각각의 구성은 일례일 뿐이고, 그 이외의 구성을 배제하는 취지의 것이 아니라 예시한 구성의 일부를 다른 것으로 치환하거나, 예시한 구성의 일부를 생략하거나, 예시한 구성에 다른 기능을 부가하거나, 그들을 조합시키거나 하는 것 등에 의해 얻어지는 다른 구성도 가능하다. 예시한 구성과 논리적으로 등가인 다른 구성, 예시한 구성과 논리적으로 등가인 다른 부분을 포함한 다른 구성, 예시한 구성의 요부와 논리적으로 등가인 다른 구성 등도 가능하다. 예시한 구성과 동일 또는 유사한 목적을 달성하는 다른 구성, 예시한 구성과 동일 또는 유사한 효과를 발휘하는 다른 구성 등도 가능하다.

본 발명은 상술한 실시형태에 한정되는 것이 아니라, 그 기술적 범위에 있어서 여러 가지로 변형하여 실시할 수 있다. 예컨대, 각종 구성부분에 대한 각종 변형은 적절히 조합시켜 실시하는 것이 가능하다. 본 발명은, 암호화장치로서의 발명, 복호장치로서의 발명, 시스템 전체로서의 발명, 개별 장치 내부의 구성부분에 대한 발명, 또는 그들에 대응하는 방법의 발명, 컴퓨터에 소정의 기능을 실행시키기 위한, 컴퓨터를 소정의 수단으로서 기능시키기 위한, 혹은 컴퓨터에 소정의 기능을 실현시키기 위한 프로그램을 기록한 컴퓨터독출가능한 기록매체 등, 여러 가지의 관점, 단계, 개념 또는 카테고리에 관계된 발명을 포함·내재하는 것이다.

발명의 효과

이상 설명한 바와 같이 본 발명에 의하면, 확대키 생성을 위한 지연시간의 발생을 회피하거나 또는 작게 하고, 또한 쉴 사이 없는 키 생성을 가능하게 한 암호화장치, 복호장치 및 확대키 생성장치, 확대키 생성방법 및 기록매체를 제공할 수 있다.

(57) 청구의 범위

청구항 1.

암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 암호화장치로,

초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성하는 복수단의 라운드 처리부(31_1 31_{n+1})와,

상기 복수단의 라운드 처리부의 적어도 일부에서 생성된 상기 서브키에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력하는 복수의 확대키 처리부(33_1 33_n)를 구비하고,

상기 최종단의 출력이 초단에 입력된 공통키와 같은 것을 특징으로 하는 암호화장치.

청구항 2.

제1항에 있어서, i단째의 라운드 처리부의 라운드함수는 (j-i+1)단째(여기서, i = 1 j, j는 총단수의 1/2)의 라운드 처리부의 라운드함수의 역함수인 것을 특징으로 하는 암호화장치.

청구항 3.

제1항에 있어서, L+i단째의 라운드 처리부의 라운드함수는 (H-i)단째(여기서, i = 0 j, j는 (H-L)/2의 양의 정수)의 라운드 처리부의 라운드함수의 역함수인 것을 특징으로 하는 암호화장치.

청구항 4.

제1항에 있어서, L+i단째의 라운드 처리부의 라운드함수는 (H-i)단째(여기서, i = 0 j, j는 (H-L)/2 이하의 양의 정수)의 라운드 처리부의 라운드함수의 역함수인 것을 특징으로 하는 암호화장치.

청구항 5.

제1항에 있어서, 상기 라운드함수 계열을, L+i단째의 라운드 처리부의 라운드함수는 (H-i)단째(여기서, i = 0 j, j는 (H-L)/2의 양의 정수 또는 (H-L)/2 이하의 양의 정수)의 라운드 처리부의 라운드함수의 역함수의 부분계열을 포함하지 않도록 설정한 것을 특징으로 하는 암호화장치.

청구항 6.

제1항에 있어서, 상기 확대키 처리부는 상기 복수단의 라운드 처리부가 출력하는 복수의 서브키로부터 임의로 선택된 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력하는 것을 특징으로 하는 암호화장치.

청구항 7.

제6항에 있어서, 상기 확대키 처리부는, 초단을 포함한 소정 단수의 라운드 처리부와, 최종단을 포함한 소정 단수의 라운드 처리부 이외의 라운드 처리부가 출력하는 복수의 서브키를 선택하는 것을 특징으로 하는 암호화장치.

청구항 8.

제6항에 있어서, 상기 확대키 처리부는, 초단으로부터 센 단수와 최종단으로부터 센 단수가 같은 2개의 라운드 처리부의 한쪽으로부터 출력하는 서브키는 선택하지 않는 것을 특징으로 하는 암호화장치.

청구항 9.

제1항에 있어서, 상기 확대키 처리부는 상기 복수단의 라운드 처리부가 출력하는 복수의 서브키를 순번을 바꾸어 그 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력함으로써, 서브키의 발생순번과 확대키의 발생순번이 달라지는 것을 특징으로 하는 암호화장치.

청구항 10.

제1항에 있어서, 상기 라운드 처리부 및 상기 확대키 처리부는, 상기 데이터 교반처리에 필요한 확대키 수를 넘는 수의 확대키를 발생가능하고,

상기 라운드 처리부 및 상기 확대키 처리부는, 상기 발생가능한 확대키 중 실제로 상기 데이터 교반처리에 제공해야 할 확대키를 나타내는 정보를 확장공통키로 하여 확대키를 출력하는 것을 특징으로 하는 암호화장치.

청구항 11.

암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 복호장치로,

초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성하는 복수단의 라운드 처리부(42_1 42_{n+1})와,

상기 복수단의 라운드 처리부의 적어도 일부에서 생성된 상기 서브키에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력하는 복수의 확대키 처리부(44_1 44_n)를 구비하고,

상기 최종단의 출력이 초단에 입력된 공통키와 같은 것을 특징으로 하는 복호장치.

청구항 12.

암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 암호화장치 또는 복호장치에 이용되는 확대키 생성장치로,

초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성하는 복수단의 라운드 처리부(31_1 31_{n+1}, 42_1 42_{n+1})와,

상기 복수단의 라운드 처리부의 적어도 일부에서 생성된 상기 서브키에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력하는 복수의 확대키 처리부(33_1 33_n, 44_1 44_n)를 구비하고,

상기 최종단의 출력이 초단에 입력된 공통키와 같은 것을 특징으로 하는 확대키 생성장치.

청구항 13.

암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 암호화장치에 이용되는 확대키 생성방법으로,

복수단의 라운드 처리부의 초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성하며,

상기 복수단의 라운드 처리부의 적어도 일부에서 생성된 상기 서브키에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력하고,

상기 최종단의 출력이 초단에 입력된 공통키와 같은 것을 특징으로 하는 확대키 생성방법.

청구항 14.

암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 복호장치에 이용되는 확대키 생성방법으로,

복수단의 라운드 처리부의 초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성하며,

상기 복수단의 라운드 처리부의 적어도 일부에서 생성된 상기 서브키에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력하고,

상기 최종단의 출력이 초단에 입력된 공통키와 같은 것을 특징으로 하는 확대키 생성방법.

청구항 15.

암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 암호화장치에 이용되는 확대키 생성프로그램을 격납한 컴퓨터독출가능한 기록매체로,

복수단의 라운드 처리부의 초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성시키는 프로그램 코드수단과,

상기 복수단의 라운드 처리부의 적어도 일부에서 생성된 상기 서브키에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력시키는 프로그램 코드수단을 구비하고,

상기 최종단의 출력이 초단에 입력된 공통키와 같은 것을 특징으로 하는 컴퓨터독출가능한 기록매체.

청구항 16.

암호화시의 데이터 교반처리와 복호시의 데이터 교반처리에서 복수의 확대키를 역의 순번으로 사용하는 공통키 암호방식에 의한 복호장치에 이용되는 확대키 생성프로그램을 격납한 컴퓨터독출가능한 기록매체로,

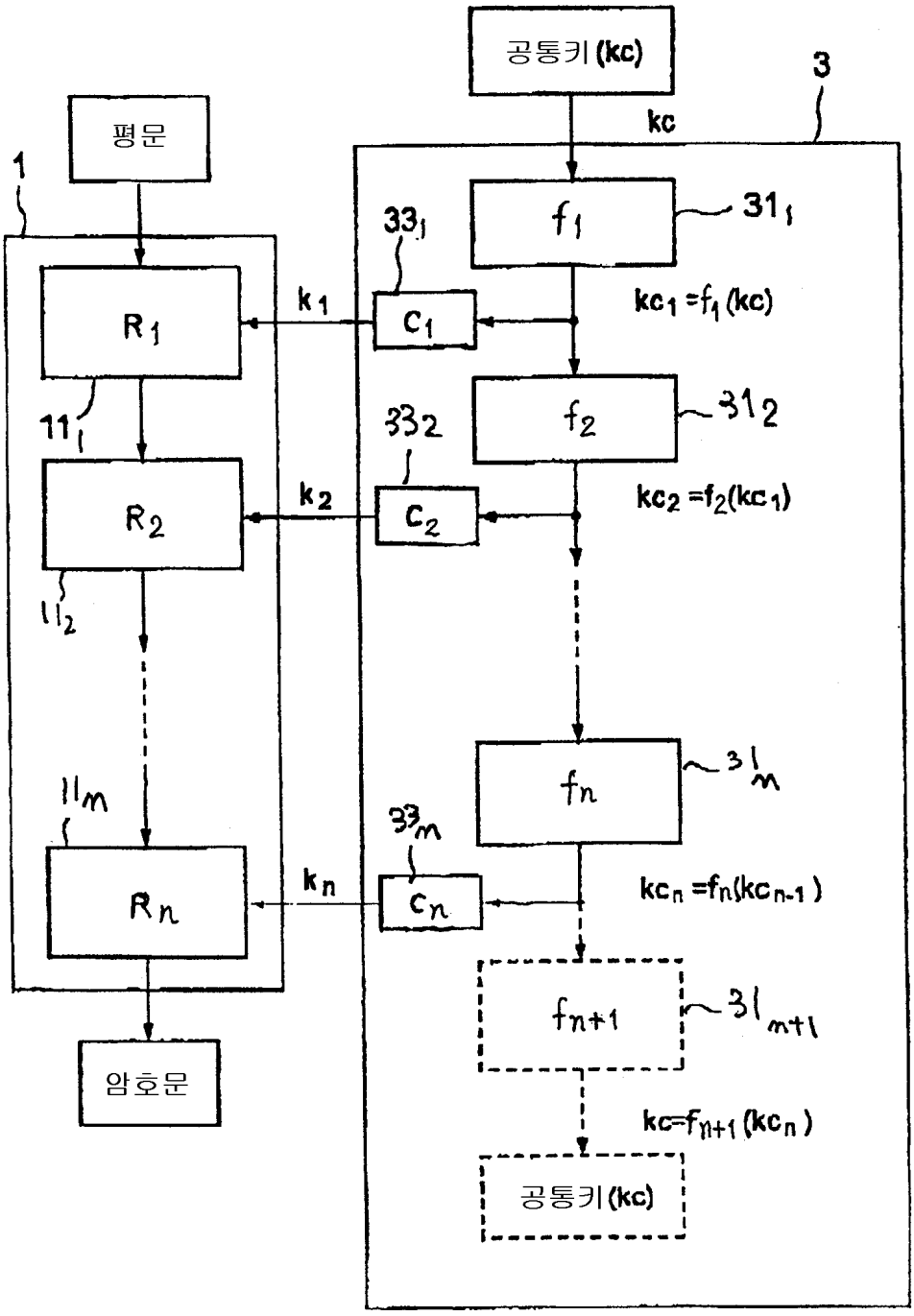
복수단의 라운드 처리부의 초단에서는 공통키를 입력으로 하여 소정의 라운드함수를 이용해 서브키를 생성하고, 2단째 이후에서는 전단에서 생성된 서브키를 입력으로 하여 소정의 라운드함수를 이용해 새로운 서브키를 생성시키는 프로그램 코드수단과,

상기 복수단의 라운드 처리부의 적어도 일부에서 생성된 상기 서브키에 대해 그 서브키의 모든 비트 또는 그 일부를 그대로 또는 이것에 소정의 변환처리를 실시한 후에 확대키로서 출력시키는 프로그램 코드수단을 구비하고,

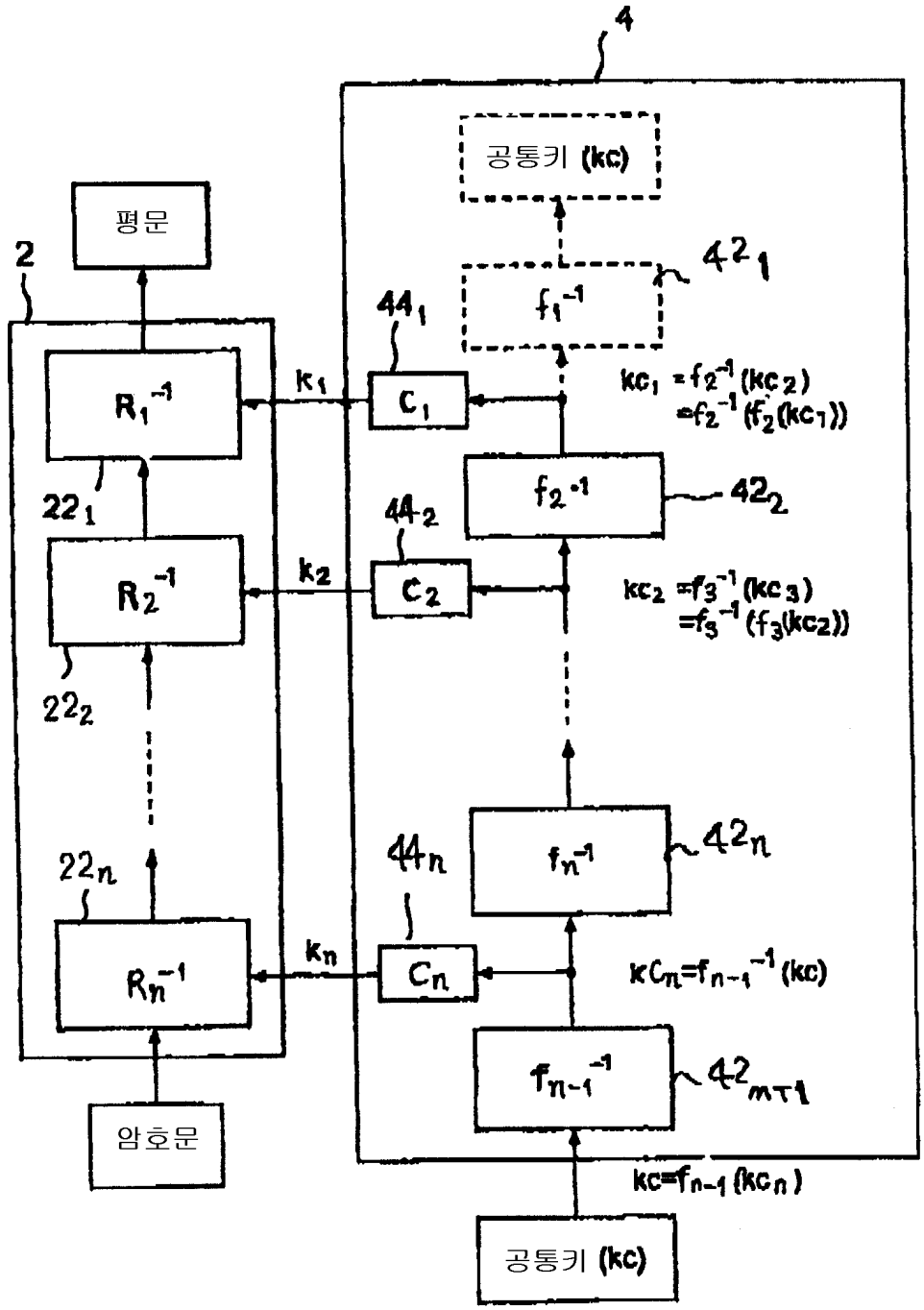
상기 최종단의 출력이 초단에 입력된 공통키와 같은 것을 특징으로 하는 컴퓨터독출가능한 기록매체.

도면

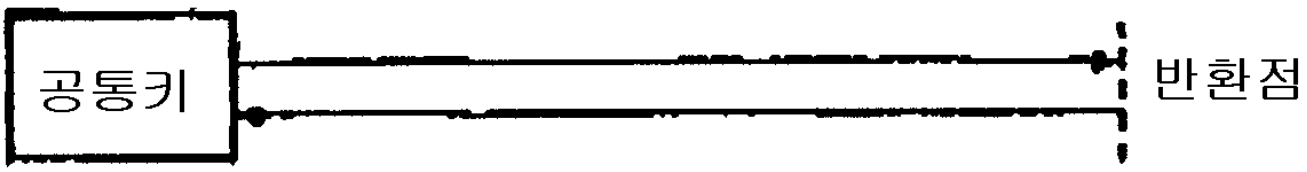
도면 1


공통키(kc)
3
kc
평문
1
f_1
31_1
33_1
R_1
k_1
C_1
$kc_1 = f_1(kc)$
11_1
f_2
31_2
33_2
R_2
k_2
C_2
$kc_2 = f_2(kc_1)$
11_2
f_n
31_m
11_m
33_m
R_n
k_n
C_n
$kc_n = f_n(kc_{n-1})$
f_{n+1}
31_{m+1}
$kc = f_{n+1}(kc_n)$
공통키(kc)
암호문

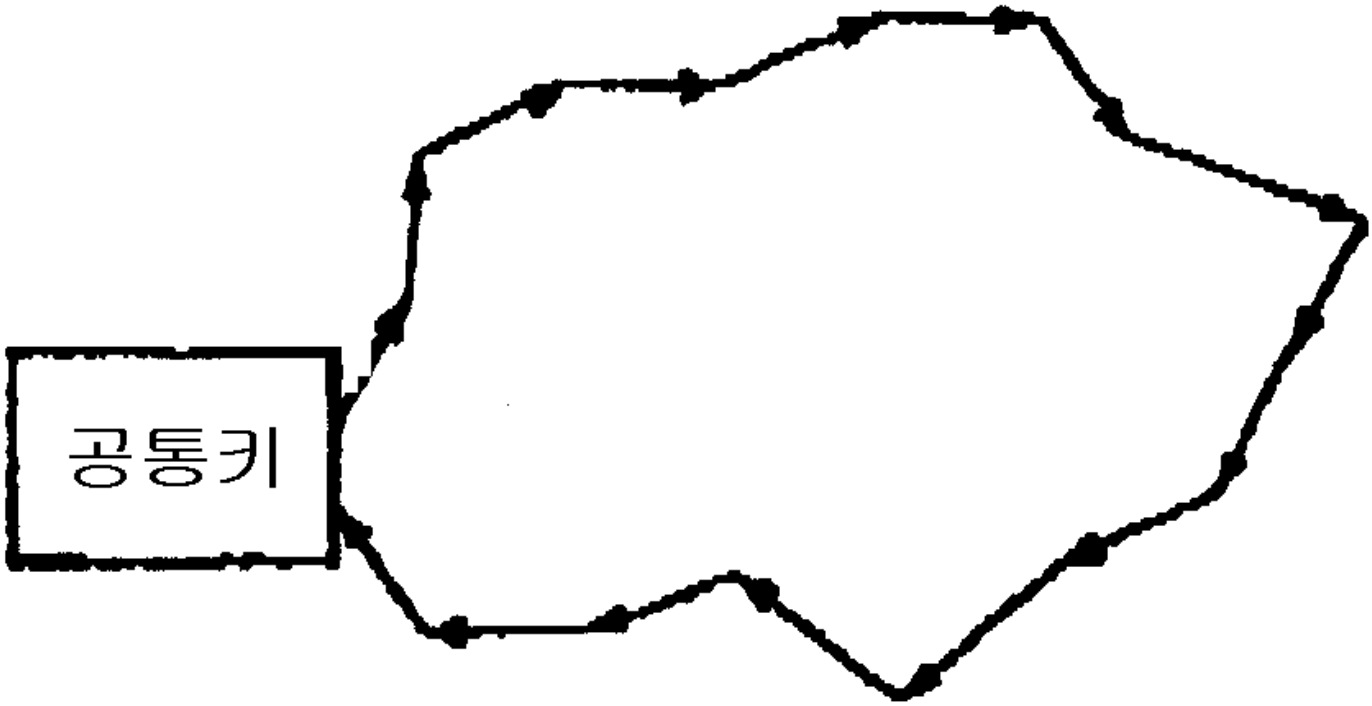
도면 2


4
공통키 (kc)
평문
2
42_1
44_1
f_1^{-1}
R_1^{-1}
k_1
C_1
$kc_1 = f_2^{-1}(kc_2)$
$= f_2^{-1}(f_2(kc_1))$
22_1
f_2^{-1}
42_2
44_2
R_2^{-1}
k_2
C_2
$kc_2 = f_3^{-1}(kc_3)$
$= f_3^{-1}(f_3(kc_2))$
22_2
22_n
44_n
f_n^{-1}
42_n
R_n^{-1}
k_n
C_n
$kc_n = f_{n-1}^{-1}(kc)$
f_{n-1}^{-1}
42_{m-1}
암호문
$kc = f_{n-1}(kc_n)$
공통키 (kc)

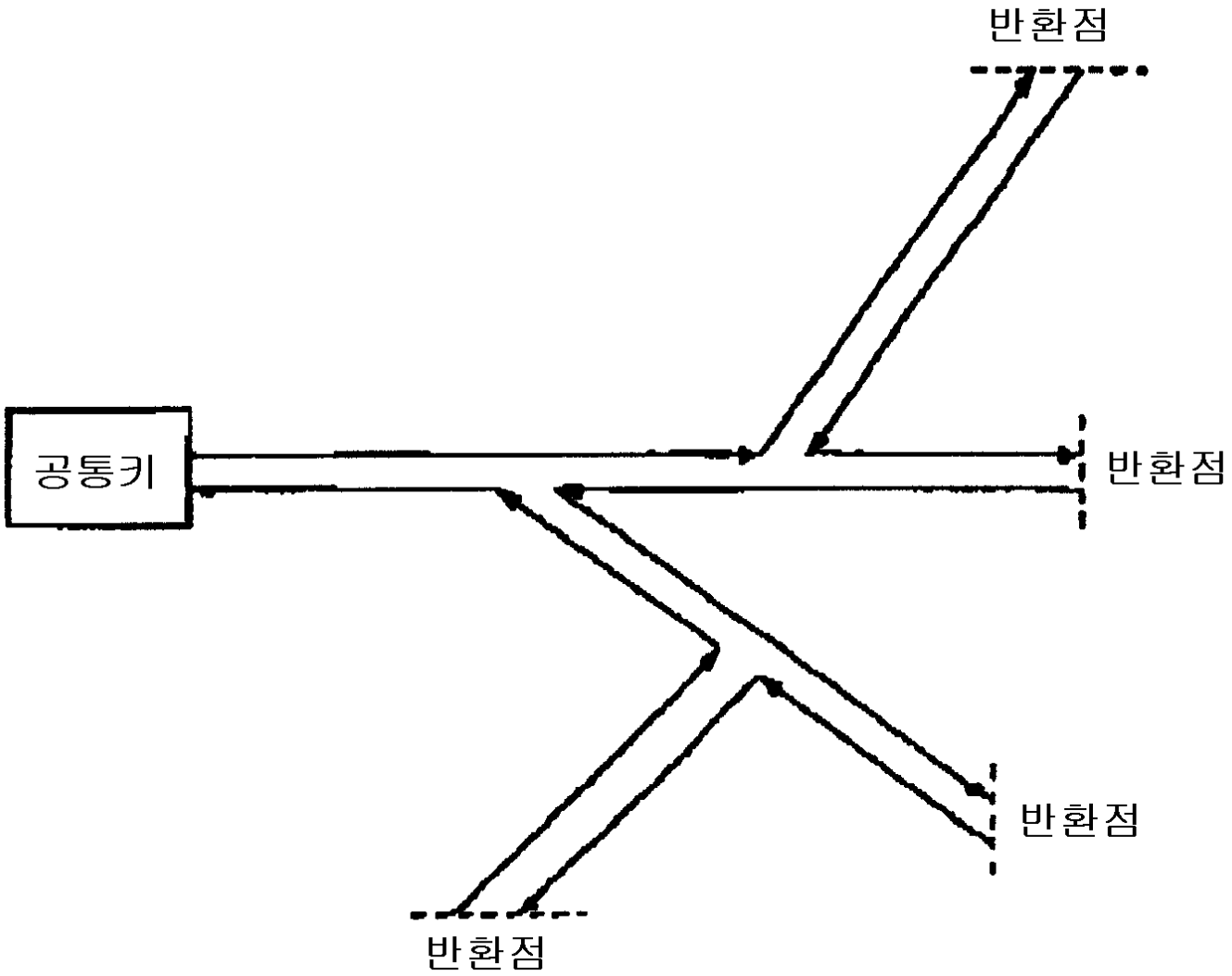
도면 3


공통키
반환점

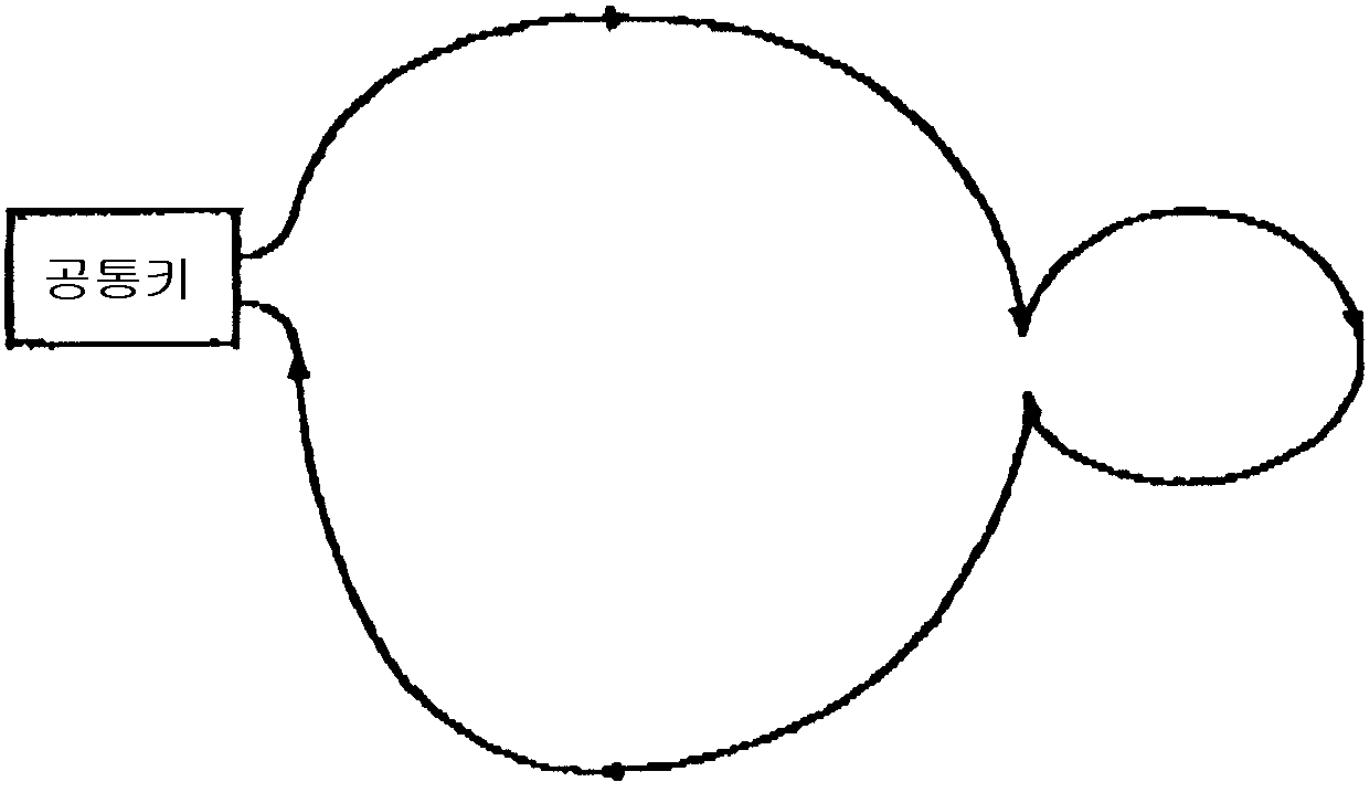
도면 4



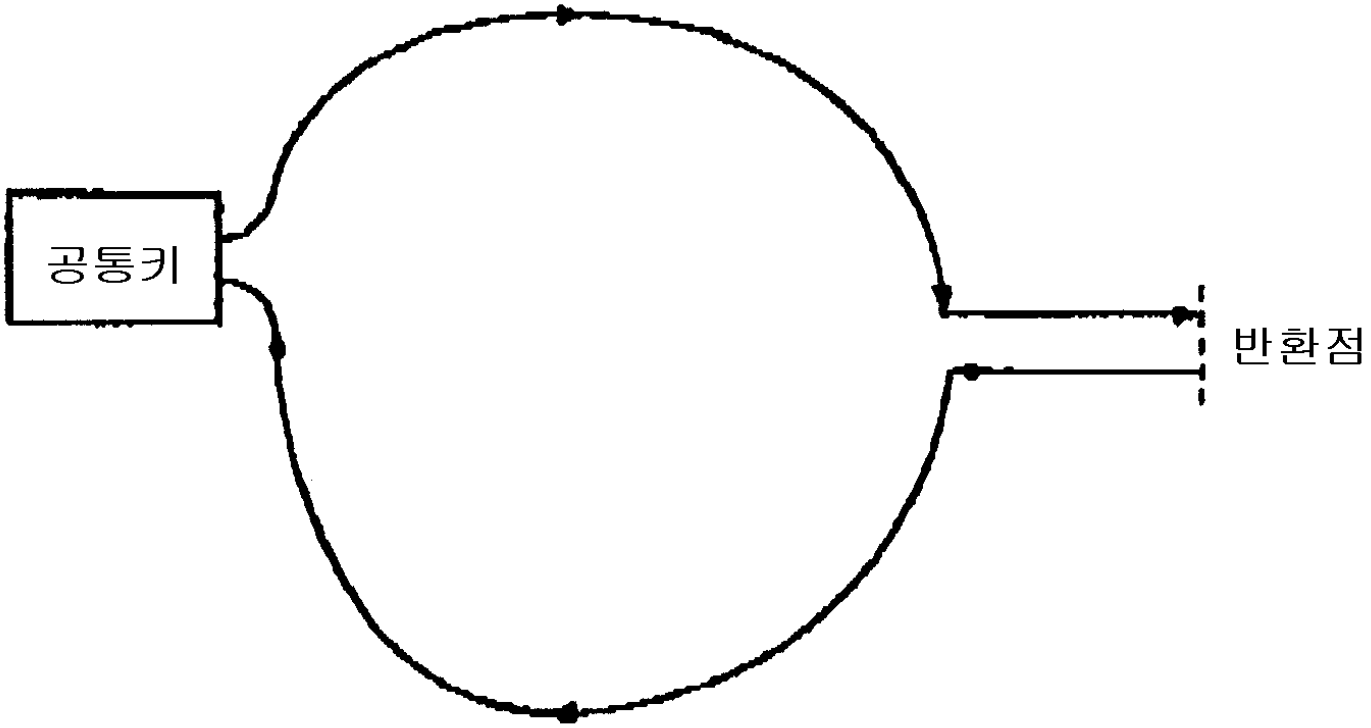
도면 5



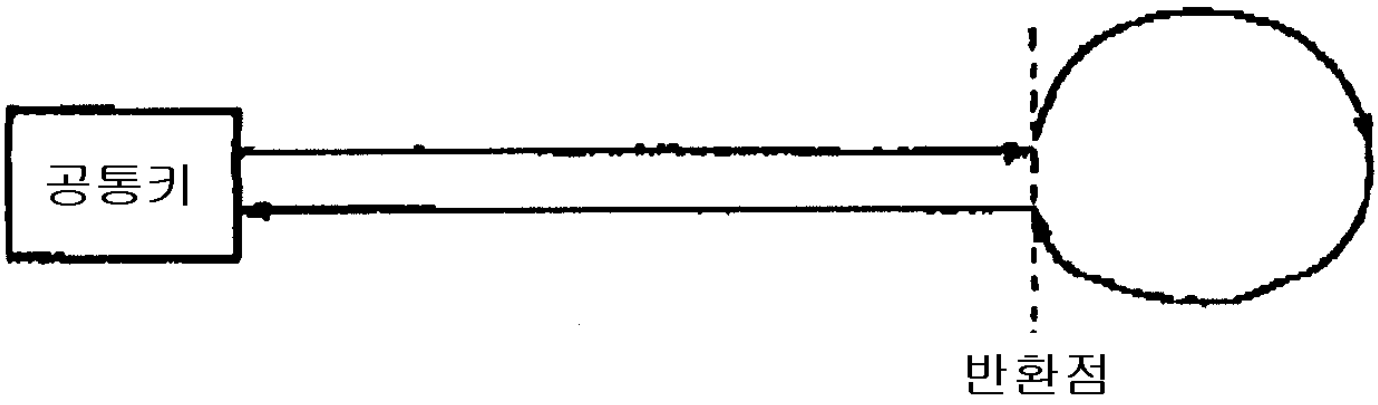
도면 6



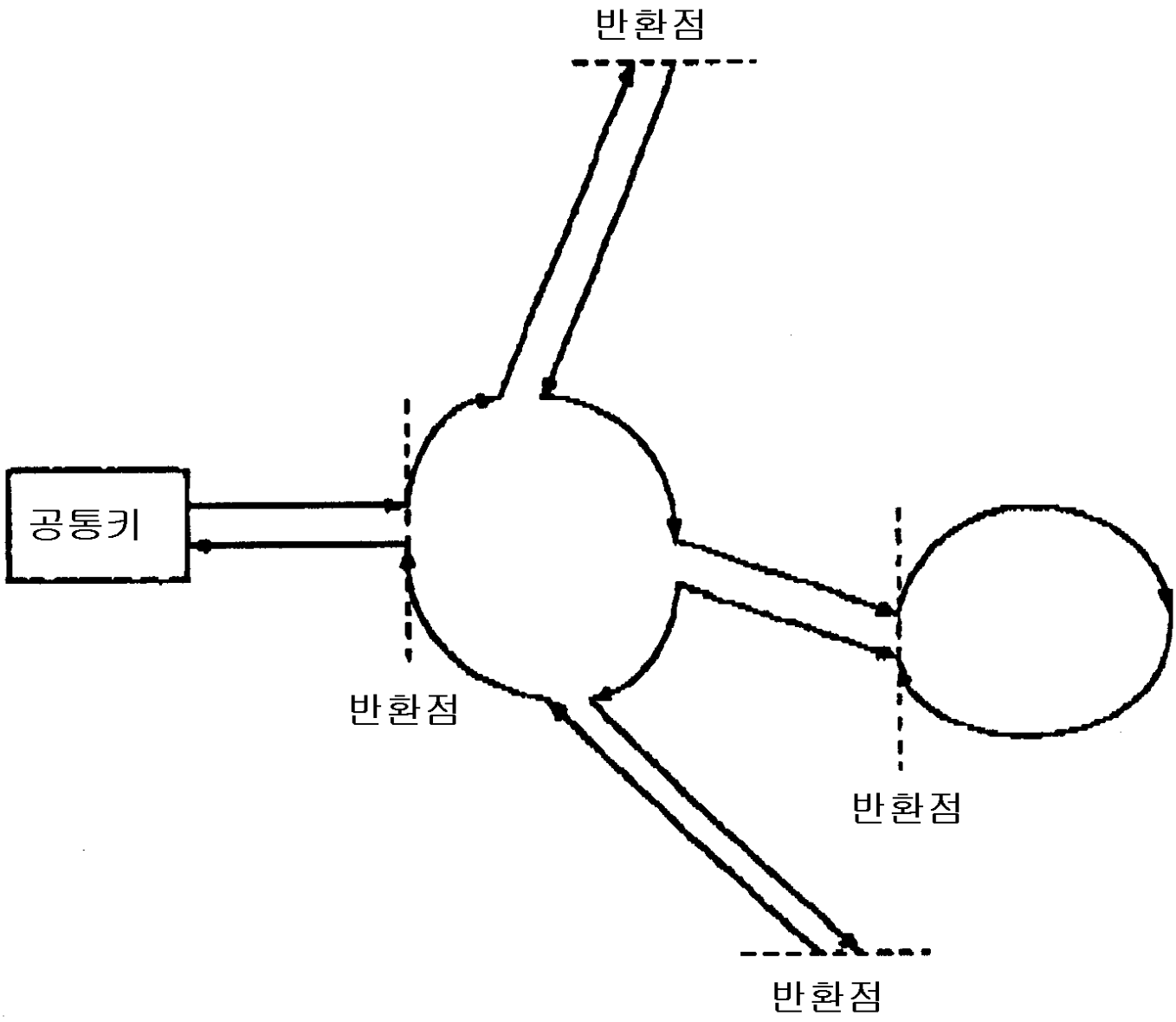
도면 7



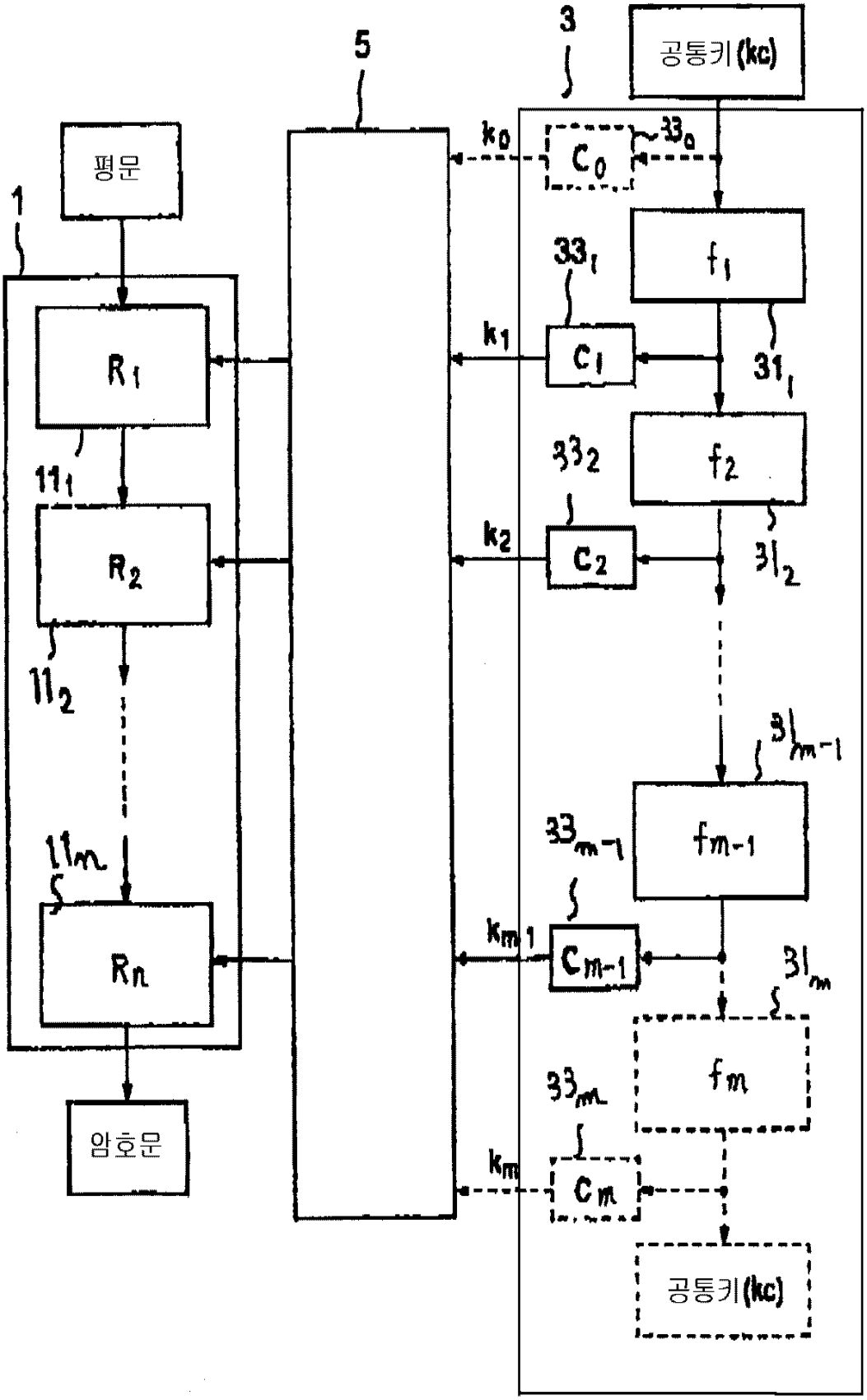
도면 8



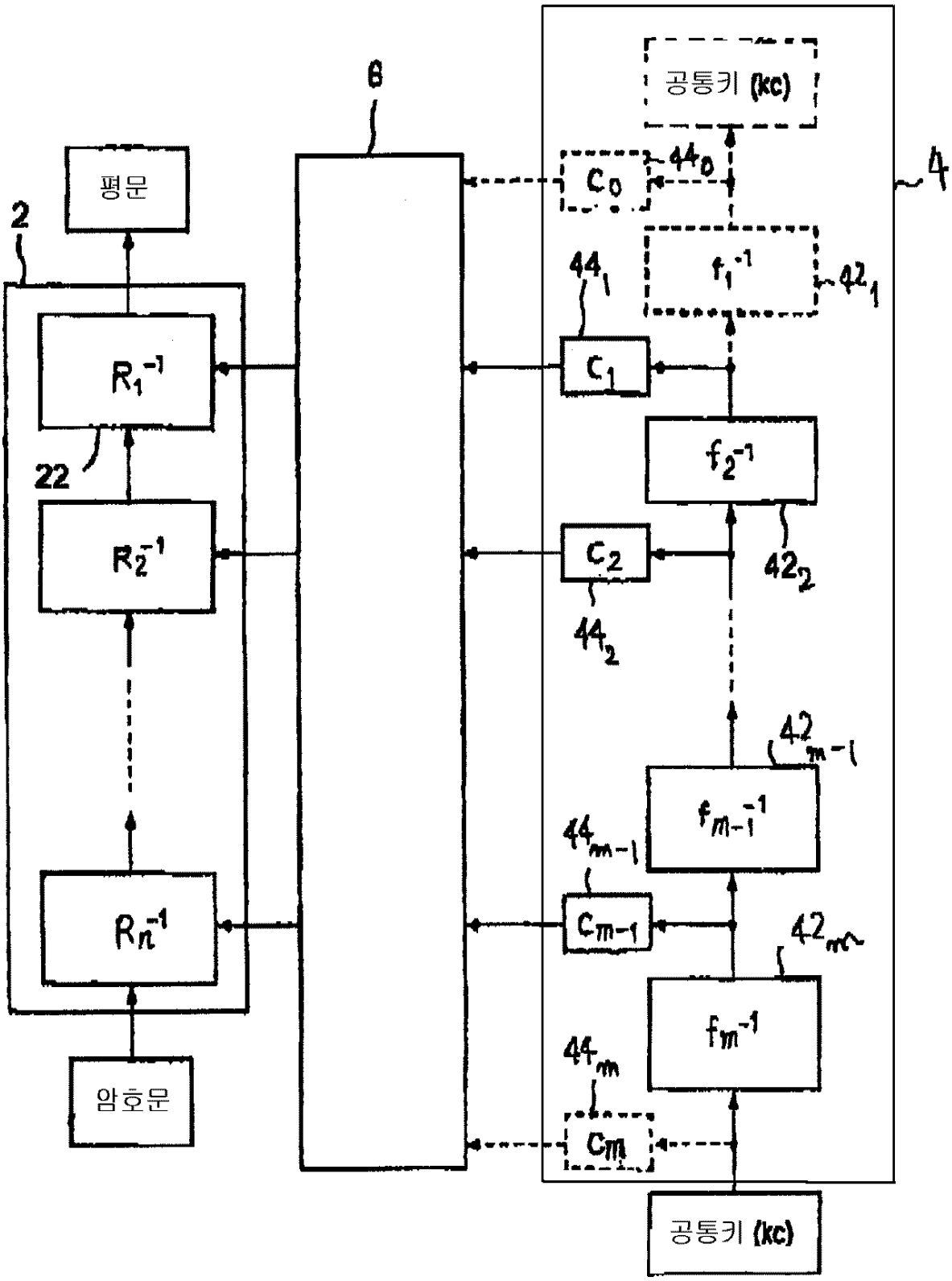
도면 9



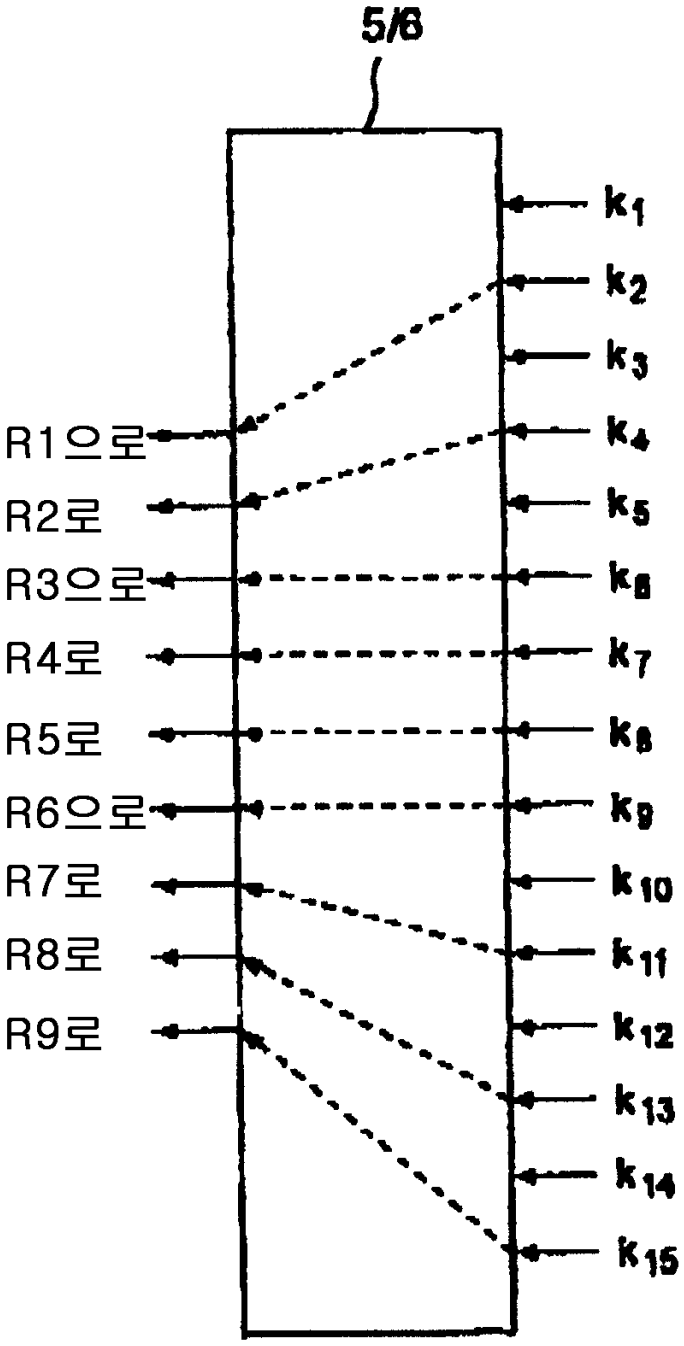
도면 10


3
공통키(kc)
5
평문
1
k_0
C_0
33_0
33_1
f_1
k_1
C_1
31_1
R_1
11_1
33_2
f_2
k_2
C_2
31_2
R_2
11_2
31_{m-1}
33_{m-1}
f_{m-1}
11_m
k_{m1}
C_{m-1}
31_m
R_n
f_m
33_m
k_m
C_m
암호문
공통키(kc)

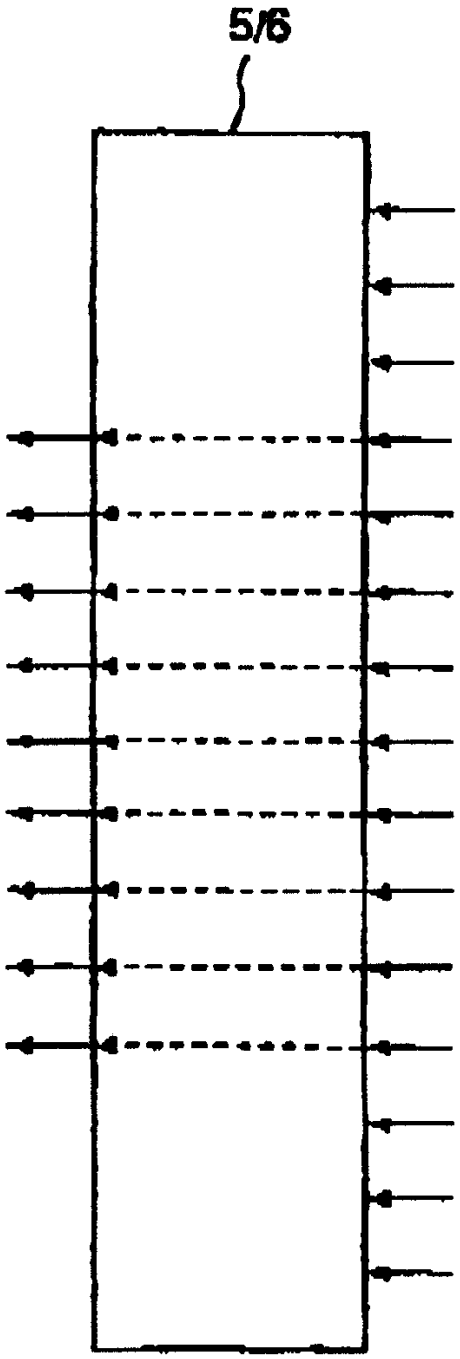
도면 11


6
공통키 (kc)
평문
2
4
C_0
44_0
44_1
f_1^{-1}
42_1
R_1^{-1}
C_1
22
f_2^{-1}
R_2^{-1}
C_2
42_2
44_2
42_{m-1}
f_{m-1}^{-1}
44_{m-1}
R_n^{-1}
C_{m-1}
42_m
f_m^{-1}
44_m
암호문
C_m
공통키 (kc)

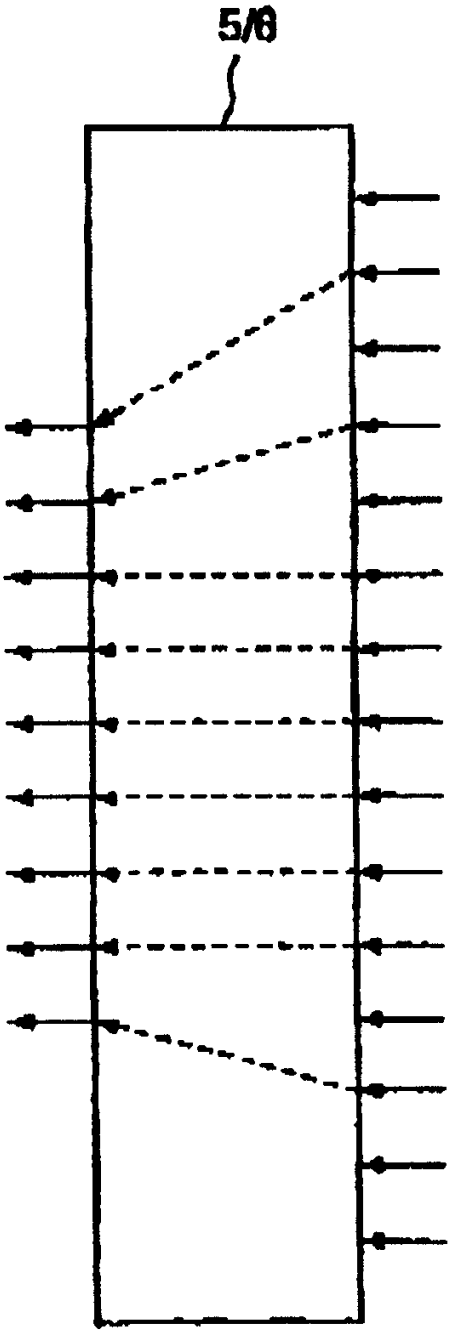
도면 12


5/6
R1으로
R2로
R3으로
R4로
R5로
R6으로
R7로
R8로
R9로
k1
k2
k3
k4
k5
k6
k7
k8
k9
k10
k11
k12
k13
k14
k15

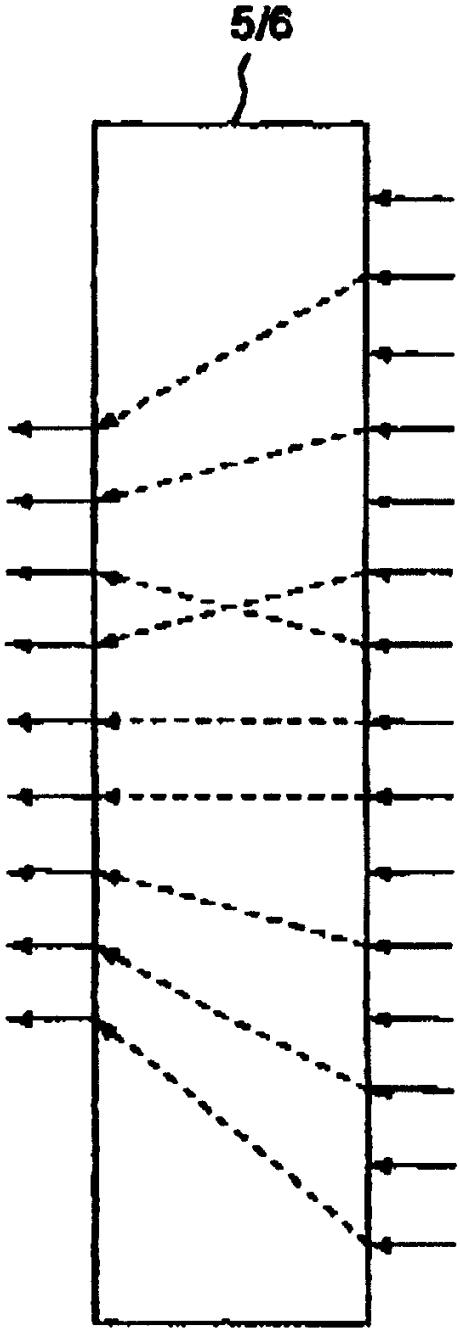
도면 13



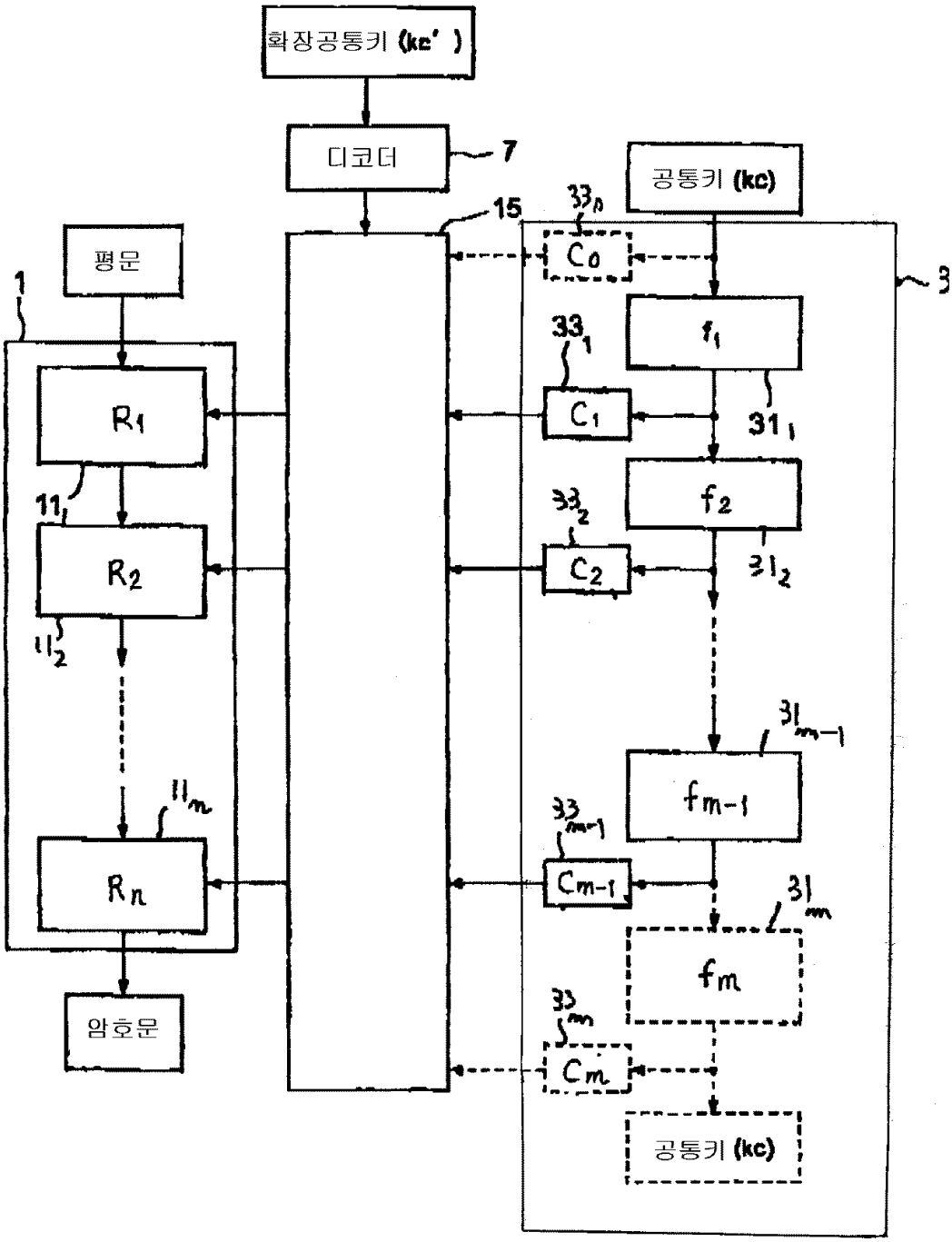
도면 14



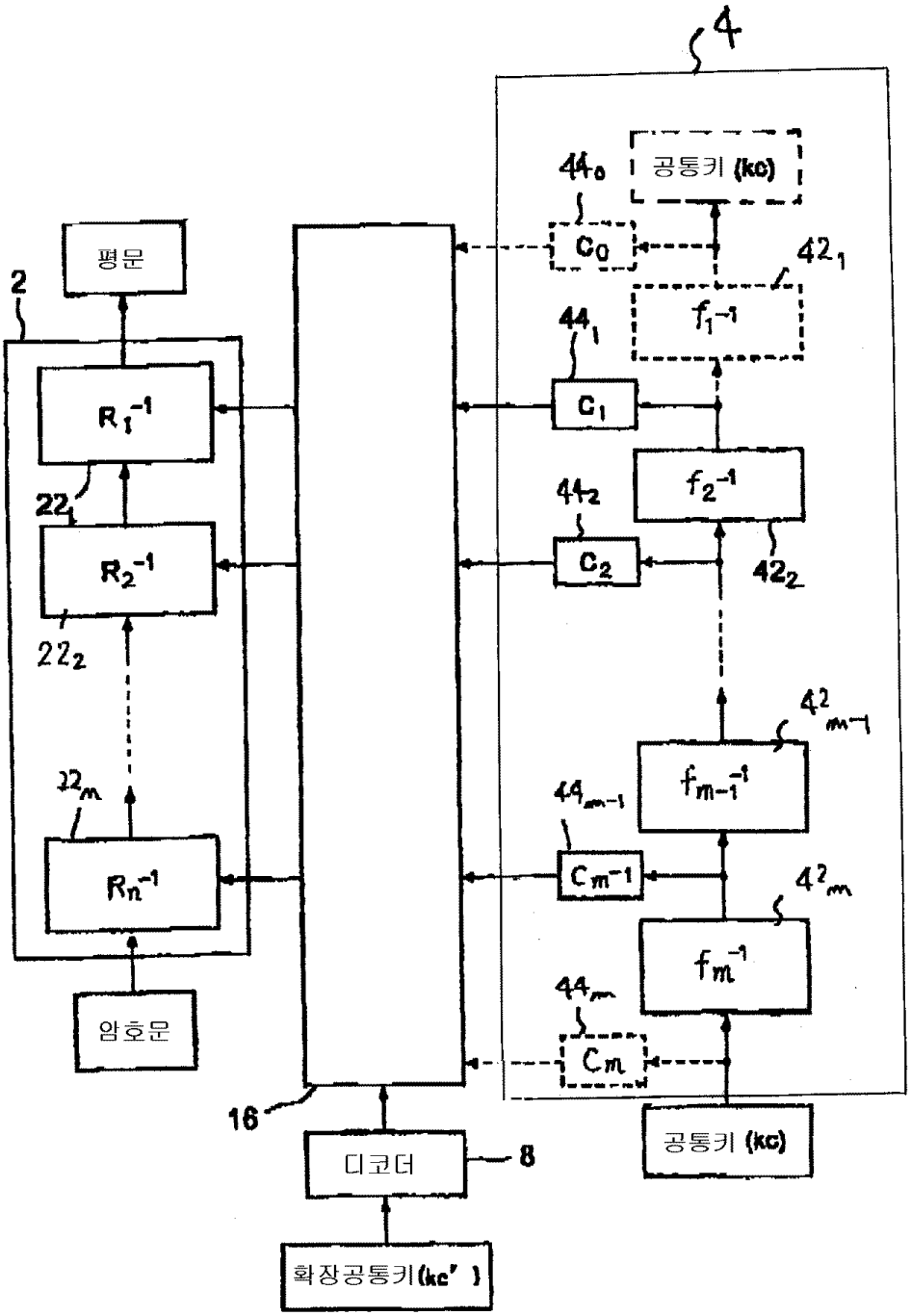
도면 15



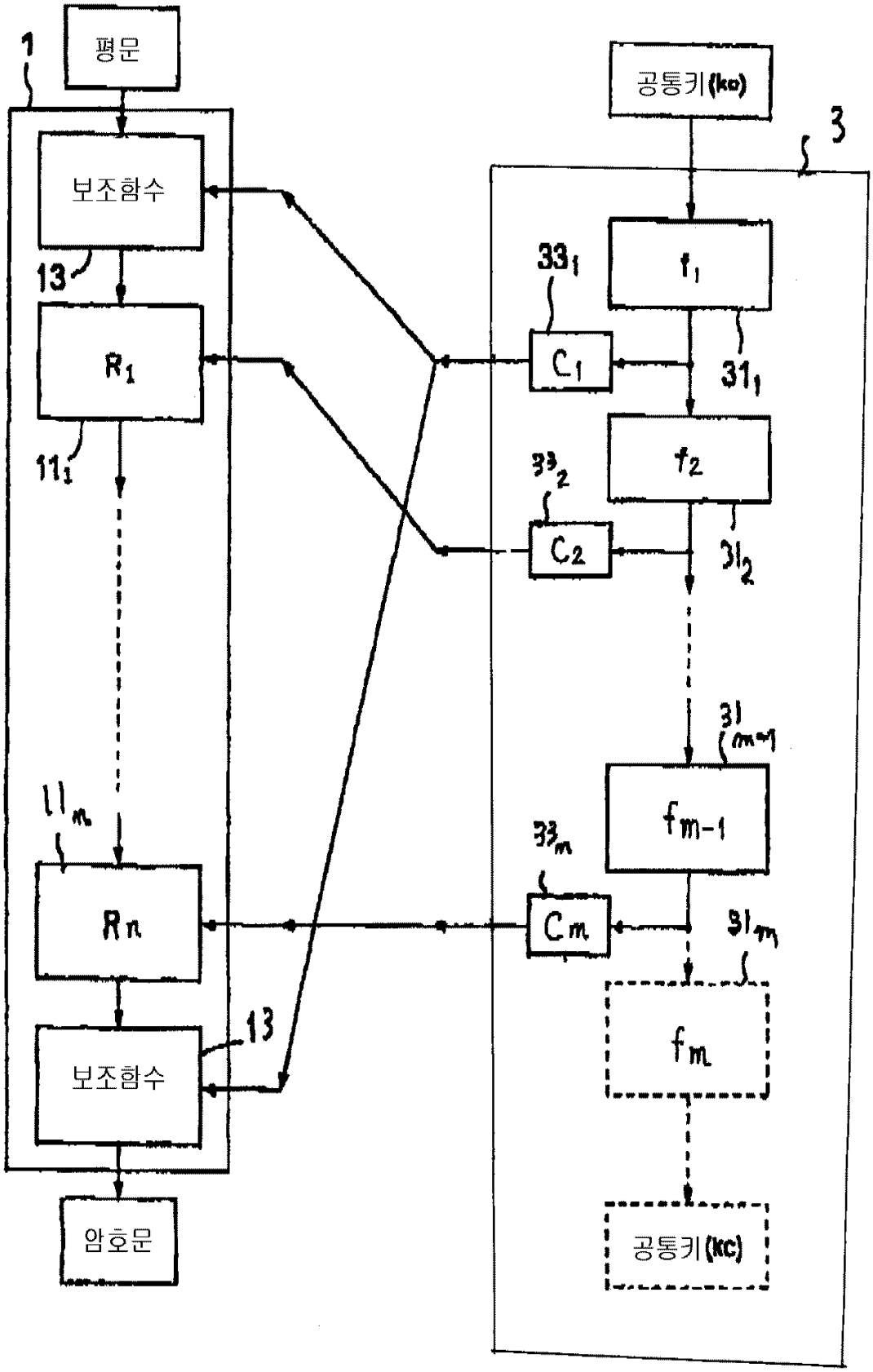
도면 16


확장공통키 (kc')
디코더
7
15
공통키 (kc)
평문
1
3
33_0
C_0
f_1
33_1
C_1
31_1
R_1
11_1
f_2
33_2
C_2
31_2
R_2
11_2
31_{m-1}
f_{m-1}
33_{m-1}
C_{m-1}
31_m
11_n
R_n
f_m
33_m
C_m
암호문
공통키 (kc)

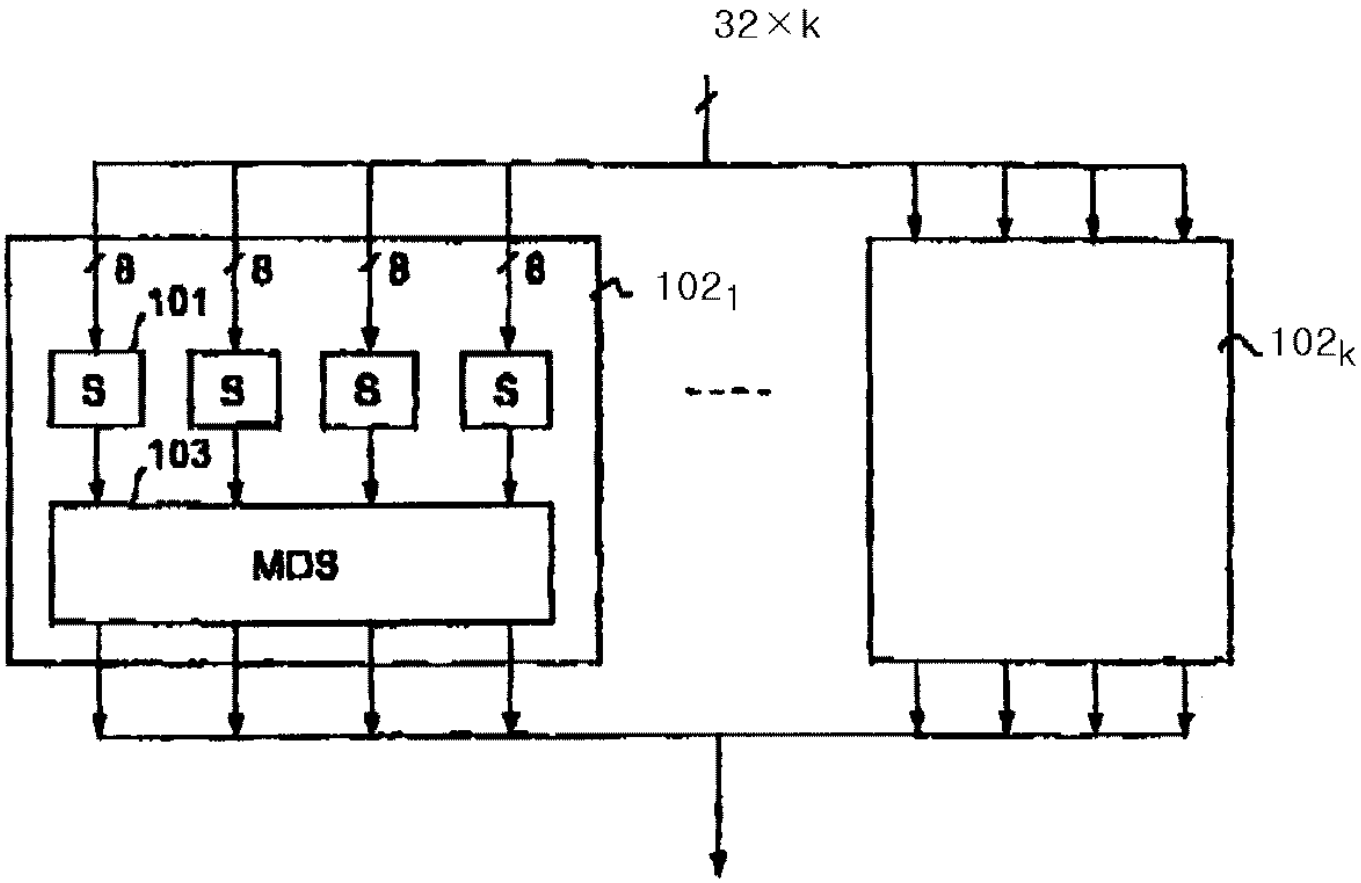
도면 17


4
44_0
공통키 (kc)
C_0
42_1
f_1^{-1}
44_1
C_1
f_2^{-1}
44_2
C_2
42_2
42_{m-1}
f_{m-1}^{-1}
44_{m-1}
C_{m-1}
42_m
f_m^{-1}
44_m
C_m
공통키 (kc)
평문
2
R_1^{-1}
22_1
R_2^{-1}
22_2
22_m
R_n^{-1}
암호문
16
디코더
8
확장공통키(kc')

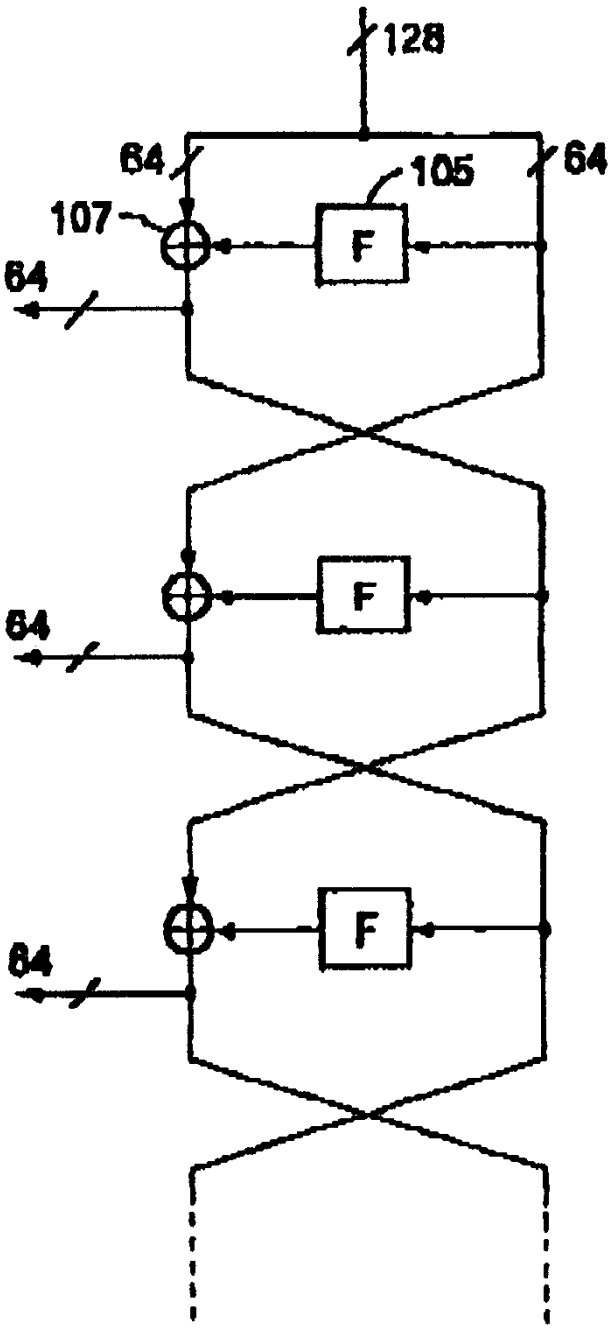
도면 18


1
평문
보조함수
13
R1
11 1
11 n
Rn
보조함수
13
암호문
3
공통키(ko)
33 1
f1
C1
31 1
33 2
f2
C2
31 2
31 (m-1)
fm-1
33 m
Cm
31 m
fm
공통키(kc)

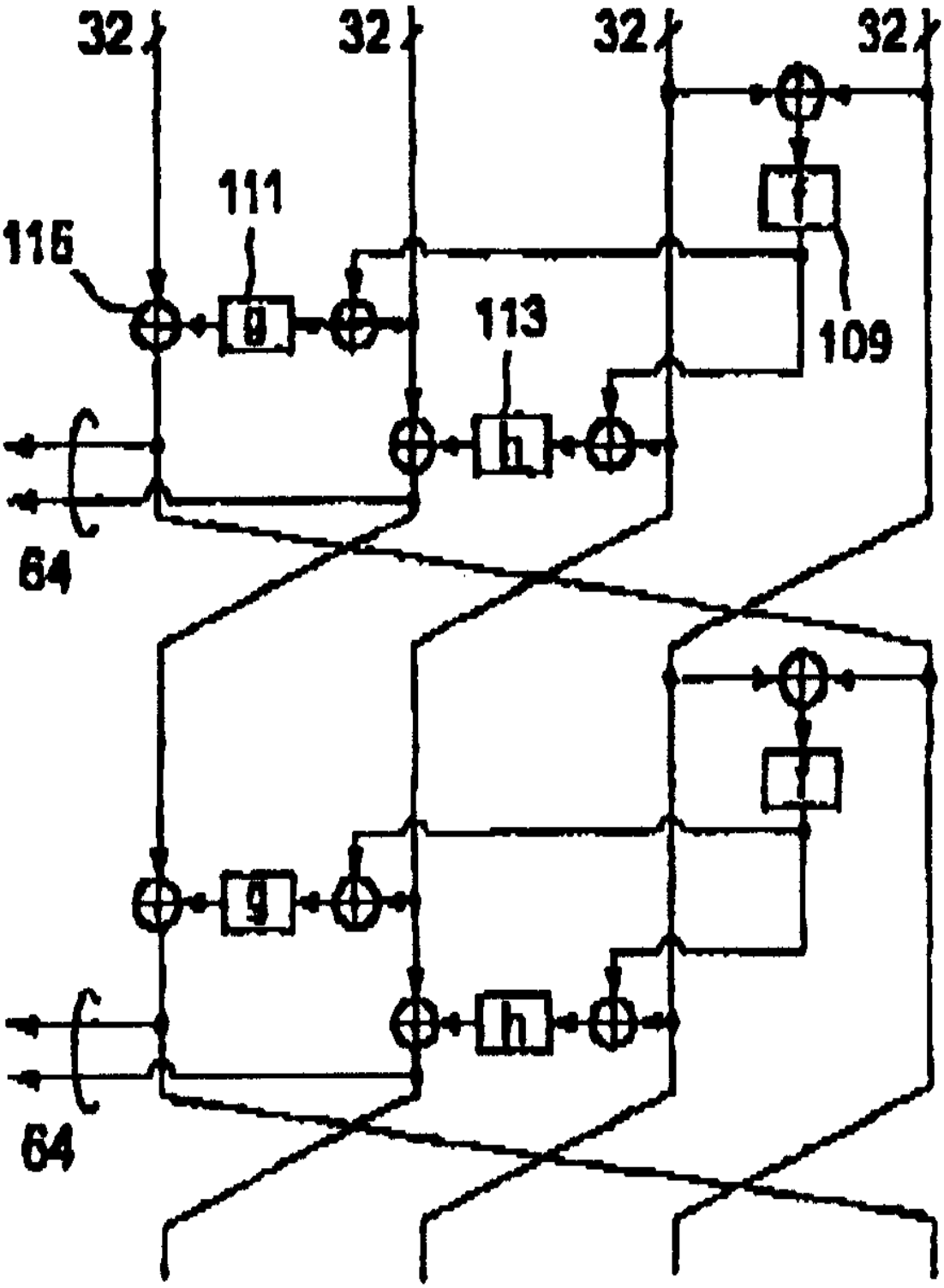
도면 19



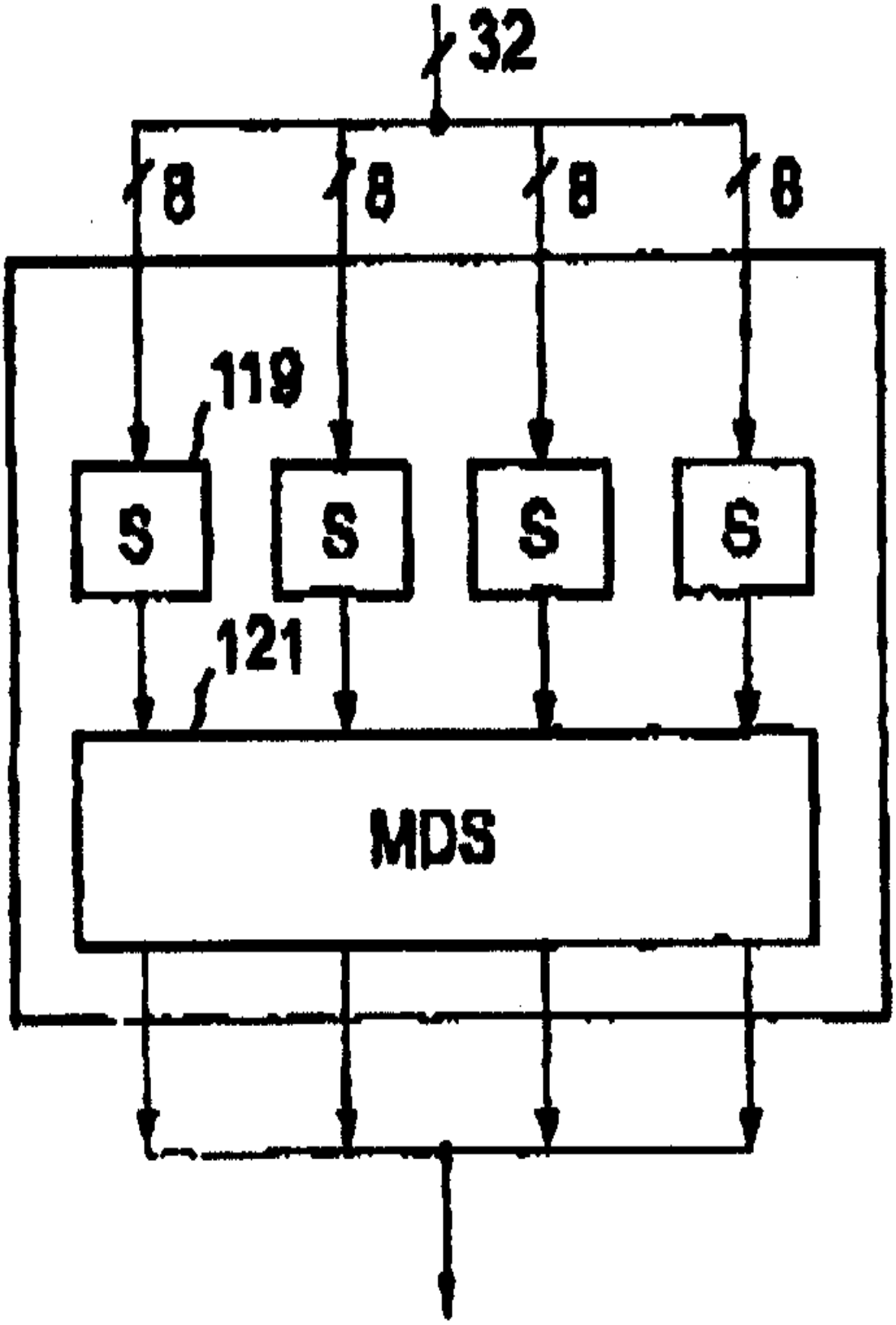
도면 20



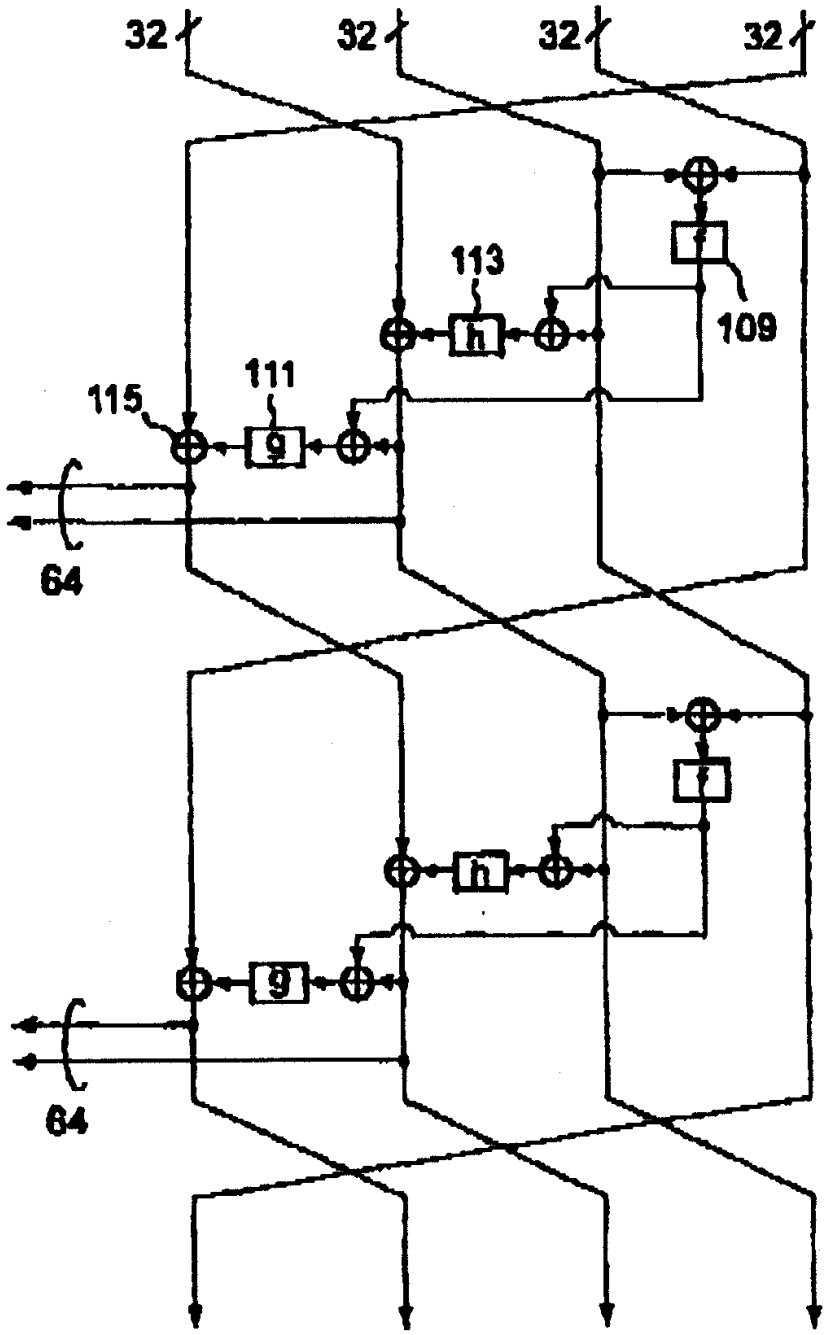
도면 21



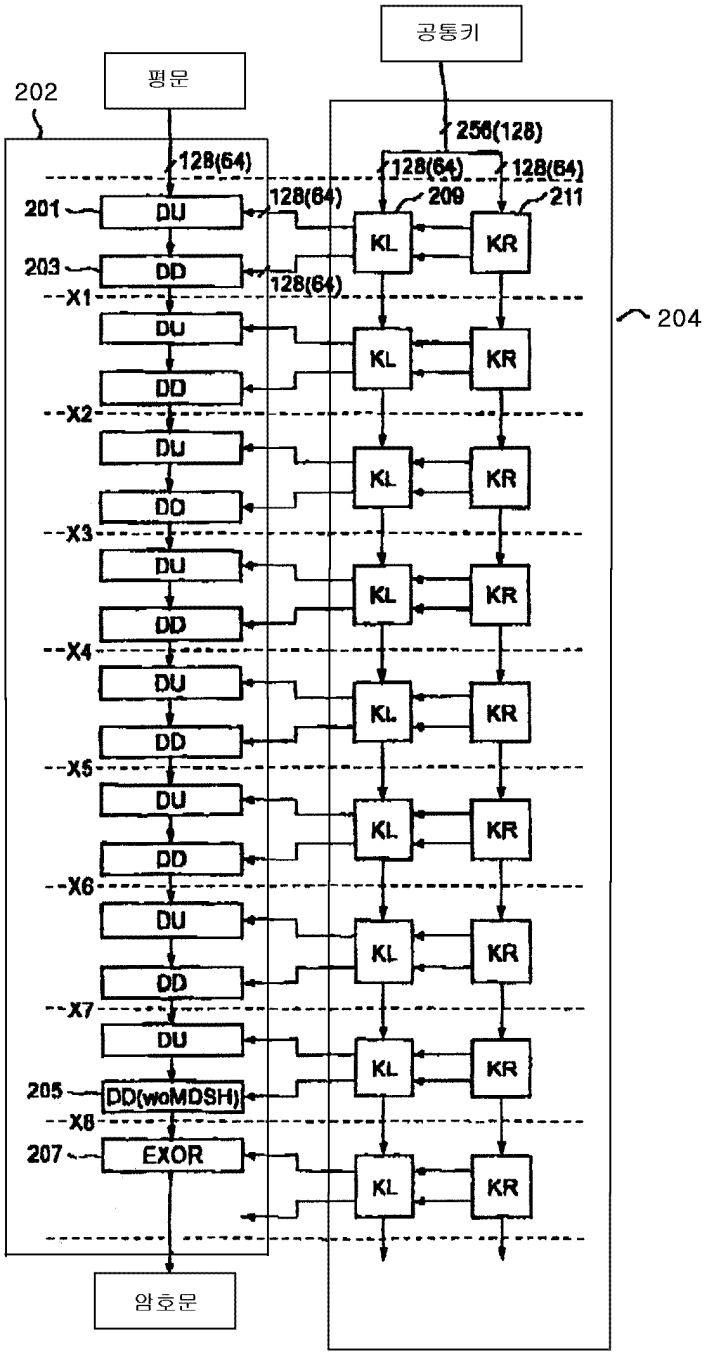
도면 22



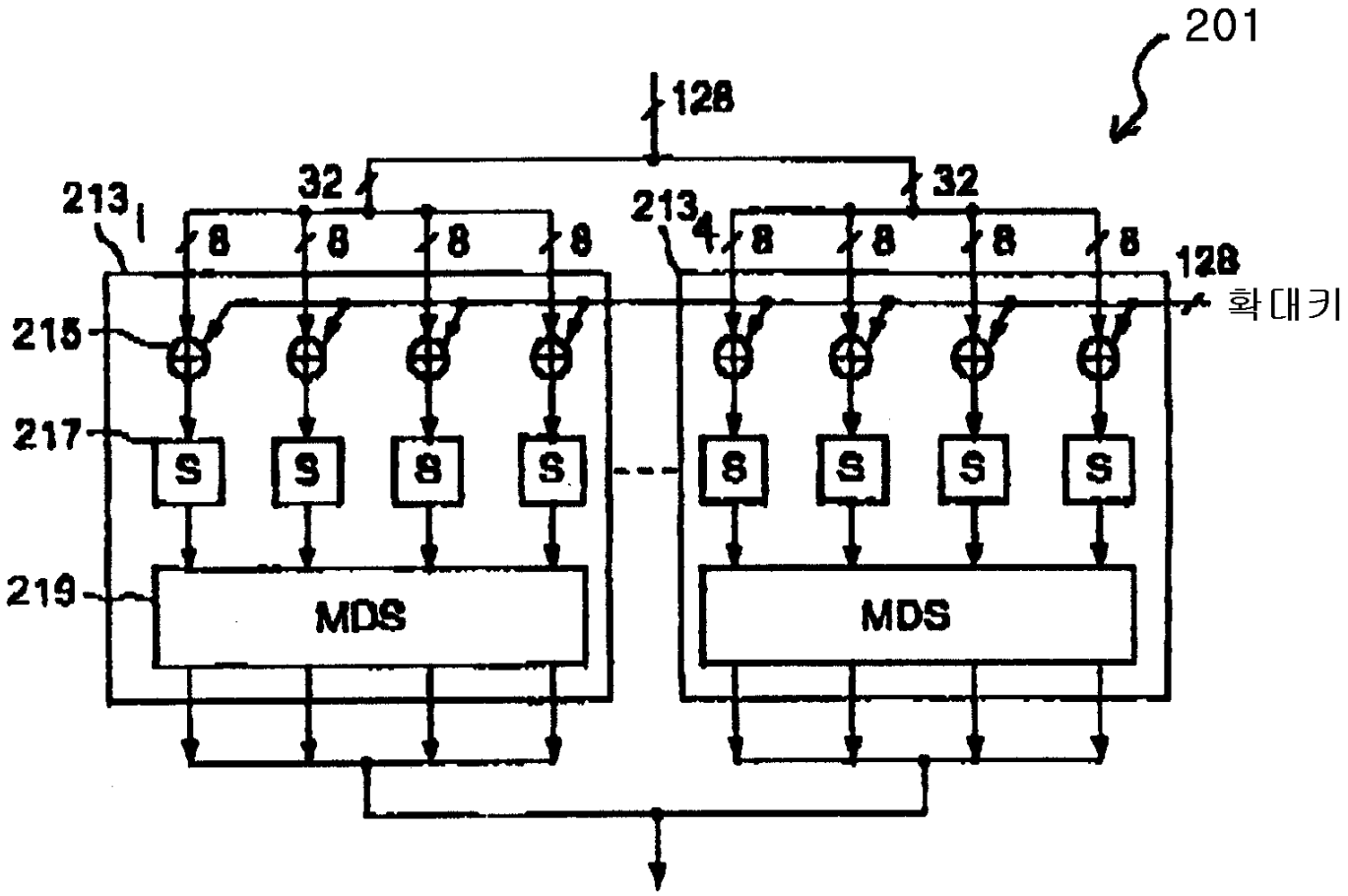
도면 23



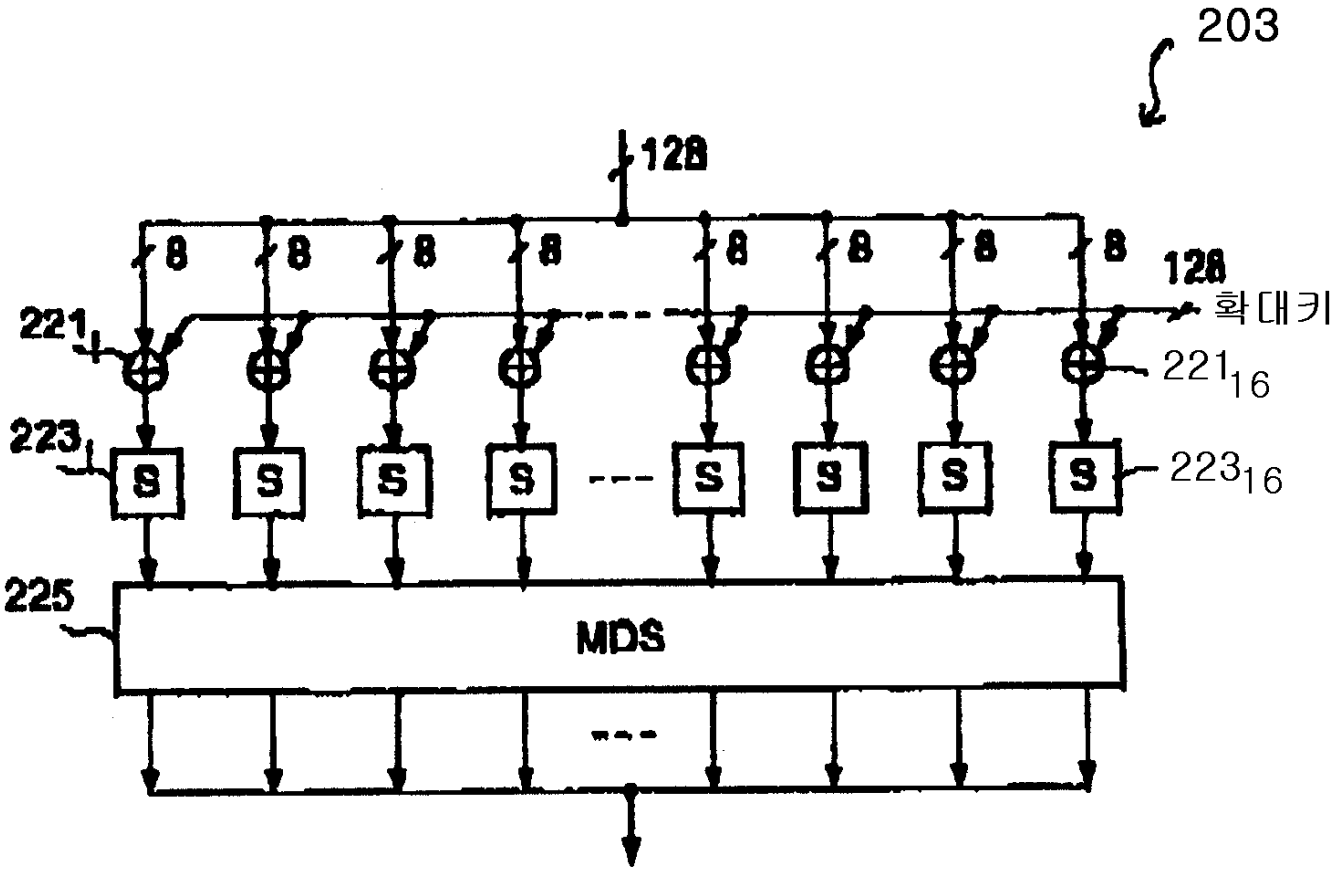
도면 24


공통키
평문
202
256(128)
128(64)
128(64)
128(64)
128(64)
209
211
201
203
DU
DD
KL
KR
X1
X2
X3
X4
X5
X6
X7
X8
204
205
DD(woMDSH)
207
EXOR
암호문

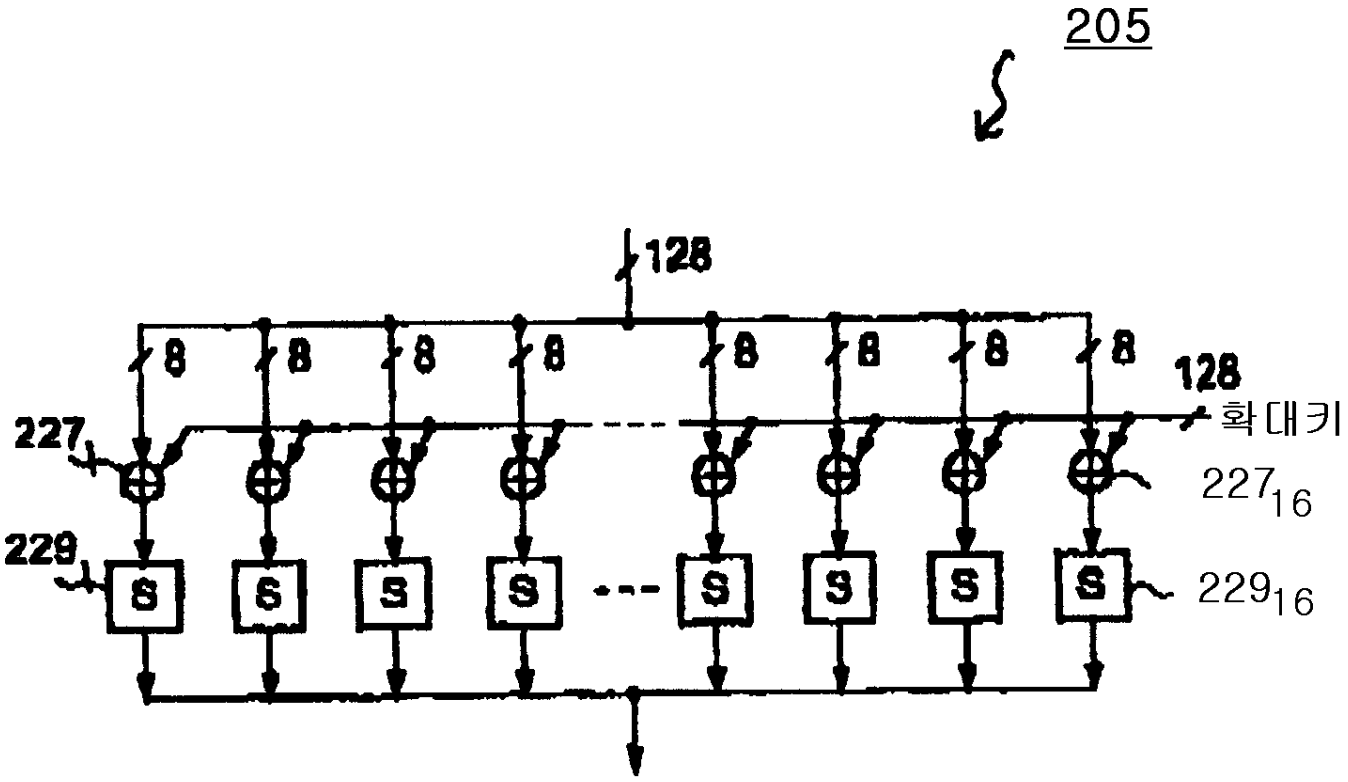
도면 25


201
128
32
32
213
213
8
8
8
8
8
8
8
8
128
확대키
215
217
219
S
S
S
S
S
S
S
S
MDS
MDS

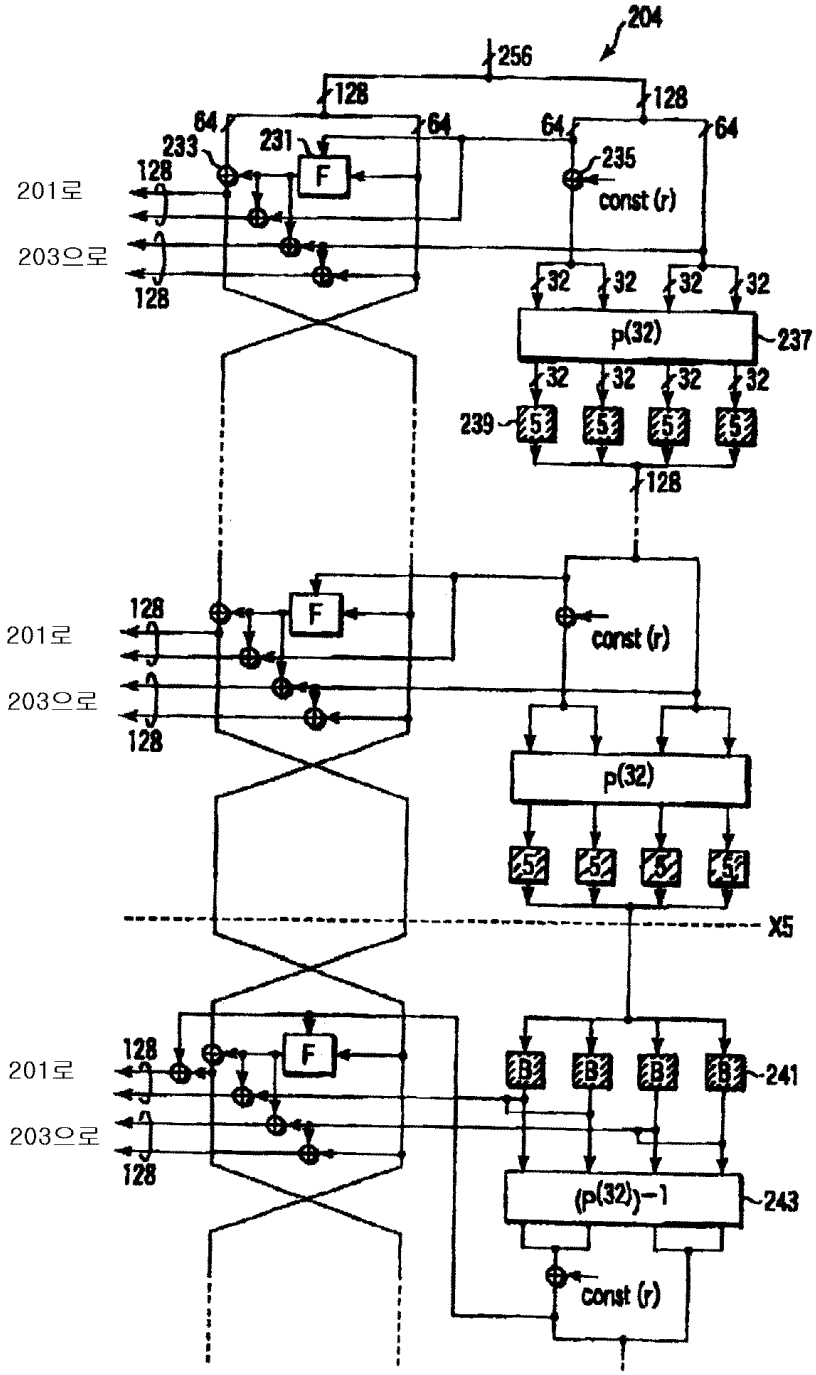
도면 26


203
128
8
128
확대키
221
221_{16}
223
S
223_{16}
225
MDS

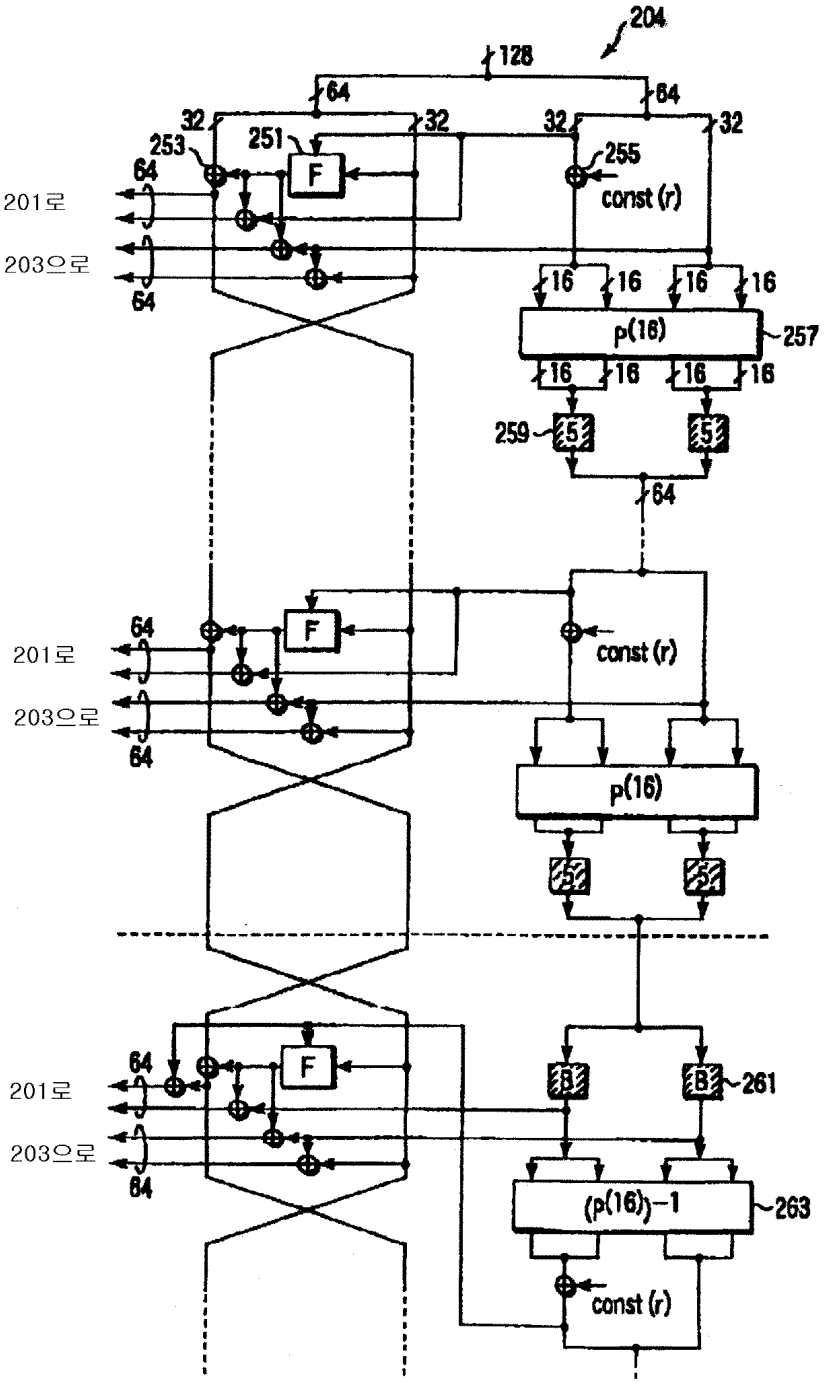
도면 27


205
128
8
128
확대키
227
227_{16}
229
S
229_{16}

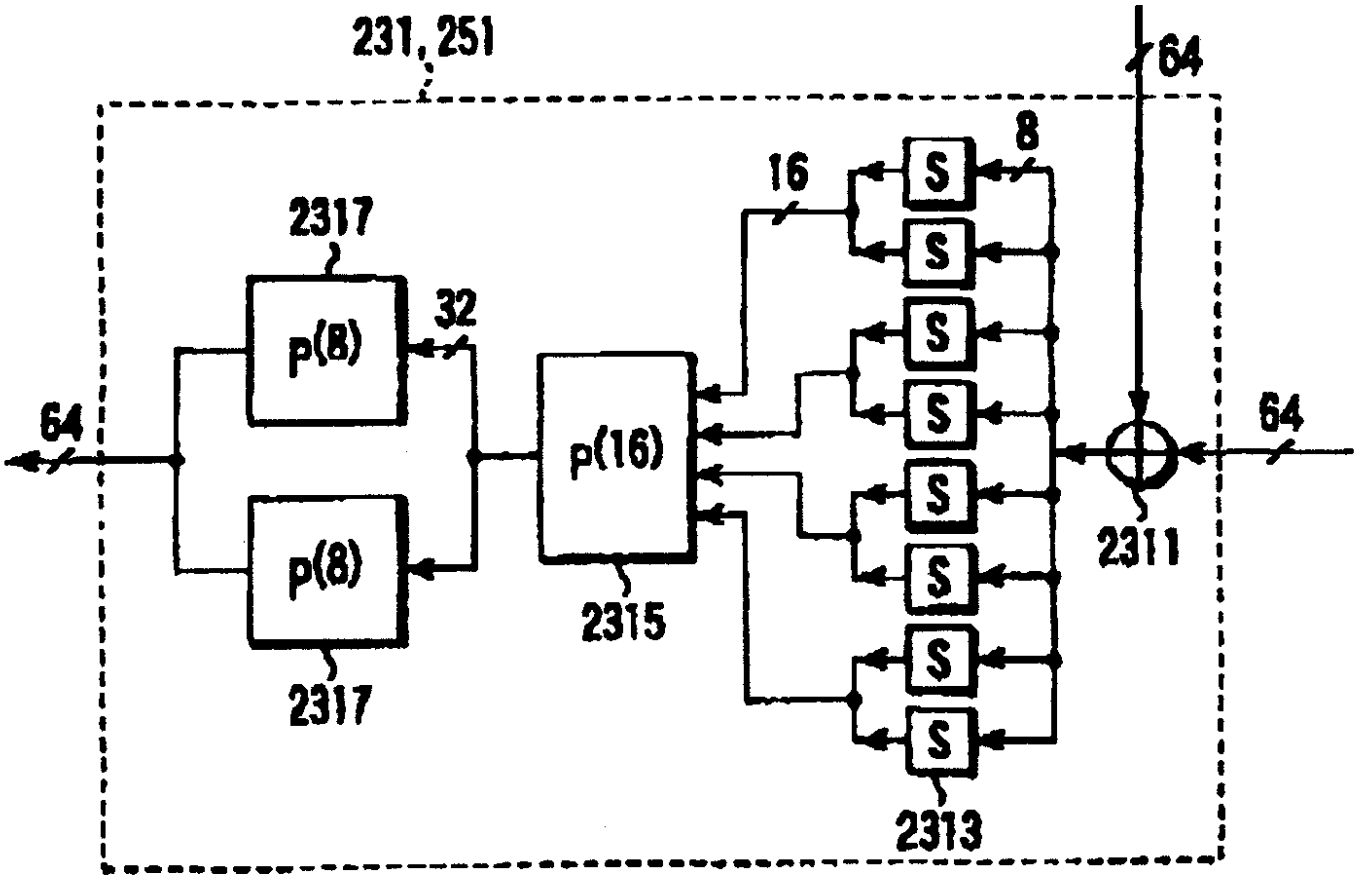
도면 28


204
256
128
128
64
64
64
64
233
231
235
F
const (r)
128
201로
203으로
128
32
32
32
32
p(32)
237
239
128
X5
241
(p(32))-1
243

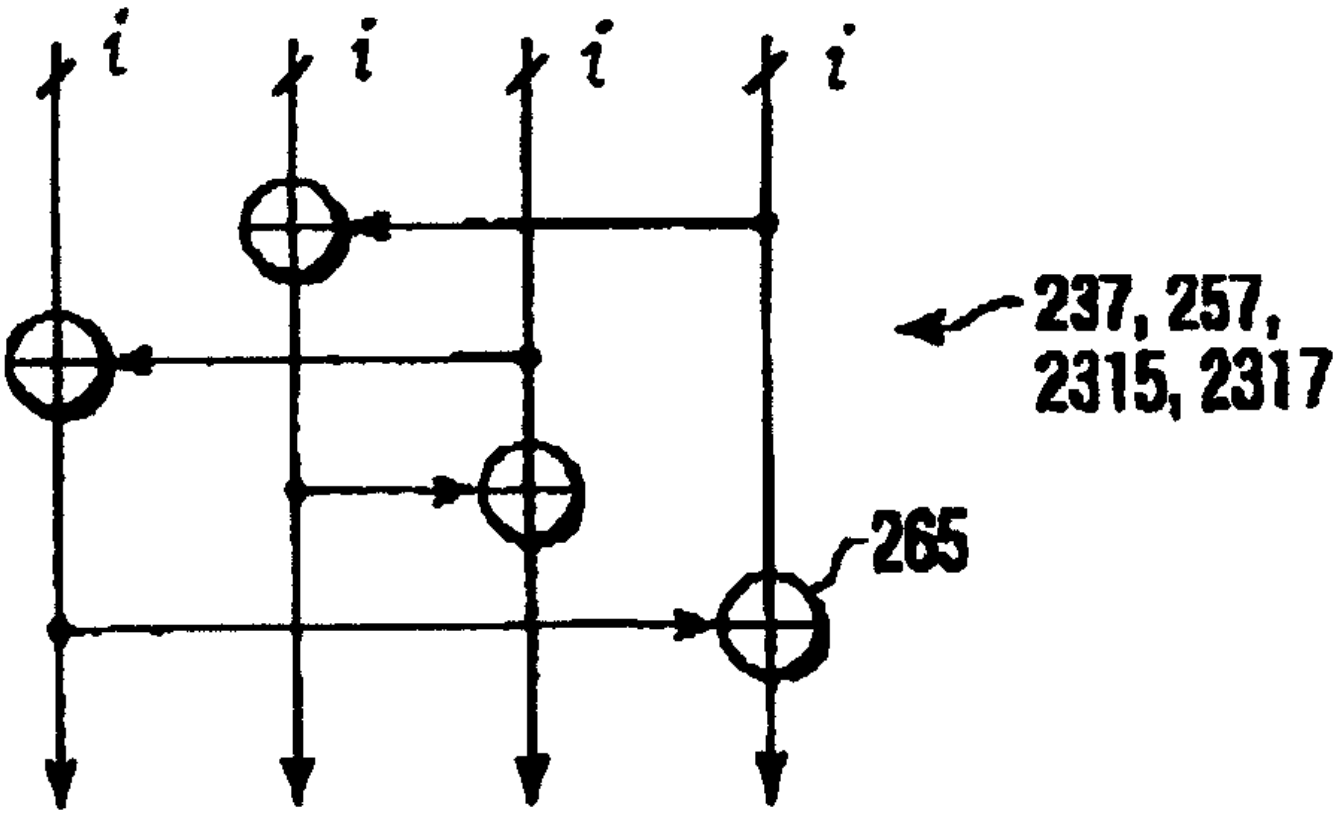
도면 29


204
128
64
64
32
32
32
32
253
251
255
F
const (r)
64
201로
203으로
64
16
16
16
16
p(16)
257
16
16
16
16
259
5
5
64
64
F
const (r)
201로
203으로
64
p(16)
B
B
64
F
201로
203으로
64
261
(p(16))-1
263
const (r)

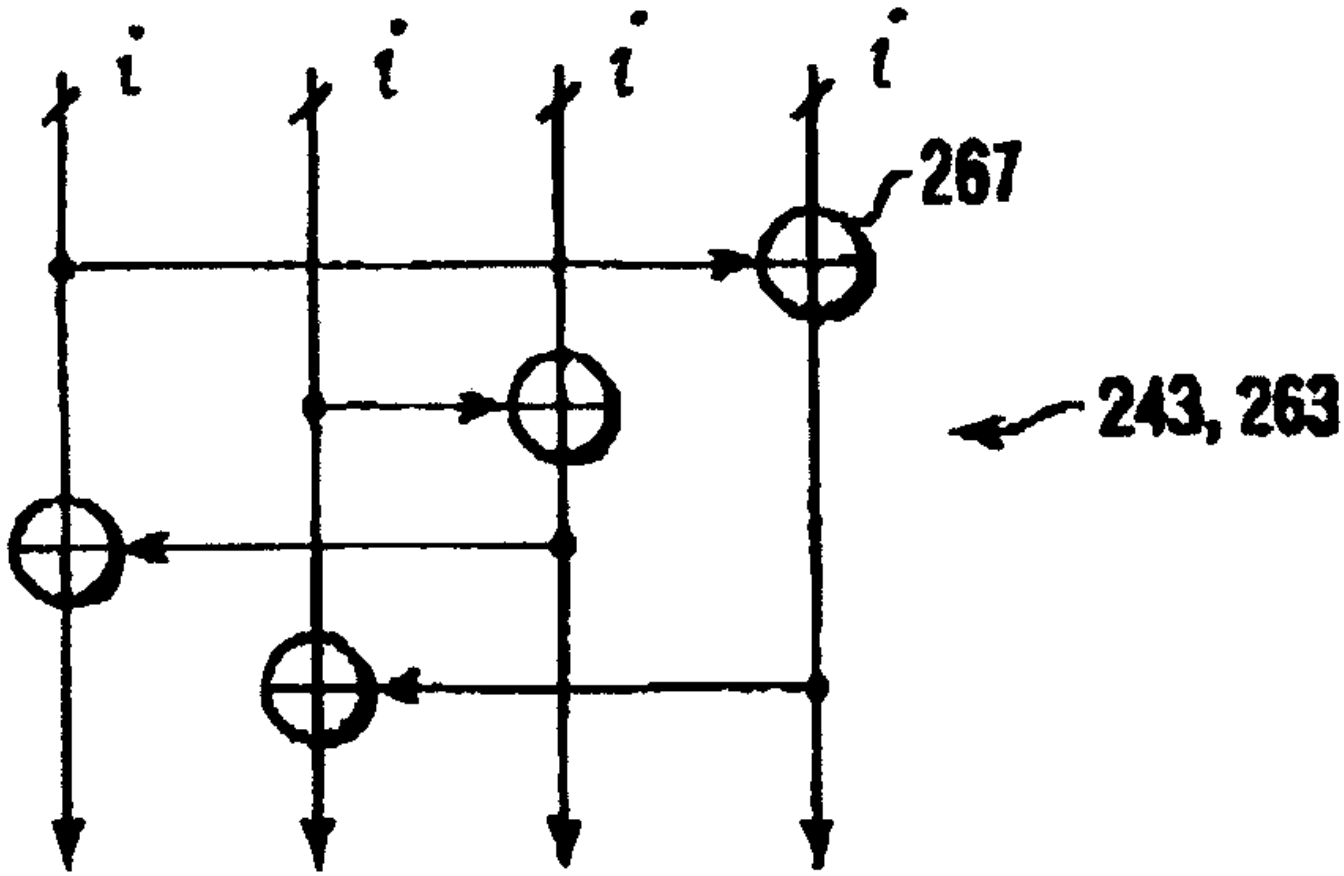
도면 30



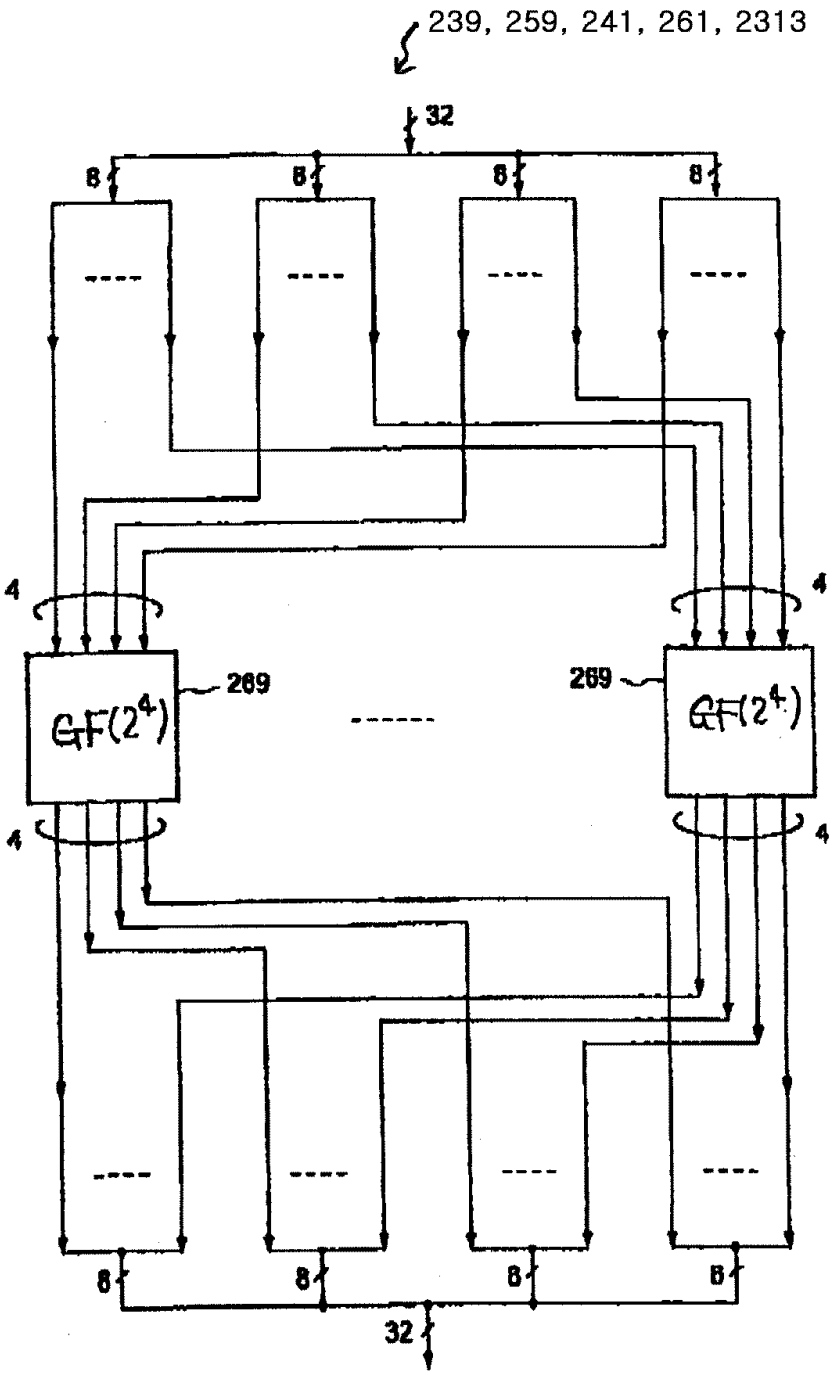
도면 31



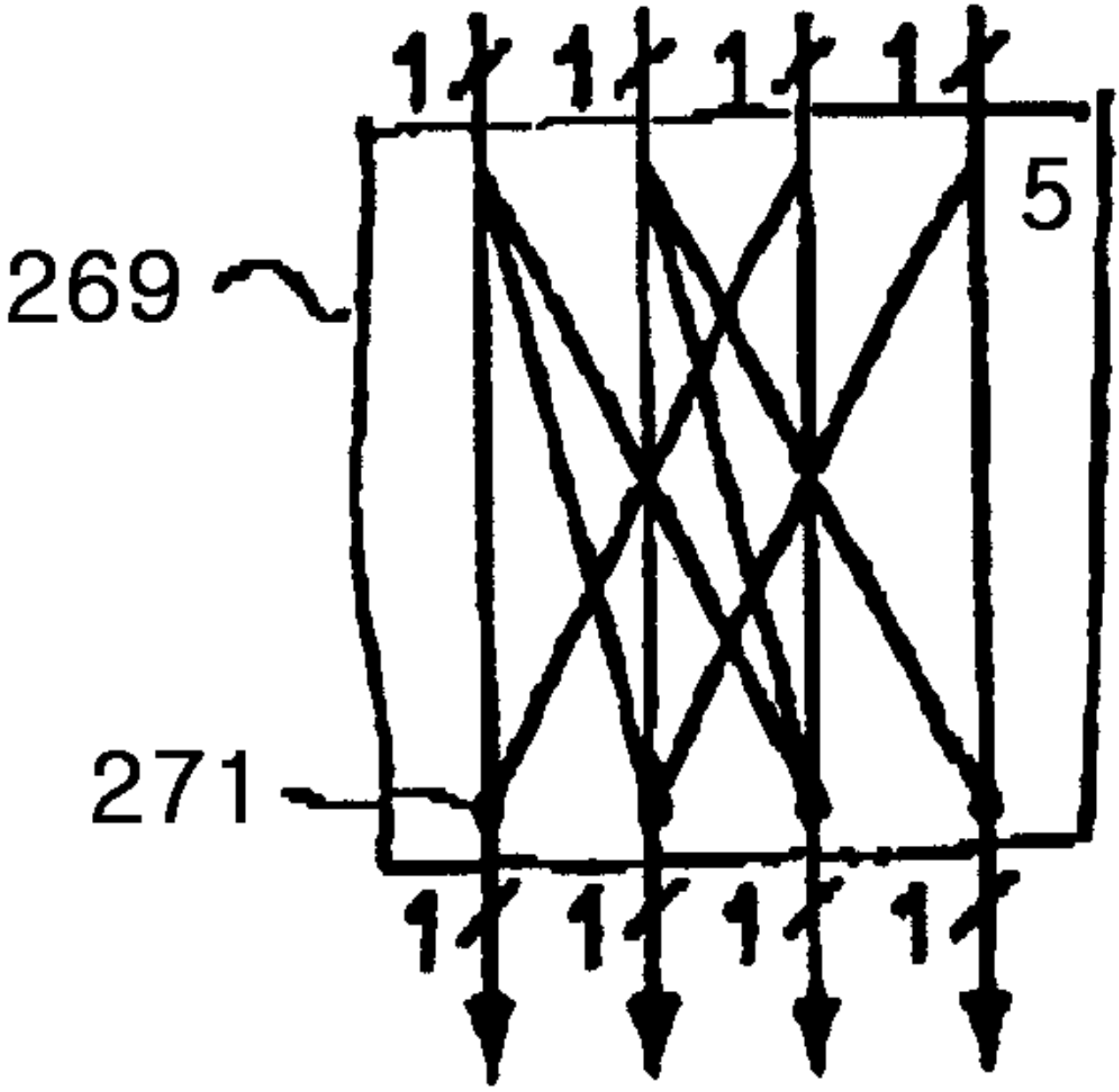
도면 32



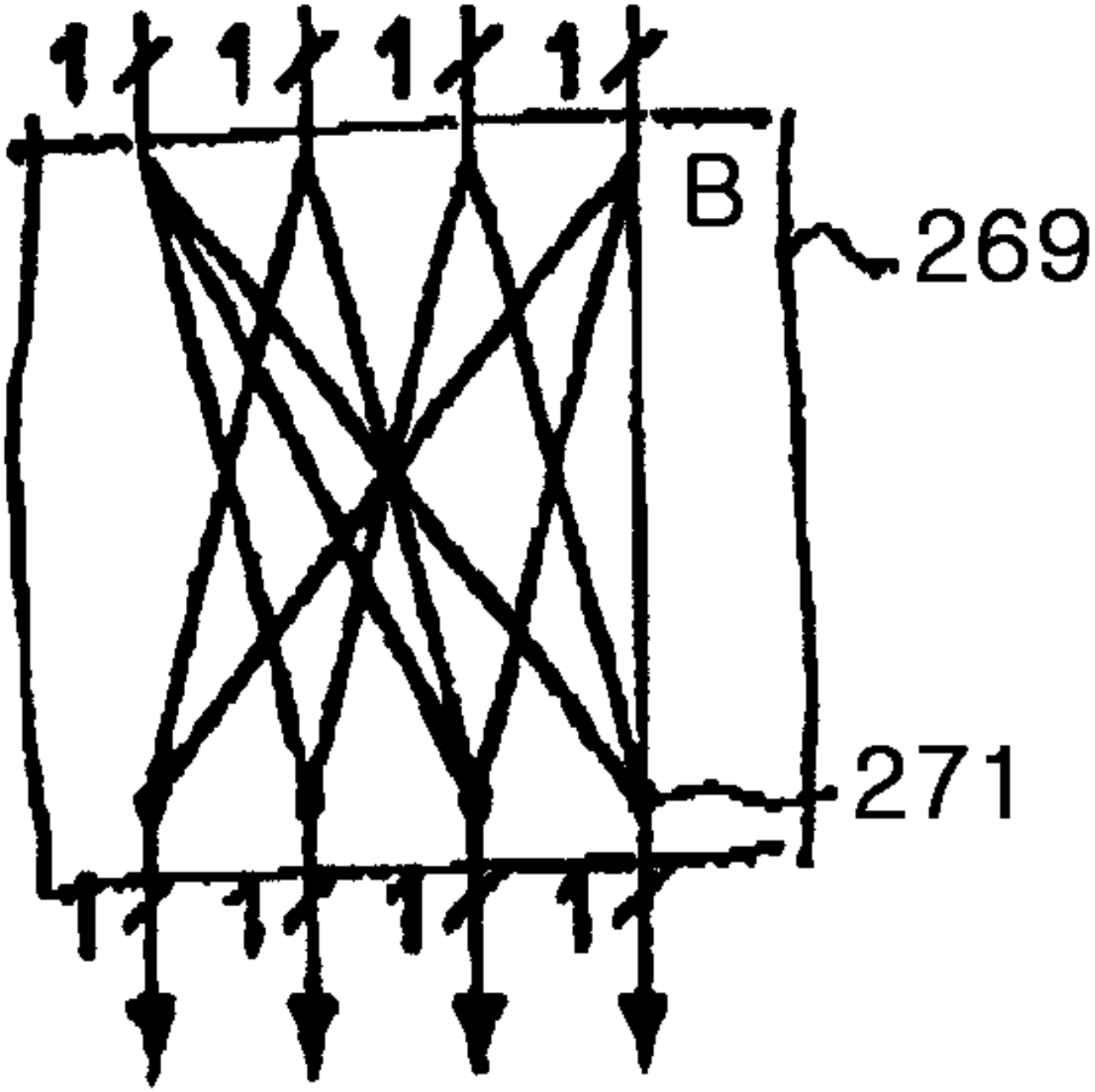
도면 33



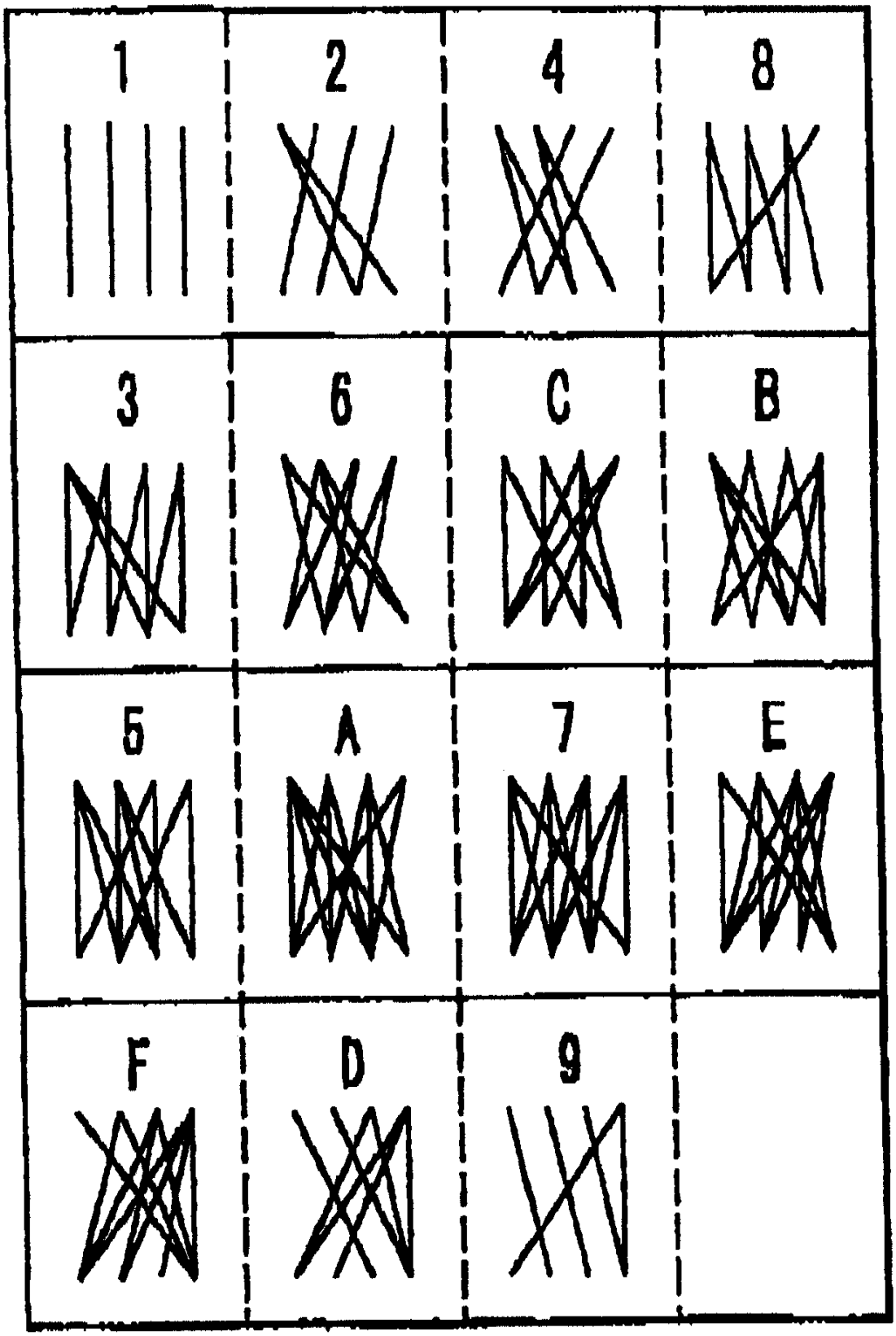
도면 34a



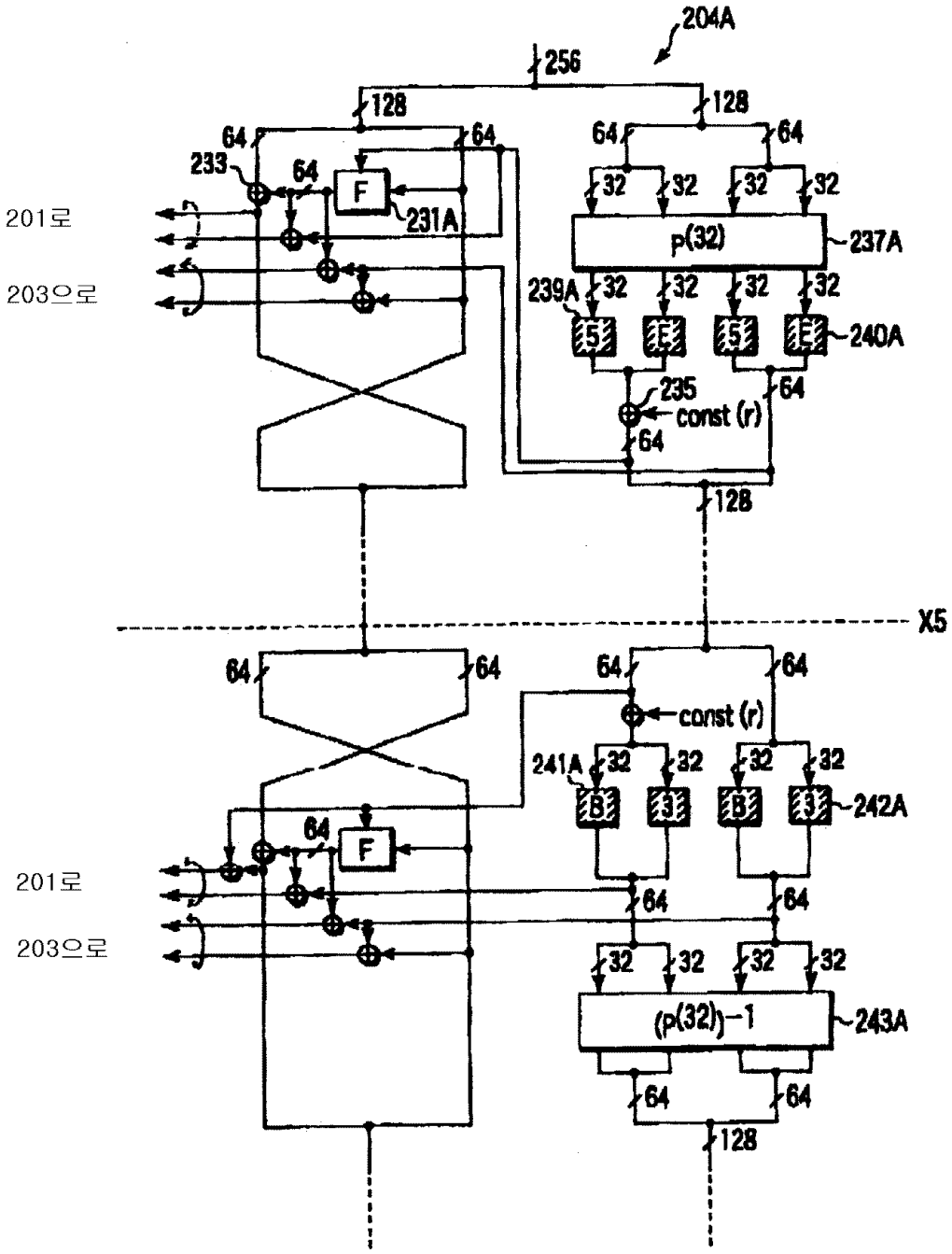
도면 34b



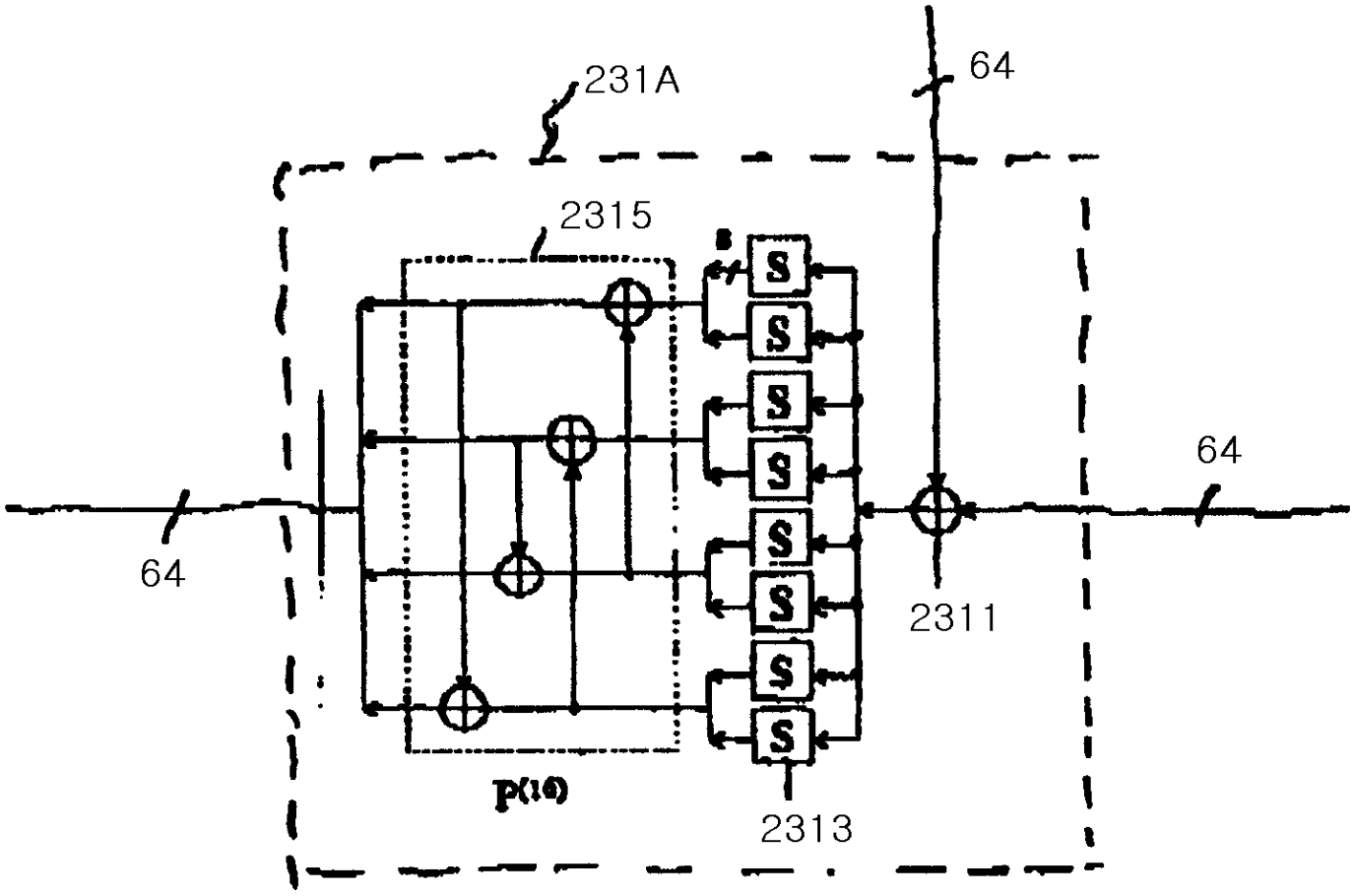
도면 35



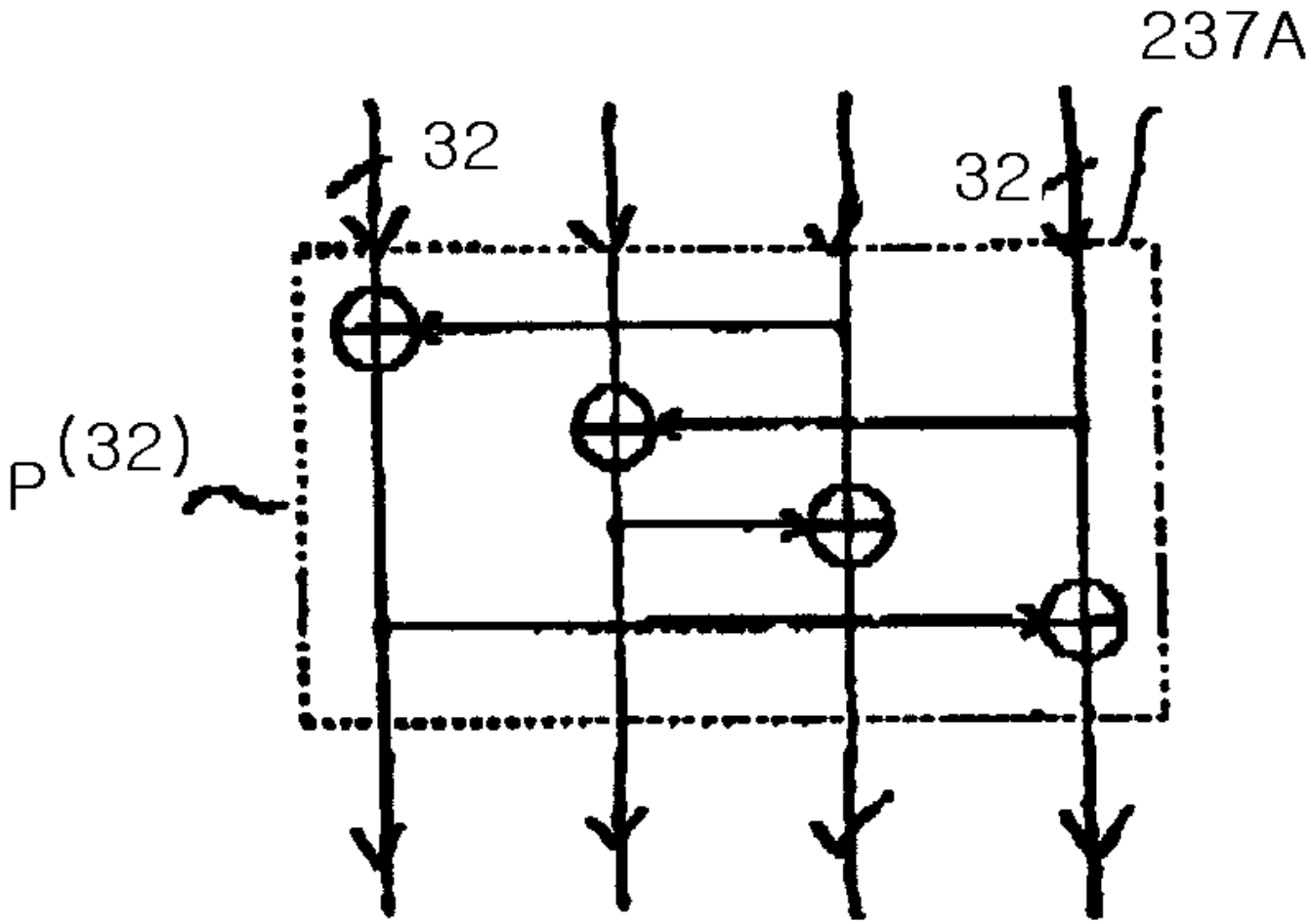
도면 36


204A
256
128
128
64
64
64
64
233
64
F
231A
32
32
32
32
p(32)
237A
201로
203으로
239A
32
32
32
32
240A
235
const (r)
64
64
128
X5
64
64
64
64
const (r)
241A
32
32
32
32
242A
64
F
64
64
201로
203으로
32
32
32
32
(p(32))-1
243A
64
64
128

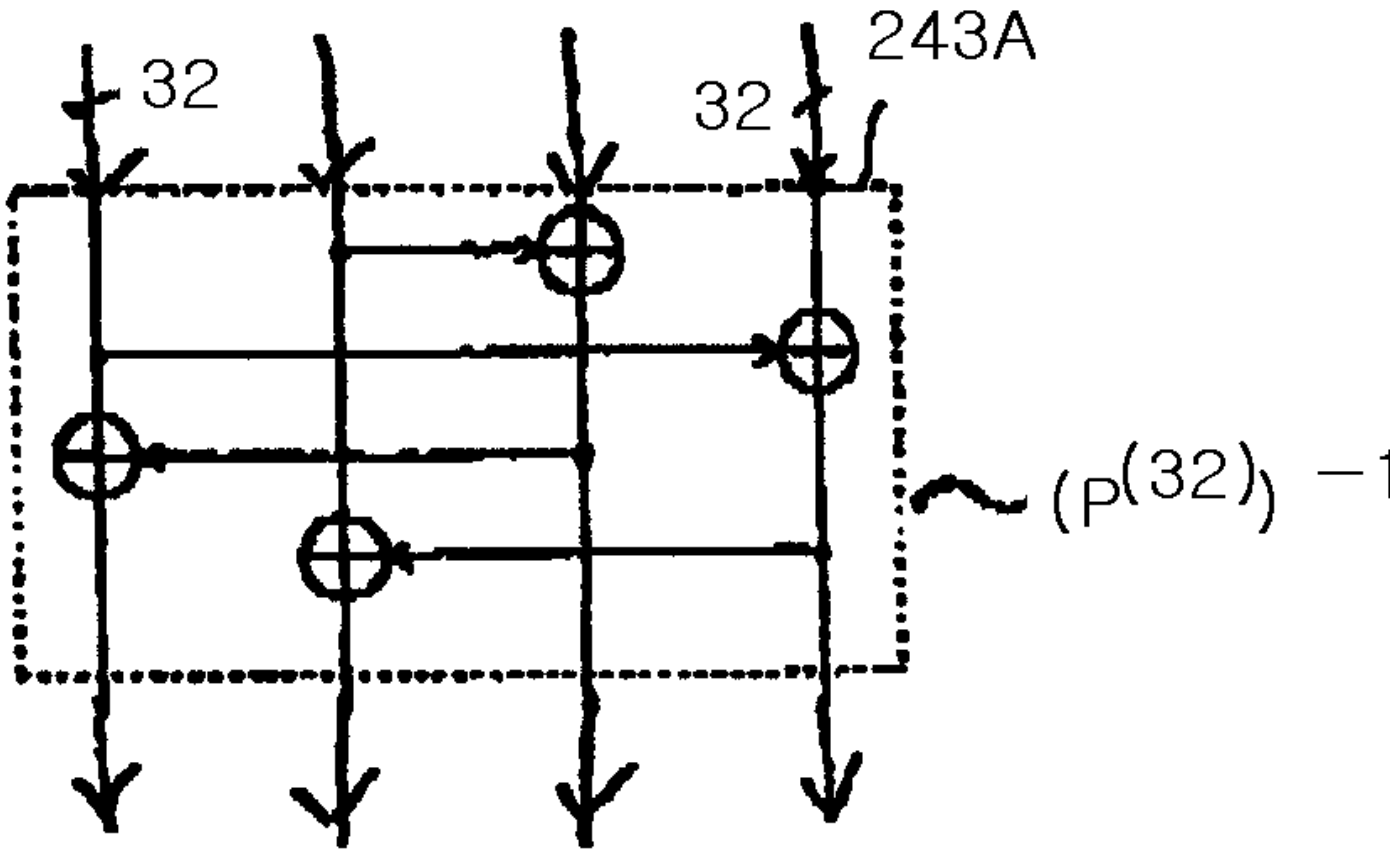
도면 37



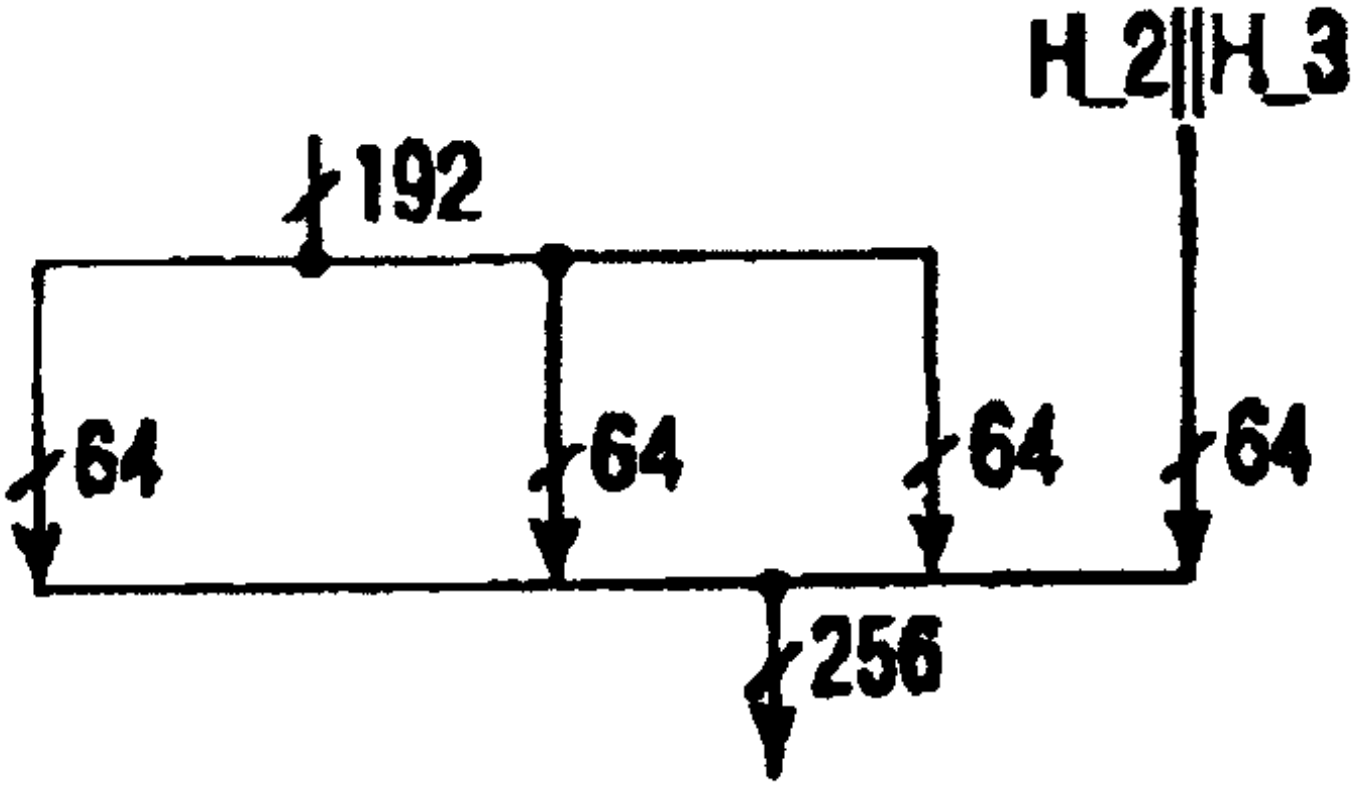
도면 38a



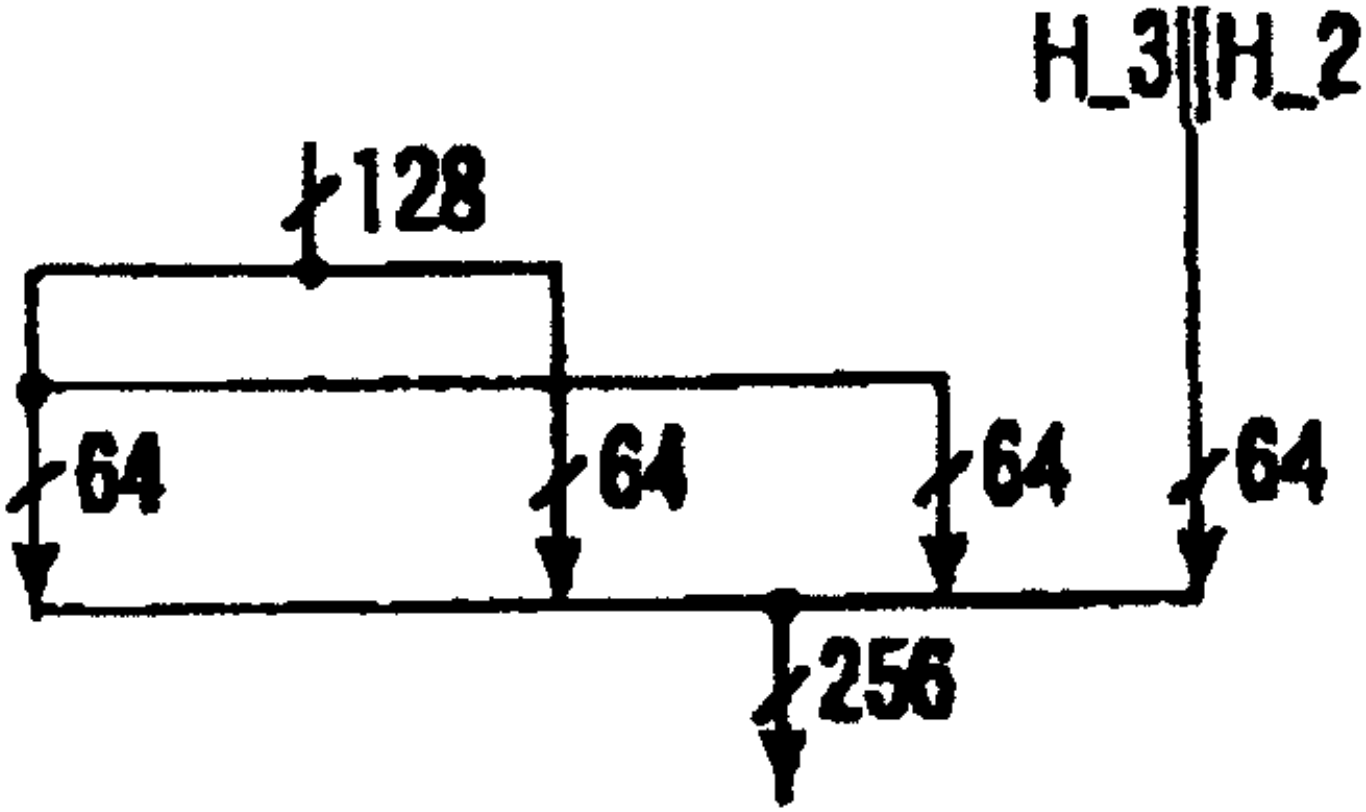
도면 38b



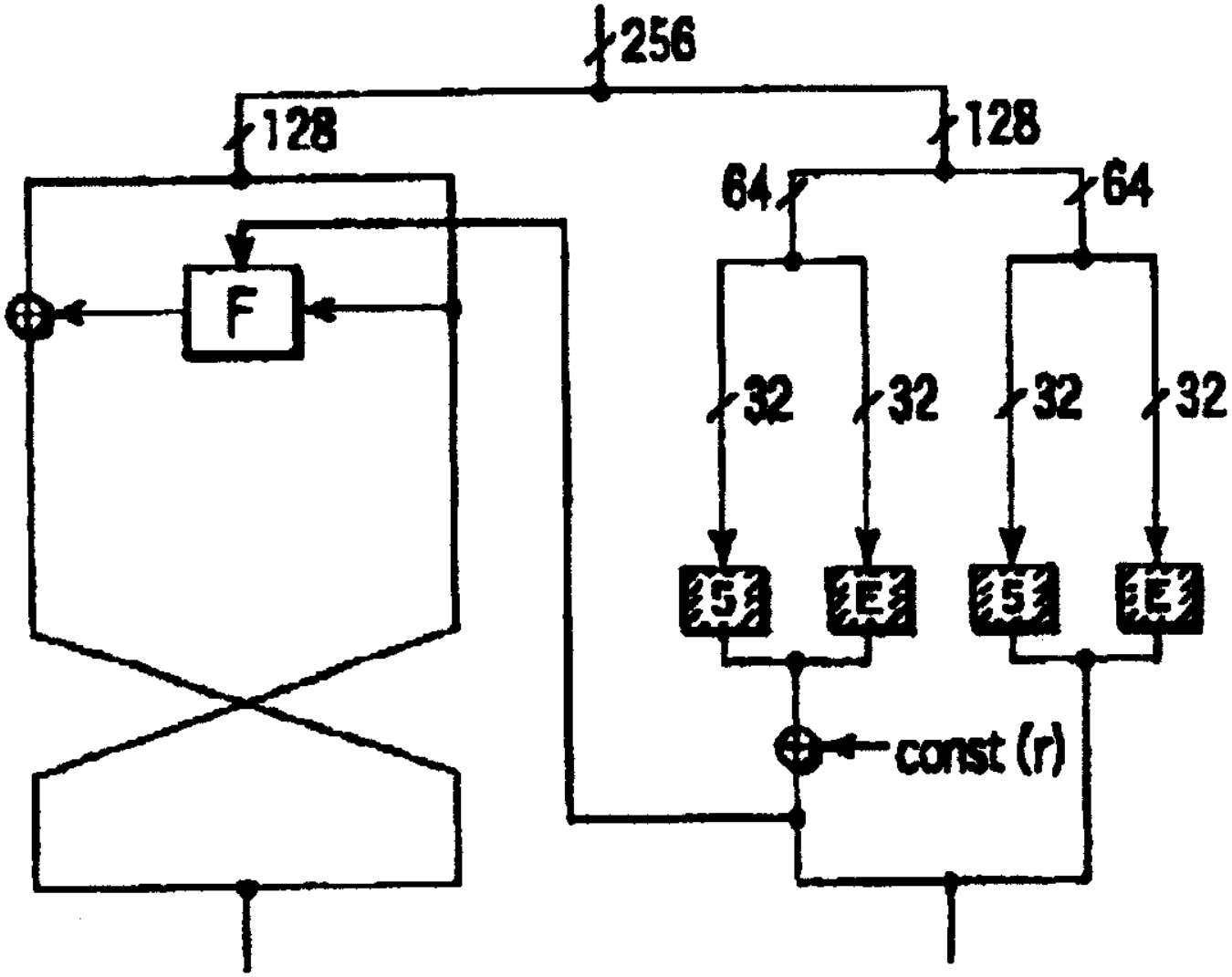
도면 39a



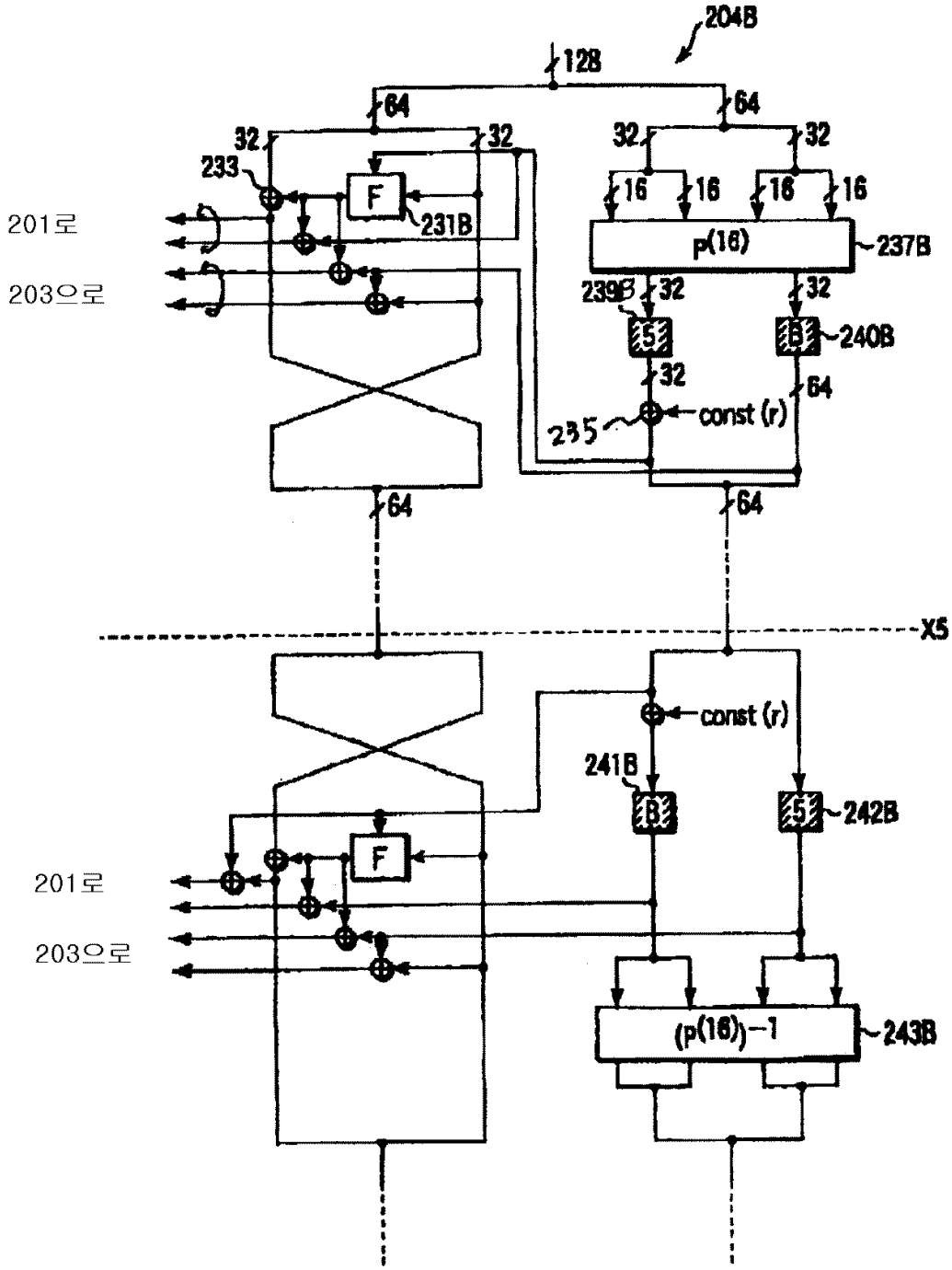
도면 39b



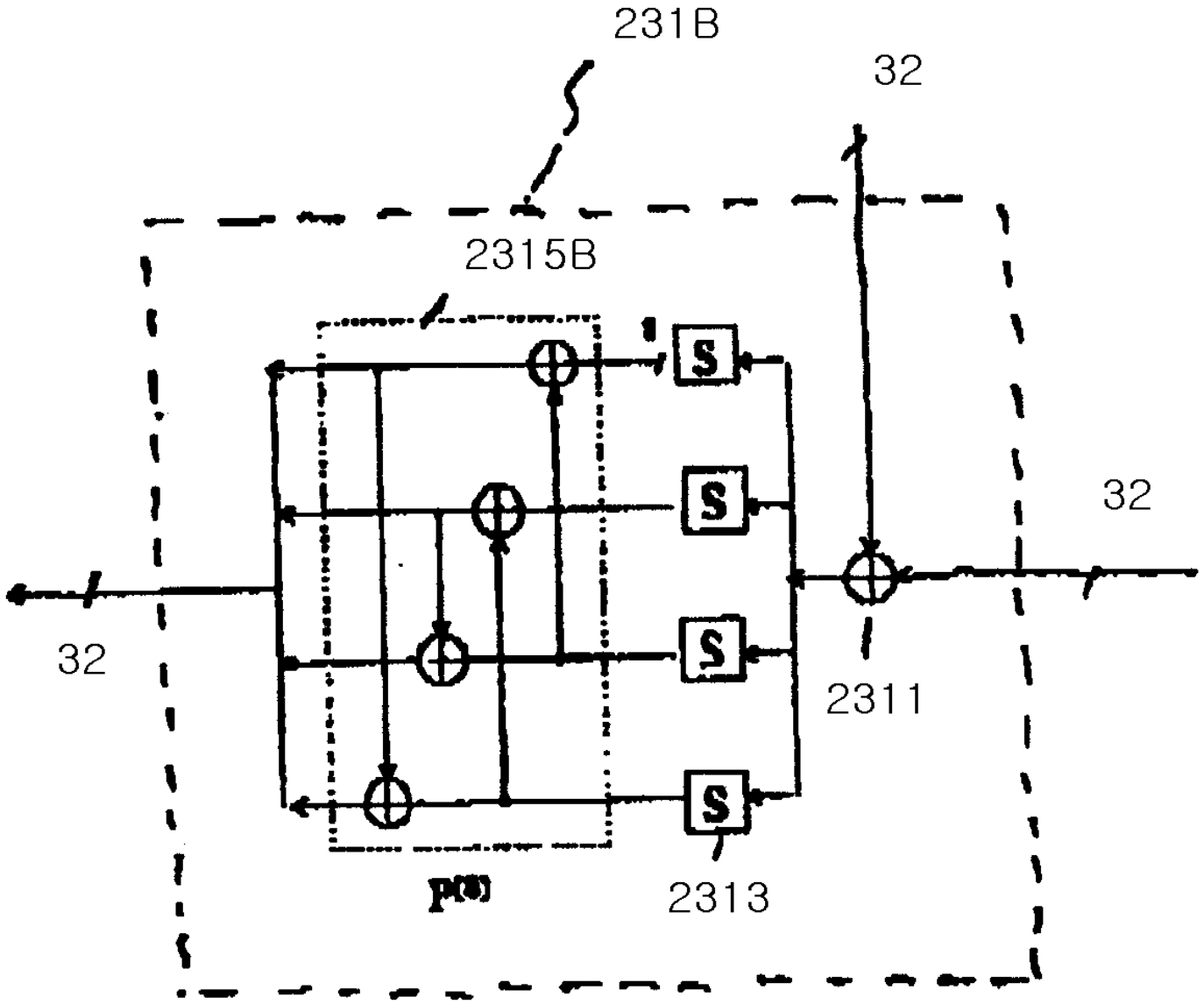
도면 40



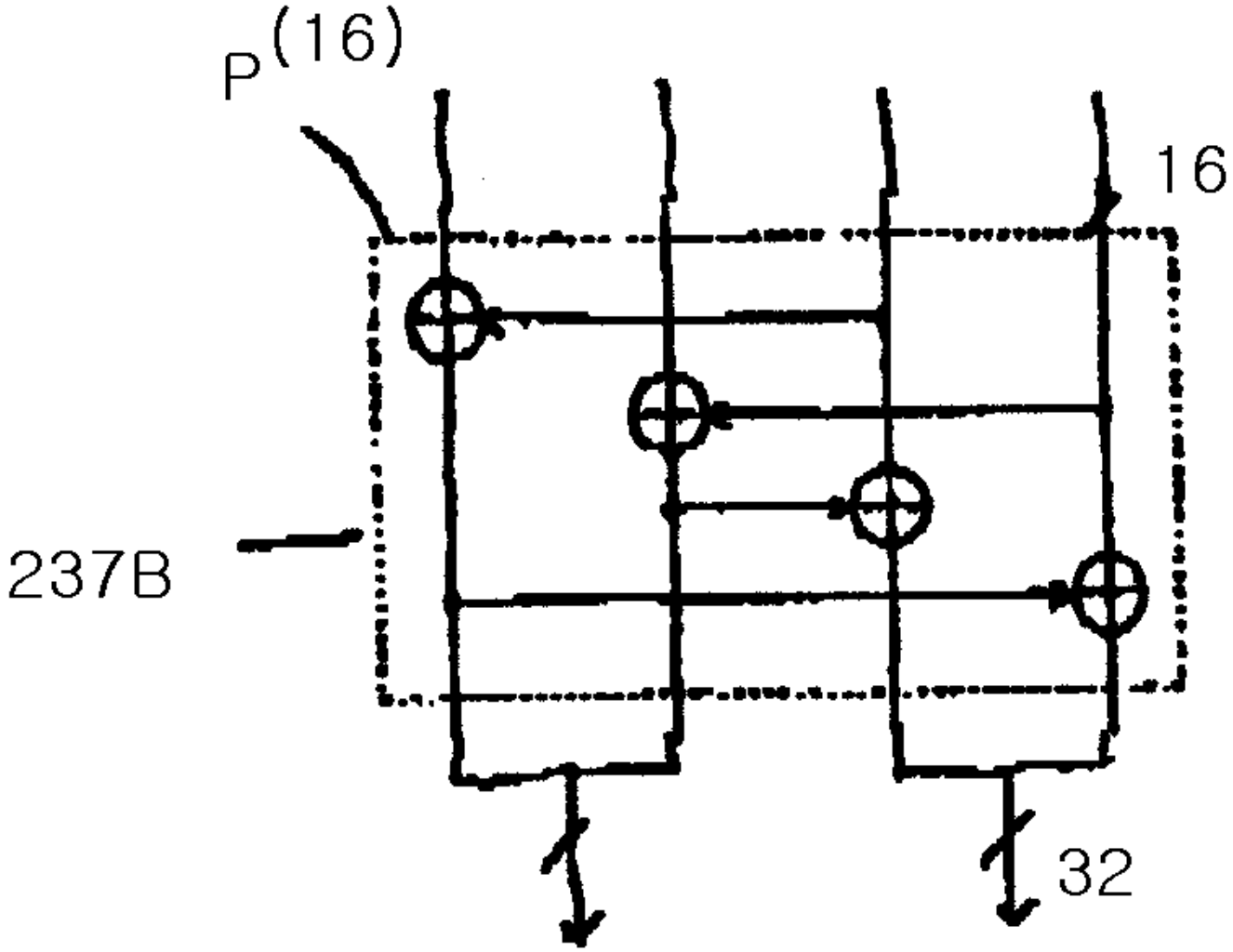
도면 41


204B
128
64
64
32
32
32
32
233
F
231B
16
16
16
16
201로
203으로
p(16)
237B
239B
32
32
5
8
240B
32
64
235
const (r)
64
64
X5
const (r)
241B
8
5
242B
F
201로
203으로
(p(16))-1
243B

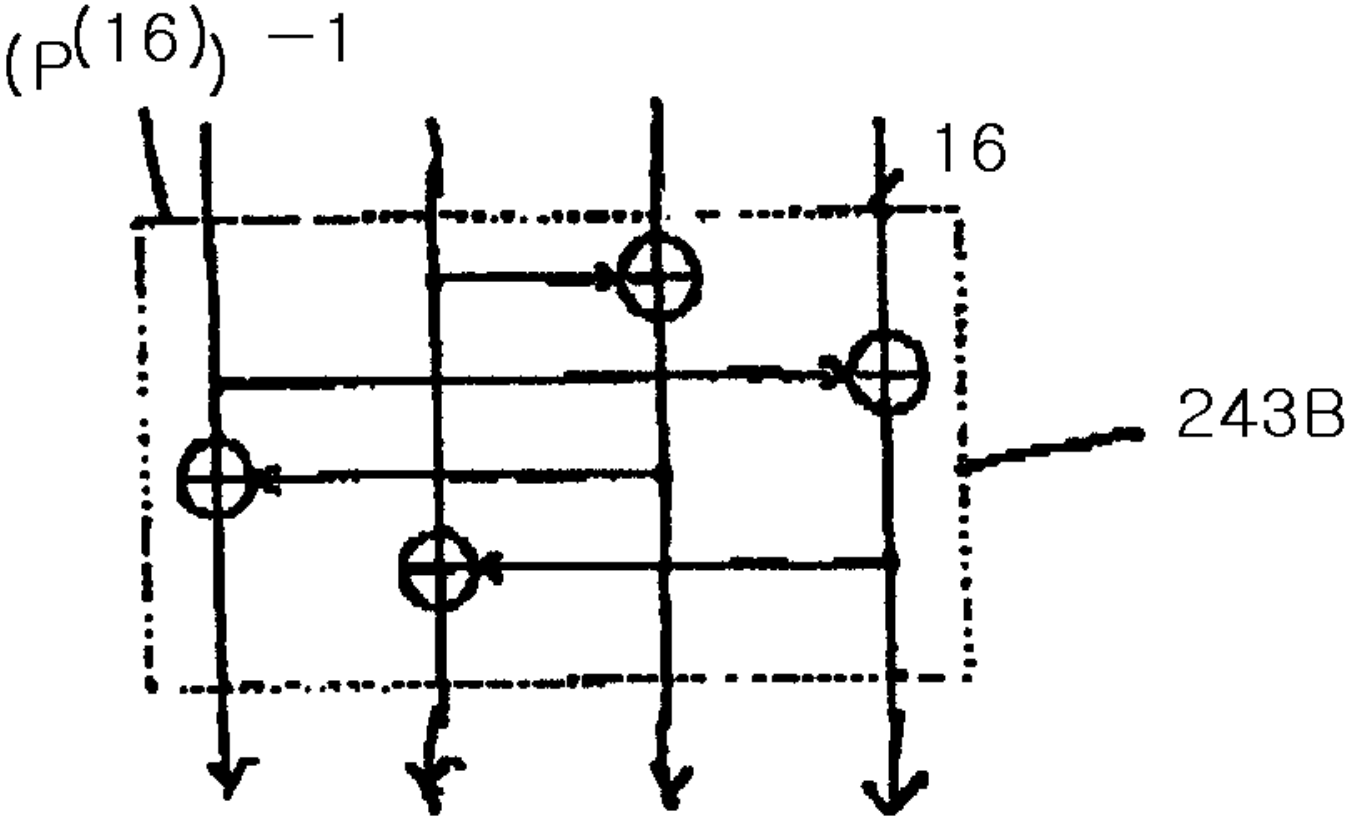
도면 42



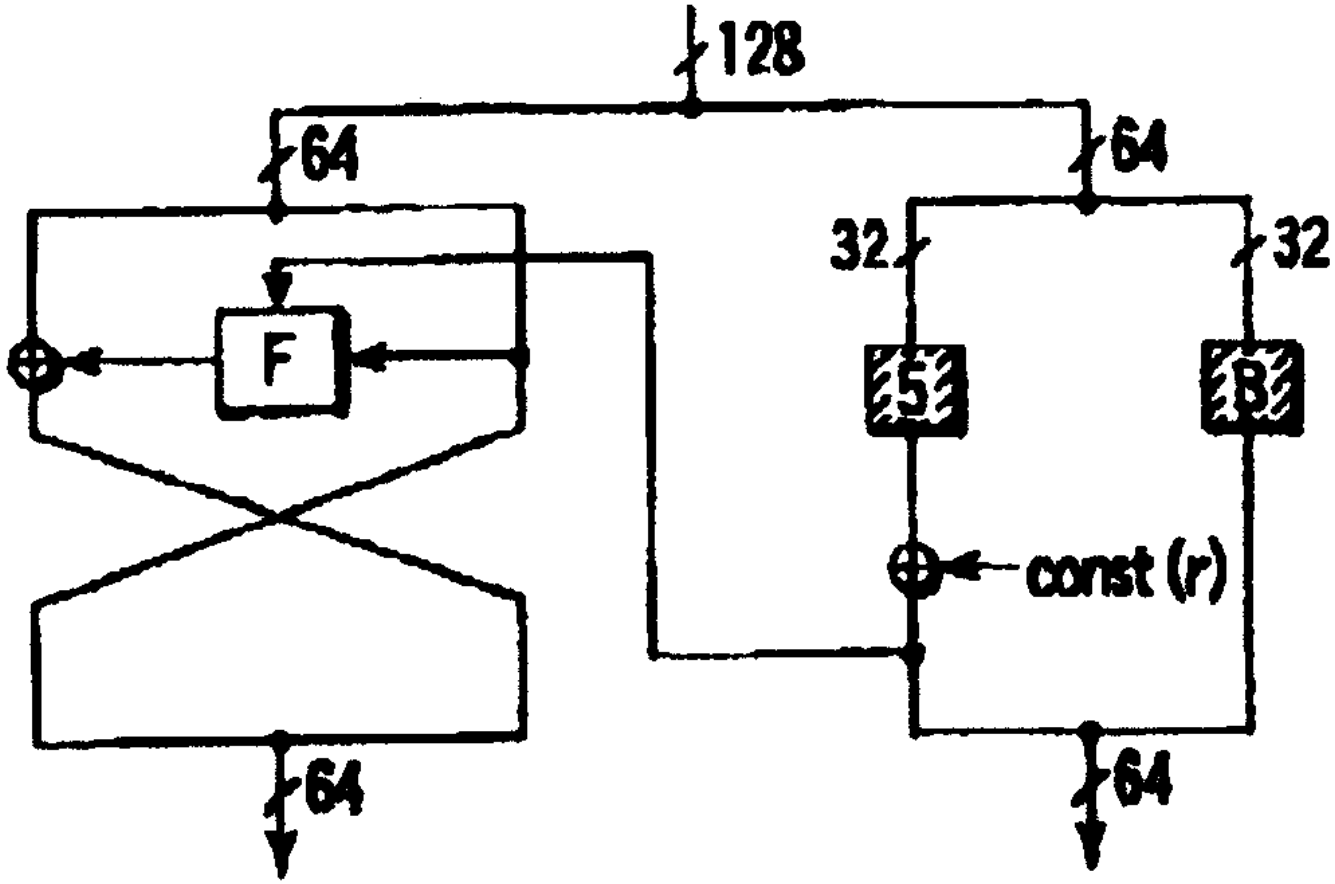
도면 43a



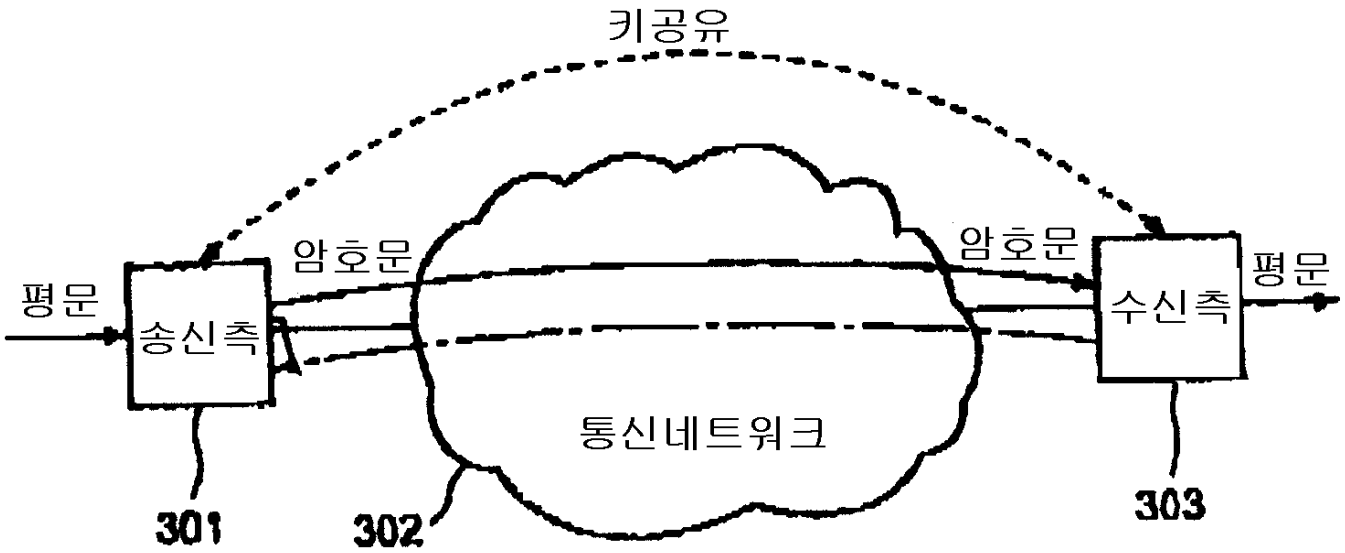
도면 43b



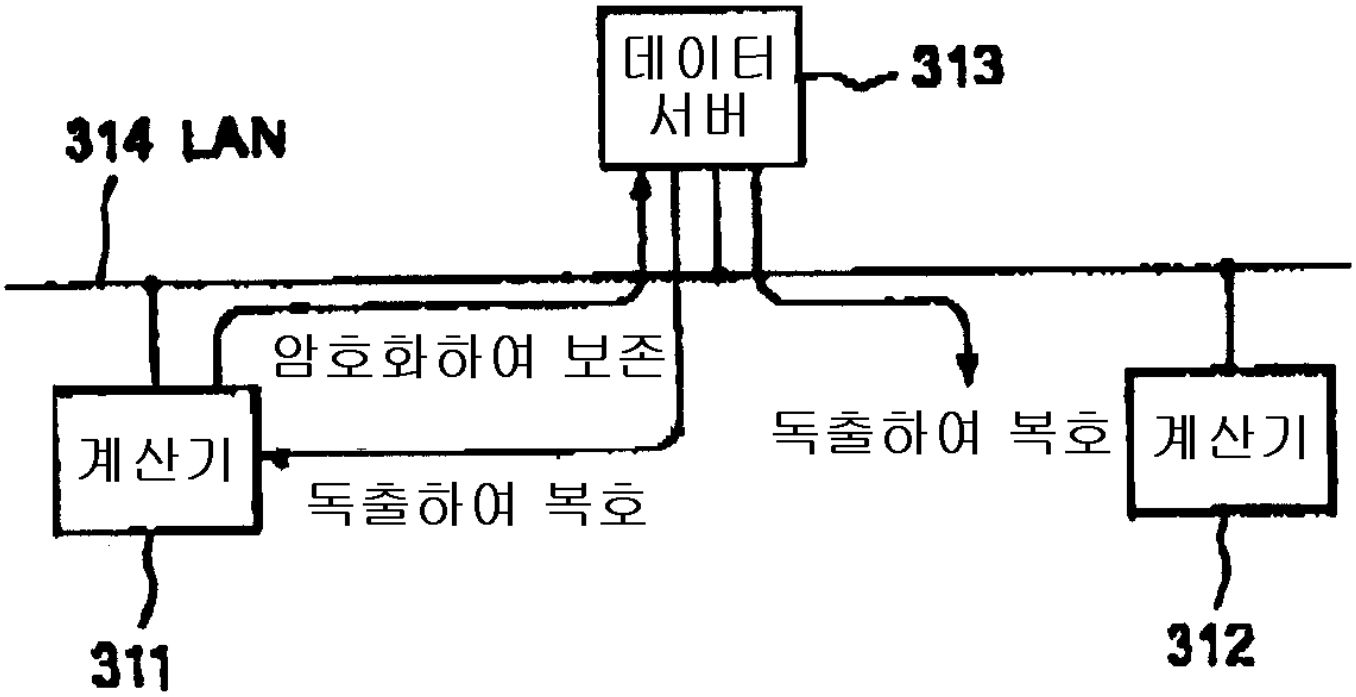
도면 44


128
64
64
32
32
F
S2
S3
const (r)
64
64

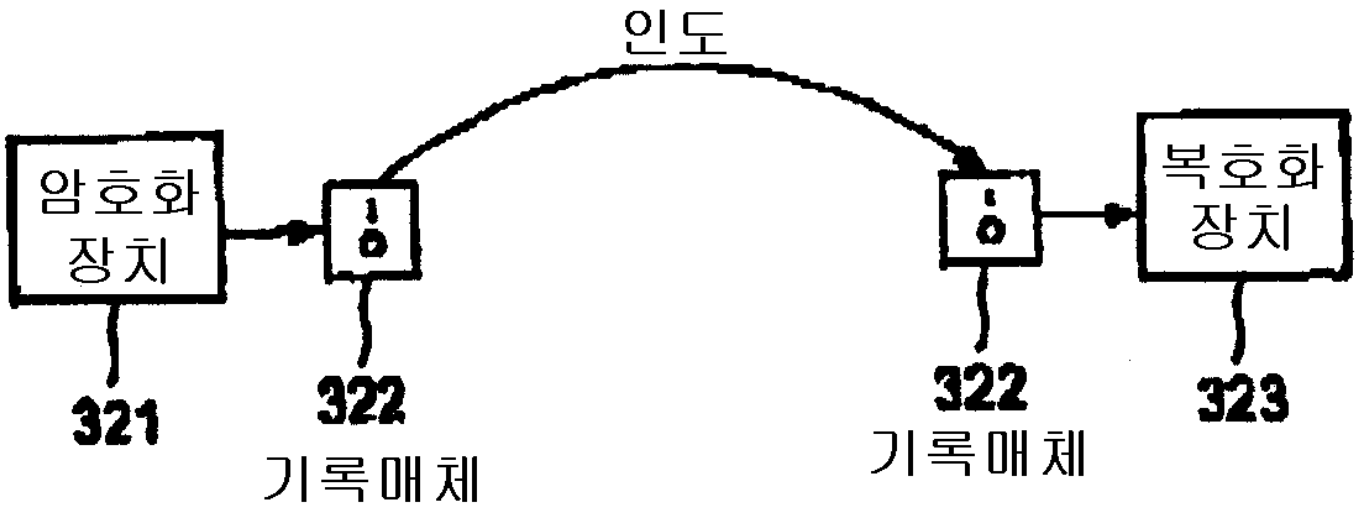
도면 45


키공유
평문
송신측
암호문
암호문
수신측
평문
통신네트워크
301
302
303

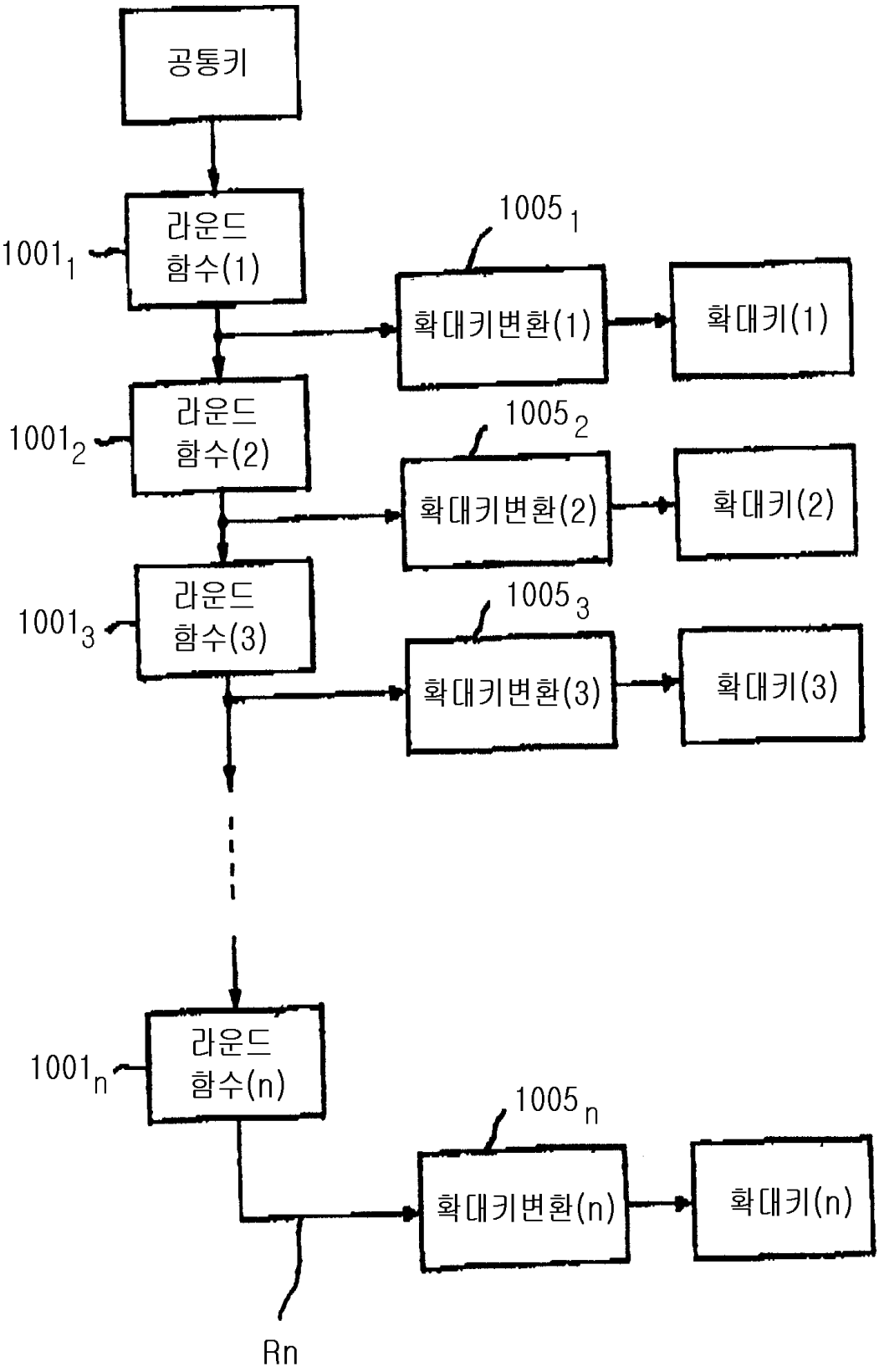
도면 46


데이터
서버
313
314 LAN
암호화하여 보존
계산기
독출하여 복호
독출하여 복호
계산기
311
312

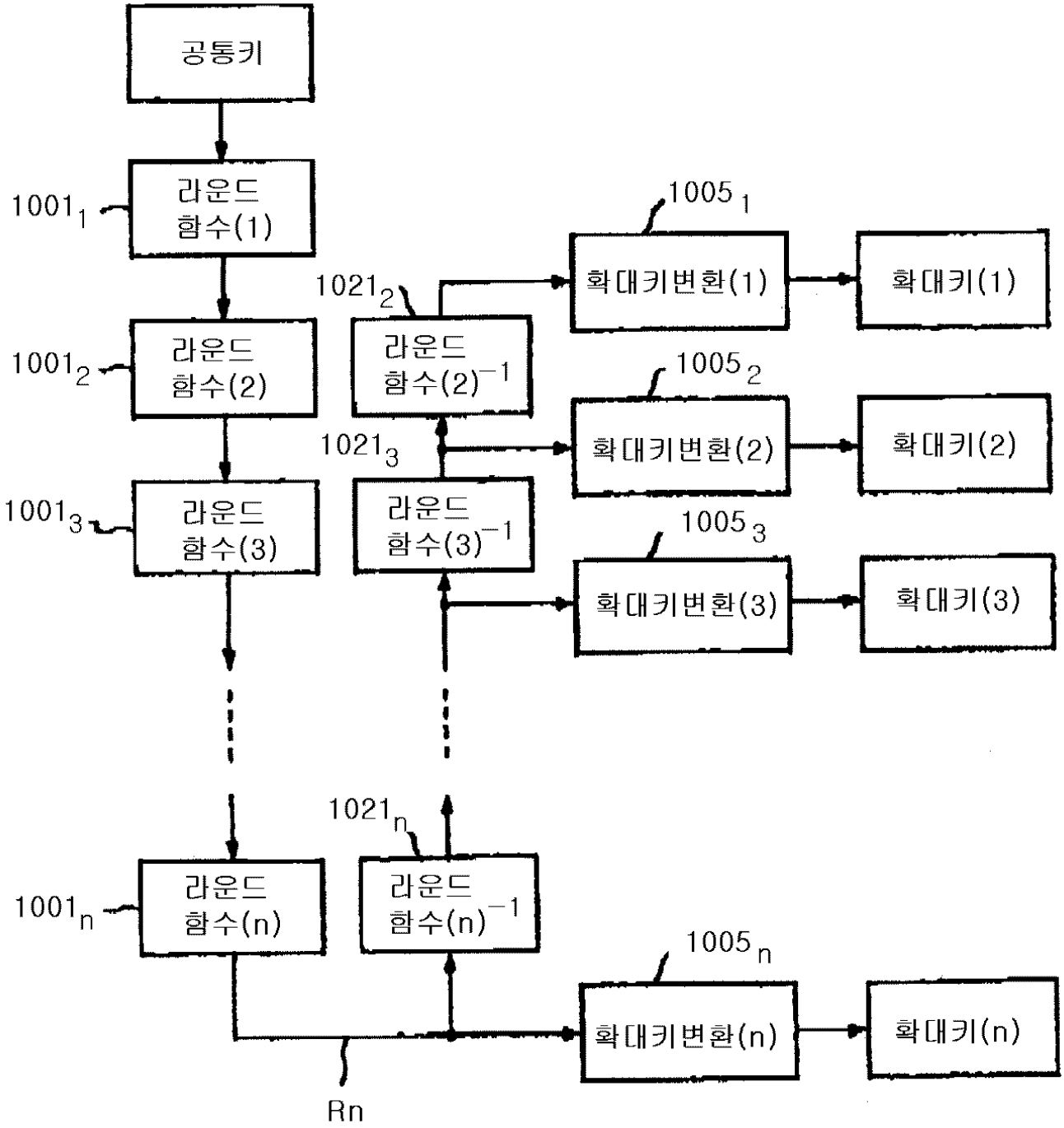
도면 47


인도
암호화
장치
복호화
장치
321
322
기록매체
322
기록매체
323

도면 48


공통키
라운드 함수(1)
1001 1
1005 1
확대키변환(1)
확대키(1)
라운드 함수(2)
1001 2
1005 2
확대키변환(2)
확대키(2)
라운드 함수(3)
1001 3
1005 3
확대키변환(3)
확대키(3)
라운드 함수(n)
1001 n
1005 n
확대키변환(n)
확대키(n)
Rn

도면 49


공통키
라운드 함수(1)
1001 1
라운드 함수(2)
1001 2
라운드 함수(3)
1001 3
라운드 함수(n)
1001 n
라운드 함수(2)-1
1021 2
라운드 함수(3)-1
1021 3
라운드 함수(n)-1
1021 n
Rn
확대키변환(1)
1005 1
확대키변환(2)
1005 2
확대키변환(3)
1005 3
확대키변환(n)
1005 n
확대키(1)
확대키(2)
확대키(3)
확대키(n)