

# 제 5 장

## 블록 암호 모드



**박 종 혁 교수**

Tel: 970-6702

Email: [jhpark1@seoultech.ac.kr](mailto:jhpark1@seoultech.ac.kr)

**1절 블록 암호 모드**

**2절 ECB 모드**

**3절 CBC 모드**

**4절 CFB 모드**

**5절 OFB 모드**

**6절 CTR 모드**

**7절 모드 선택**

# 제1절 블록 암호 모드

**1.1 블록 암호와 스트림 암호**

**1.2 모드란?**

**1.3 평문 블록과 암호문 블록**

**1.4 적극적인 공격자 메모리**

# 1.1 블록 암호와 스트림 암호

- 블록 암호(block cipher)
  - 어느 특정 비트 수의 「집합」을 한 번에 처리하는 암호 알고리즘이 「집합」을 블록(block)
  - 블록의 비트 수를 블록 길이(block length)
    - DES나 트리플 DES의 블록 길이는 64비트
    - DES: 64비트 평문, 64비트 암호문
    - AES: 블록 길이는 128비트, 192비트, 256비트

## 1.1 블록 암호와 스트림 암호

- 스트림 암호(stream cipher)는 데이터의 흐름(스트림)을 순차적으로 처리해가는 암호 알고리즘
- 1비트, 8비트, 혹은 32비트 등의 단위로 암호화와 복호화

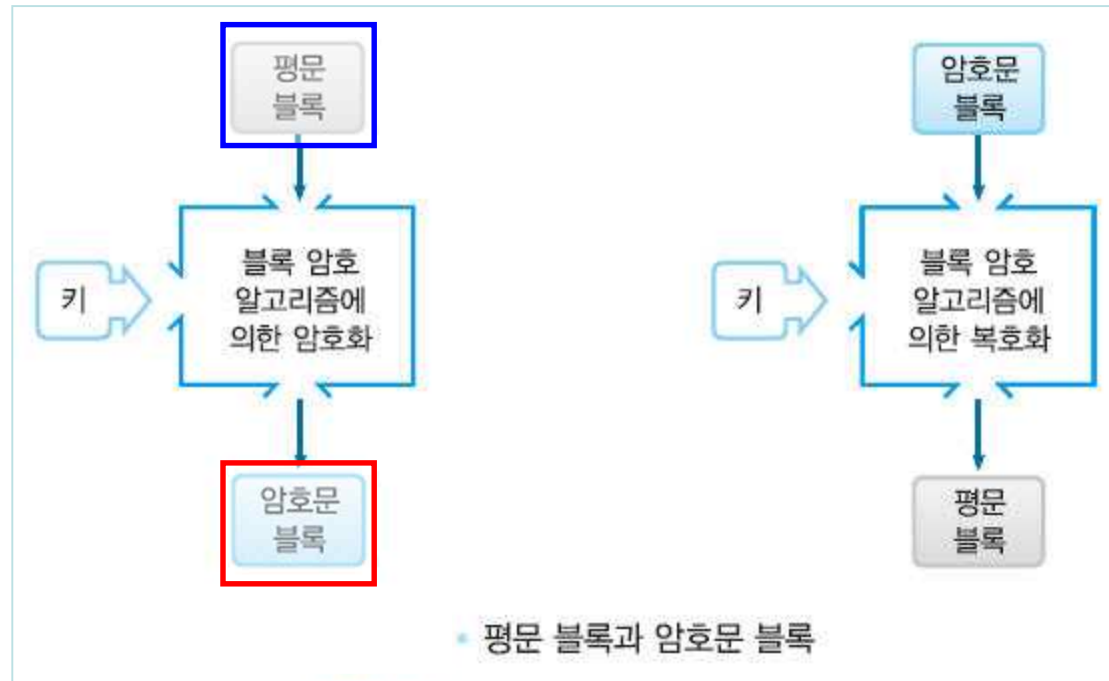
## 1.2 모드란?

- 모드란
  - 긴 평문을 블록으로 나누어 암호화
  - 각 블록에 암호 알고리즘을 반복해서 사용하여 긴 평문 전체를 암호화

## 블록암호 주요 모드

- ECB 모드 : Electric CodeBook mode(전자 부호표 모드)
- CBC 모드 : Cipher Block Chaining mode(암호 블록 연쇄 모드)
- CFB 모드 : Cipher-FeedBack mode(암호 피드백 모드)
- OFB 모드 : Output-FeedBack mode(출력 피드백 모드)
- CTR 모드 : CounTeR mode(카운터 모드)

## 1.3 평문 블록과 암호문 블록



- 평문 블록: 블록암호 알고리즘에서 암호화 대상이 되는 평문
- 암호문 블록: 블록암호 알고리즘을 써서 평문 블록을 암호화한 암호문



## 1.4 적극적인 공격자 멜로리

- 도청
- 위장
- 변조
- 공격자: 멜로리(Mallory)

## 제2절 ECB 모드

### 2.1 ECB 모드란?

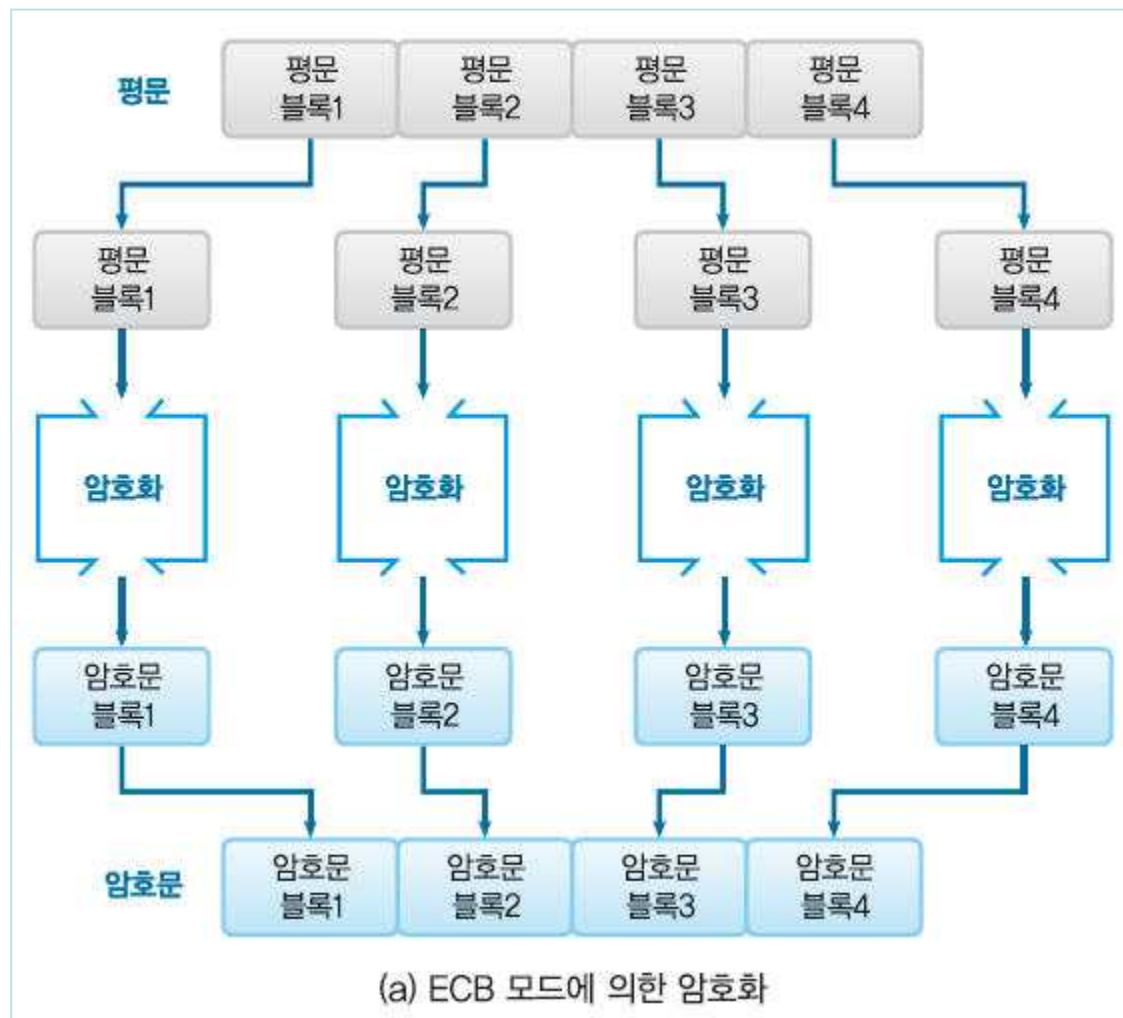
### 2.2 ECB 모드의 특징

### 2.3 ECB 모드에 대한 공격

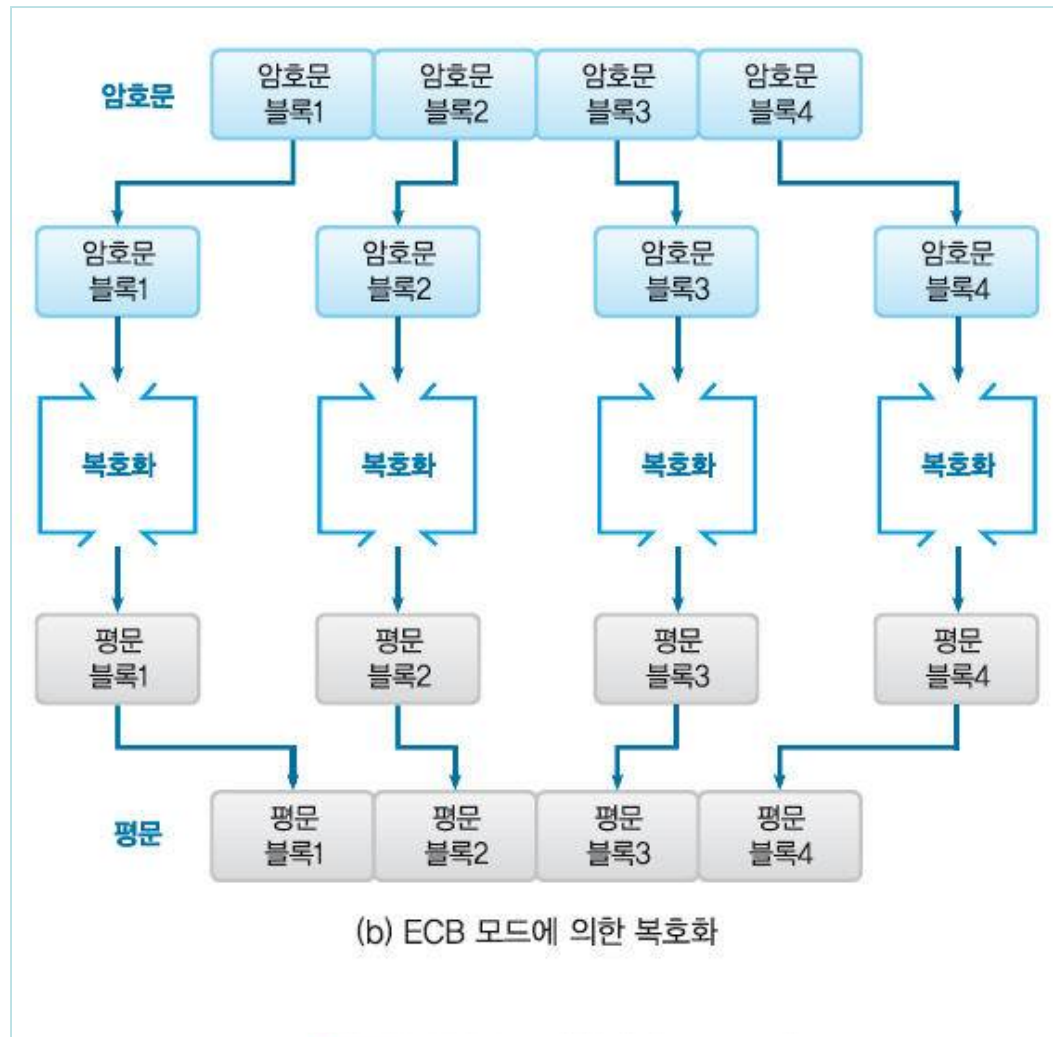
## 2.1 ECB 모드란?

- 평문 블록을 암호화한 것이 그대로 암호문 블록
- 패딩(padding)
  - 마지막 평문 블록이 블록 길이에 미치지 못할 경우에 추가하여 블록 길이가 되도록 맞춘다
- 안전하지 않다

# ECB 모드에 의한 암호화



# ECB 모드에 의한 복호화



## 2.2 ECB 모드의 특징

- 가장 기밀성이 낮은 모드
- 암호문을 살펴보는 것만으로도 평문 속에 패턴 반복성 감지
- 안전하지 않다

## 2.3 ECB 모드에 대한 공격

- 어느 은행의 송금 의뢰 데이터가 다음 3개의 블록으로 구성

블록1 = 송금자의 은행계좌번호

블록2 = 송금처의 은행계좌번호

블록3 = 송금액

- 송금 의뢰 데이터를 받은 은행은 지정된 금액을 송금자로부터 송금처의 계좌로 이동
- A-5374의 계좌로부터 B-6671의 계좌로 1억 원을 송금하라는 송금 의뢰 데이터를 만들어보자

## 2.3 ECB 모드에 대한 공격

- A-5374의 계좌로부터 B-6671의 계좌로 1억 원을 송금하라는 송금 의뢰 데이터를 만들어보자

평문 블록1 = 41 2D 35 33 37 34 20 20 20 20 20 20 20 20 20  
(송금자:A-5374)

평문 블록2 = 42 2D 36 36 37 31 20 20 20 20 20 20 20 20 20  
(송금처:B-6671)

평문 블록3 = 31 30 30 30 30 30 30 30 30 20 20 20 20 20 20  
(송금액:100000000)

- 암호화 하자

암호문 블록1 = 59 7D DE CC EF EC BA 9B BF 83 99 CF 60 D2 59 B9  
(송금자:????)

암호문 블록2 = DF 49 2A 1C 14 8E 18 B6 53 1F 38 BD 5A A9 D7 D7  
(송금처:????)

암호문 블록3 = CD AF D5 9E 39 FE FD 6D 64 8B CC CB 52 56 8D 79  
(송금액:????)



## 2.3 ECB 모드에 대한 공격

- 공격자 맬로리가 암호문 블록의 1과 2의 내용을 바꾼다

암호문 블록1 = DF 49 2A 1C 14 8E 18 B6 53 1F 38 BD 5A A9 D7 D7

(송금자:????)

암호문 블록2 = 59 7D DE CC EF EC BA 9B BF 83 99 CF 60 D2 59 B9

(송금처:????)

암호문 블록3 = CD AF D5 9E 39 FE FD 6D 64 8B CC CB 52 56 8D 79

(송금액:????)

- 은행이 이것을 복호화하면 다음과 같이 된다

평문 블록1 = 42 2D 36 36 37 31 20 20 20 20 20 20 20 20 20

(송금처:B-6671)

평문 블록2 = 41 2D 35 33 37 34 20 20 20 20 20 20 20 20 20

(송금자:A-5374)

평문 블록3 = 31 30 30 30 30 30 30 30 30 20 20 20 20 20 20

(송금액:100000000)

## 2.3 ECB 모드에 대한 공격

- 원래는 A-5374의 계좌에서 B-6671의 계좌로 1억 원을 송금하라는 지시였는데 B-6671의 계좌에서 A-5374의 계좌로 1억 원을 송금하라는 정반대의 지시가 되어 버렸다

## 제3절 CBC모드

**3.1 CBC 모드란?**

**3.2 초기화 벡터**

**3.3 CBC 모드의 특징**

**3.4 CBC 모드에 대한 공격**

**3.5 패딩 오라클 공격**

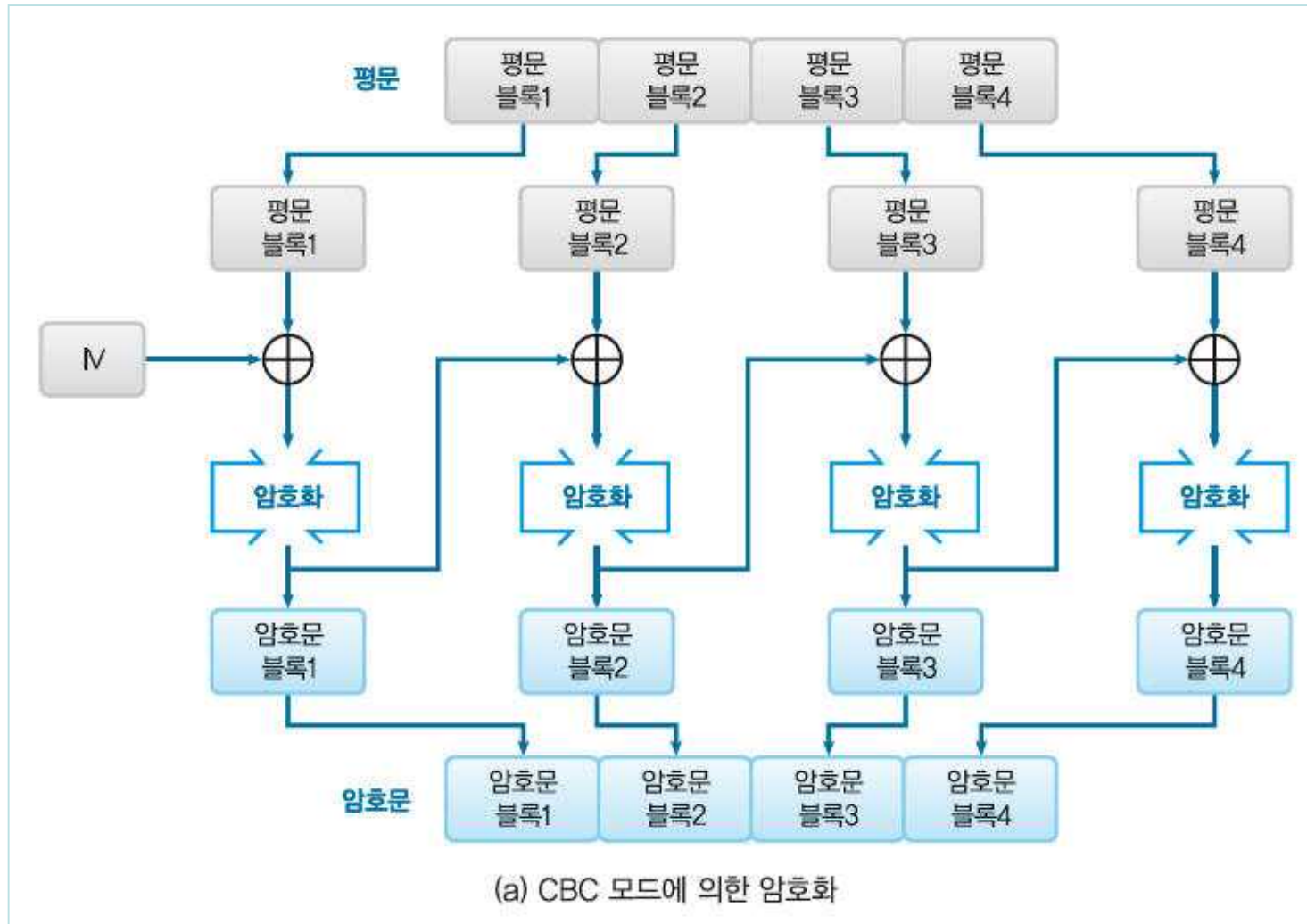
**3.6 초기화 벡터 공격**

**3.7 CBC 모드 활용의 예**

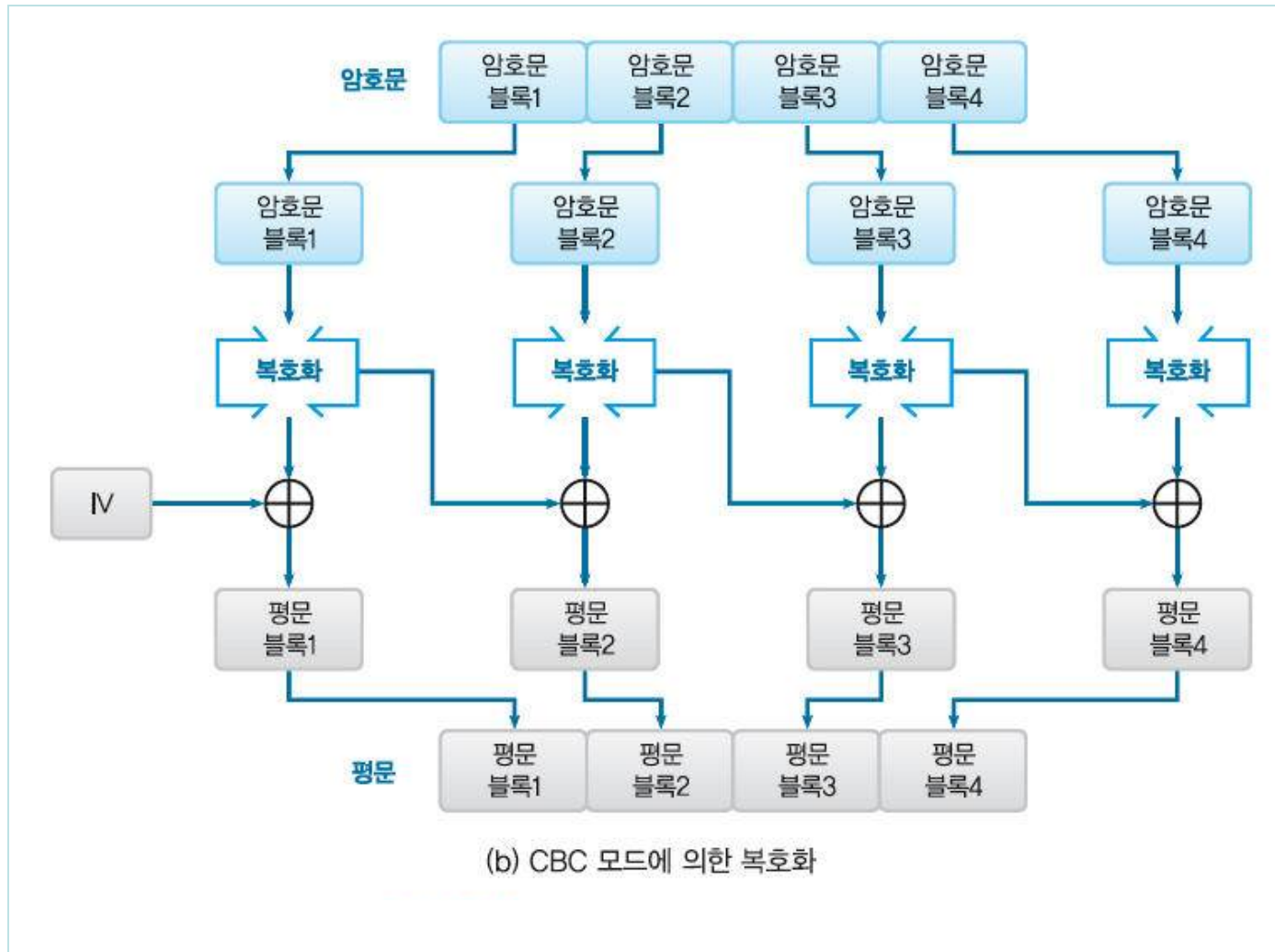
## 3.1 CBC 모드란?

- CBC 모드
  - Cipher Block Chaining 모드(암호 블록 연쇄 모드)의 약자이다. 암호문 블록을 마치 체인처럼 연결시키기 때문에 붙여진 이름
  - CBC 모드에서는 1 단계 앞에서 수행되어 결과로 출력된 암호문 블록에 평문 블록을 XOR 하고 나서 암호화를 수행
  - 각각의 암호문 블록은 단지 현재 평문블록 뿐만 아니라 그 이전의 평문 블록들의 영향도 받게 된다

# CBC 모드에 의한 암호화



# CBC 모드에 의한 복호화



## 3.2 초기화 벡터

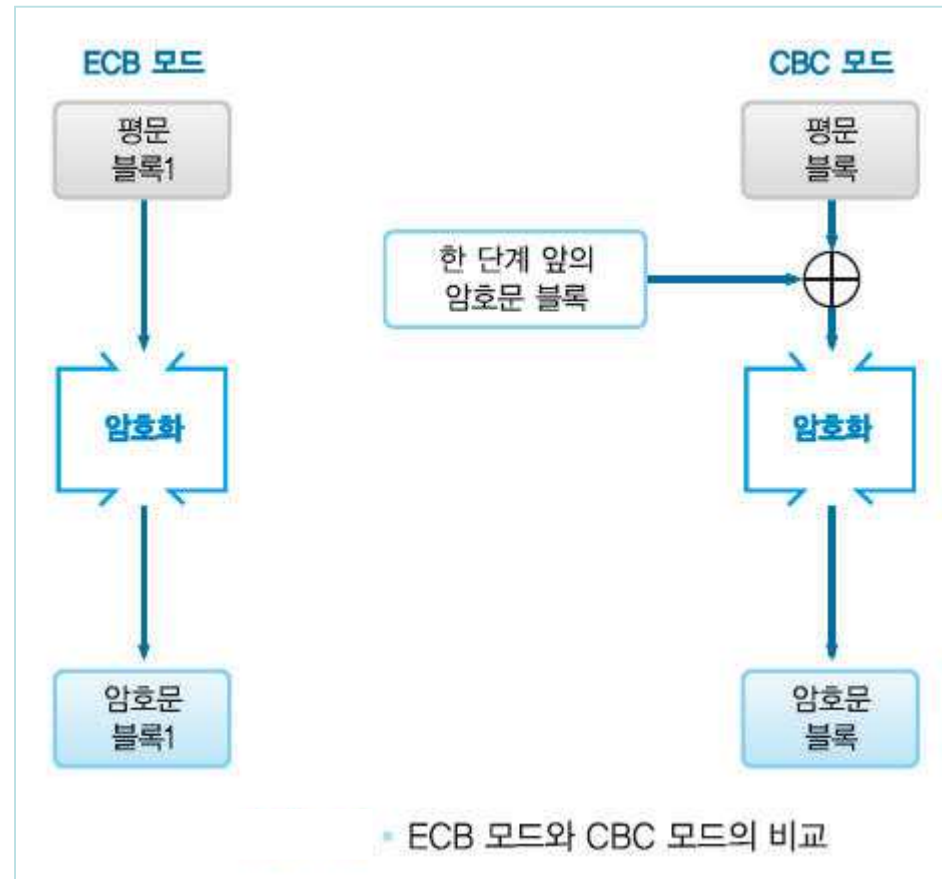
- 초기화 벡터(initialization vector)
  - 최초의 평문 블록을 암호화할 때는 「1 단계 앞의 암호문 블록」이 존재하지 않으므로 「1 단계 앞의 암호문 블록」을 대신할 비트열인 한 개의 블록을 준비할 필요

## 3.3 CBC 모드의 특징

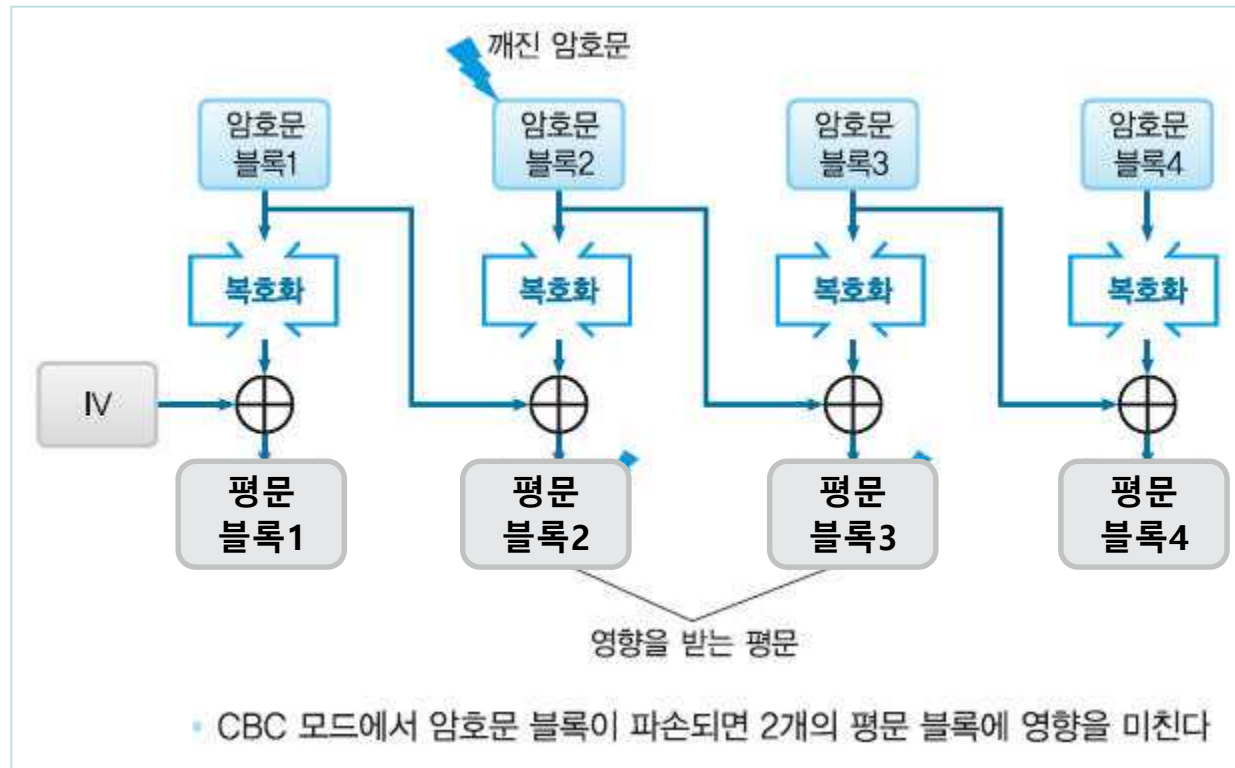
- 평문 블록은 반드시 「1 단계 앞의 암호문 블록」 과 XOR을 취하고 나서 암호화
  - 따라서 만약 평문 블록1과 2의 값이 같은 경우라도 암호문 블록1과 2의 값이 같아진다고는 할 수 없고, ECB 모드가 갖고 있는 결점이 CBC 모드에는 없다.
- 암호문 블록3을 만들고 싶다면 적어도 평문 블록의 1, 2, 3까지가 갖추어져 있어야만 한다



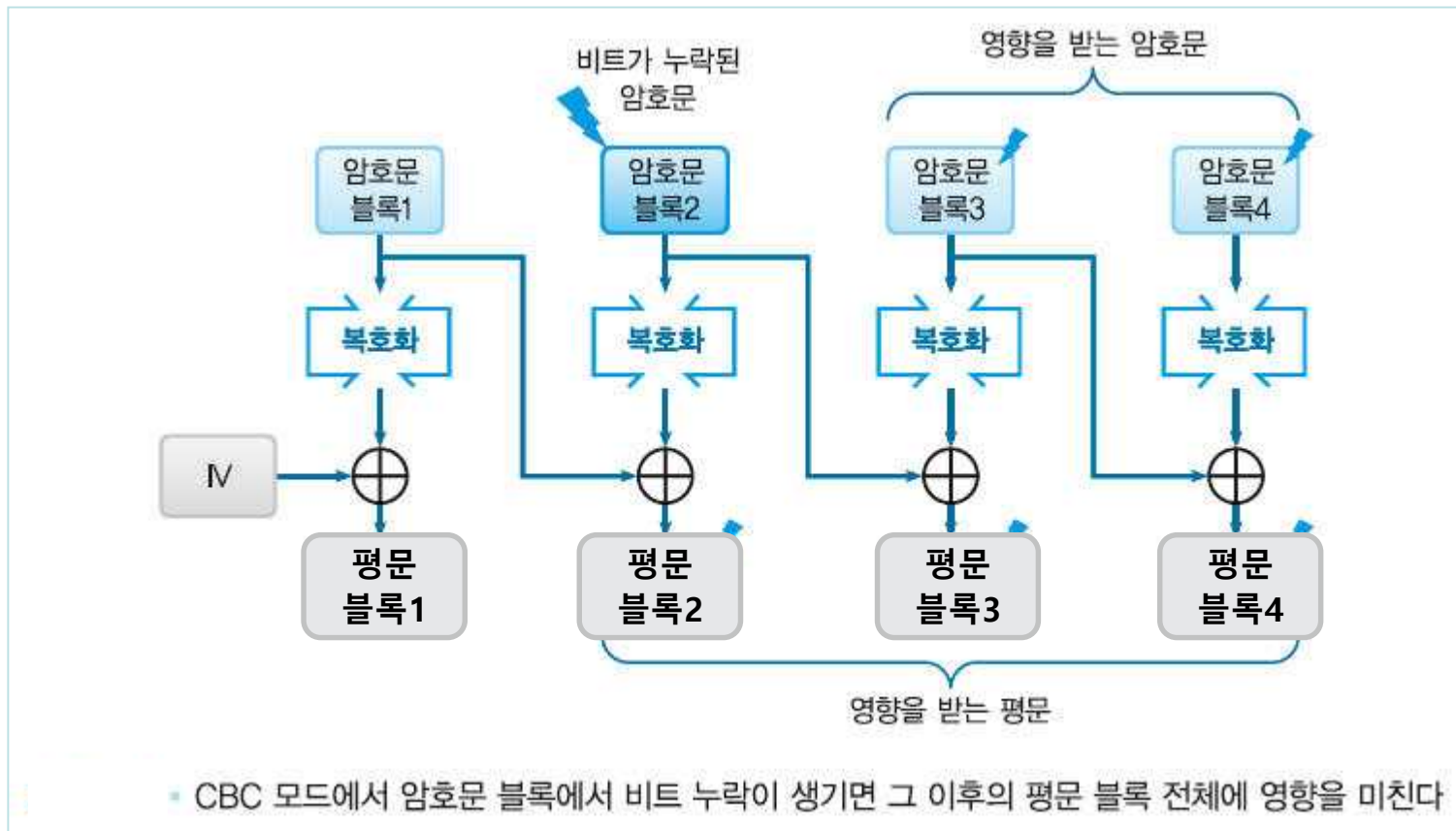
# ECB 모드와 CBC 모드



# 깨진 암호문

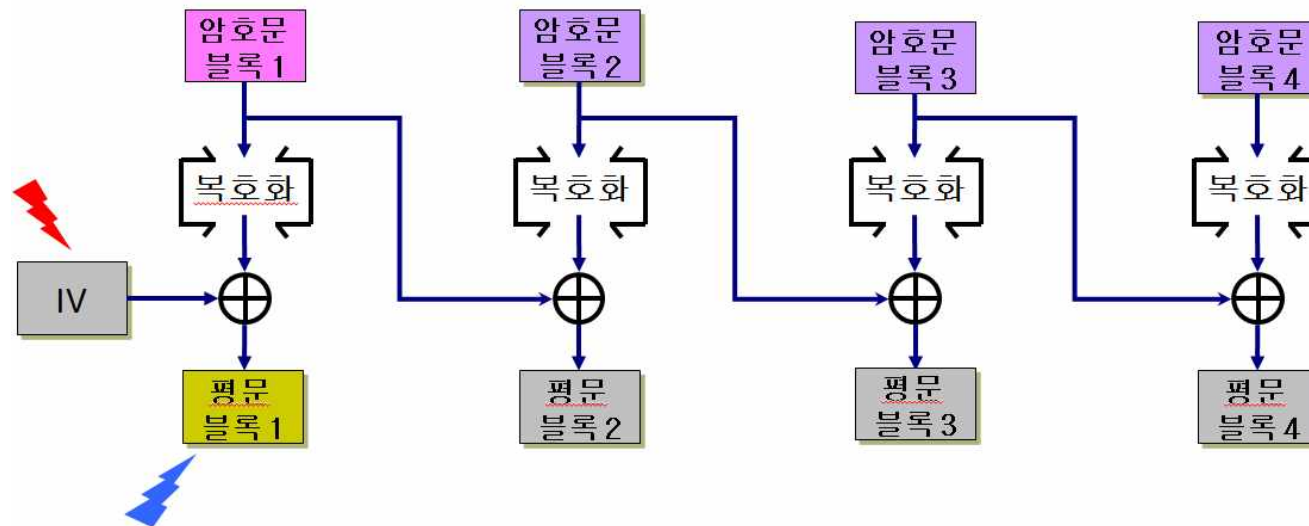


# 암호문 블록에서 비트 누락



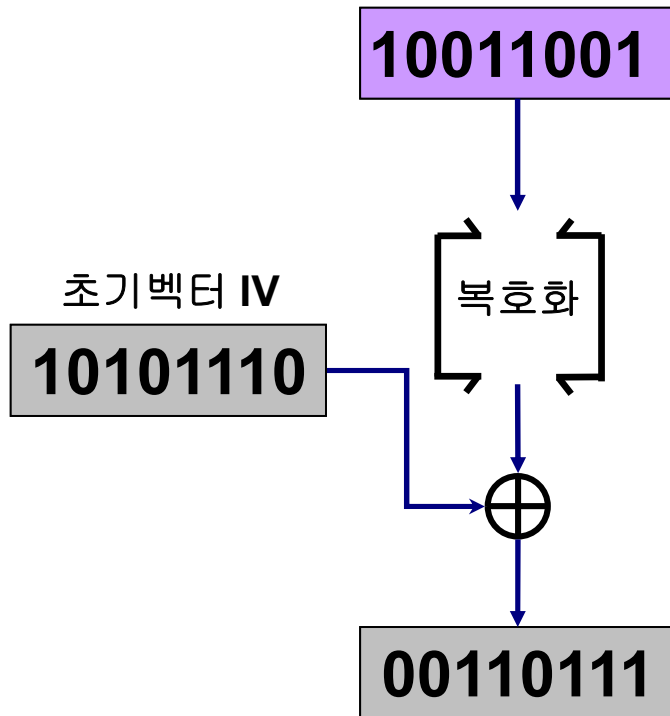
## 3.4 CBC 모드에 대한 공격(초기화 벡터의 비트 반전)

- 초기화 벡터의 비트를 반전시켜 평문 블록의 비트를 반전시키는 공격(CBC 모드)

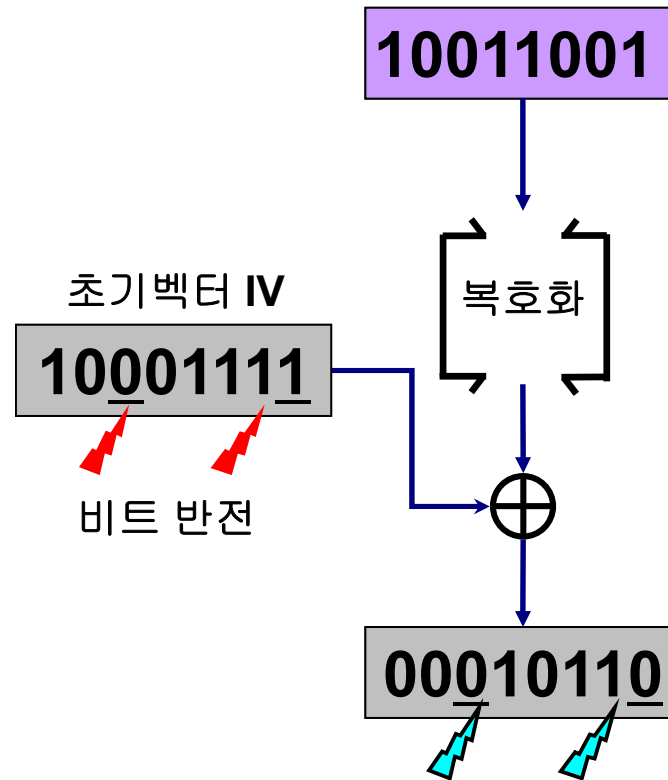


# CBC 모드에서 초기벡터의 비트반전에 대한 영향

초기벡터 IV에 비트 반전이 없을 경우



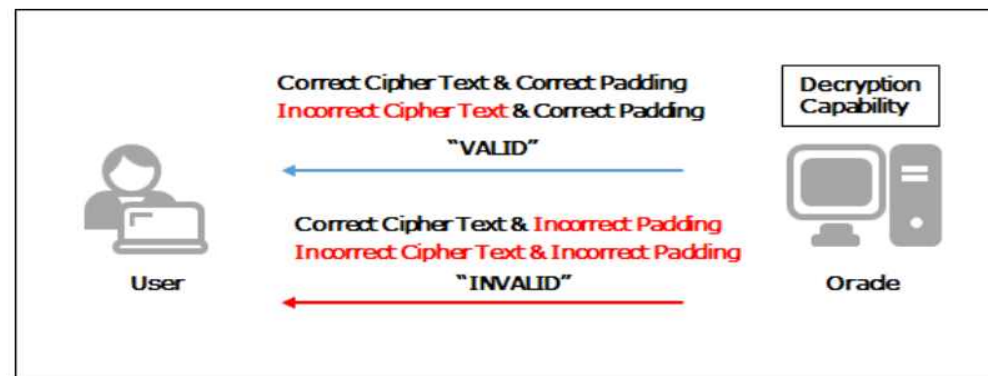
초기벡터 IV에 비트 반전이 없을 경우



영향을 받아 반전된 비트

## 3.5 패딩 오라클 공격

- 블록 암호의 패딩을 이용한 공격
- 패딩 내용을 조금씩 변화시켜 암호문을 여러 차례 송신
- 수신자가 올바르게 복호화 하지 못할 경우 오류를 관찰하여 평문 정보를 취득
- 패딩을 사용하는 모든 모드에 적용
- 공격자가 메시지의 패딩이 옳은지 아닌지의 여부를 판단하는 오라클이 있다면, 이 오라클을 이용하여 암호문에 대응하는 메시지를 알아낼 수 있다



## 3.6 초기화 벡터(IV) 공격

- 초기화 벡터는 난수(Random Number)로 부여
- SSL/TLS의 TLS 버전 1.0
  - 초기화 벡터를 이전 CBC 모드로 암호화한 마지막 블록을 사용
  - 문제점 발견 후 TLS 버전 1.1 부터는 초기화 벡터를 명시적으로 부여

## 3.7 CBC 모드 활용 예

- IPsec에는 통신의 기밀성을 지키기 위해 CBC 모드를 사용함
  - 예를 들면 트리플 DES를 CBC 모드로 사용한 3DES-CBC나, AES를 CBC 모드로 사용한 AES-CBC 등이 여기에 해당됨
- 인증을 수행하는 대칭암호 시스템의 하나인 Kerberos version 5에서도 사용하고 있음
- SSL/TLS:
  - 통신기밀성 보호
- 3DES-E0-CBC
- AES-256-CBC:
  - 키 길이가 256비트인 경우



## 제4절 CFB 모드

**4.1 CFB 모드란?**

**4.2 초기화 벡터**

**4.3 CFB 모드와 스트림 암호**

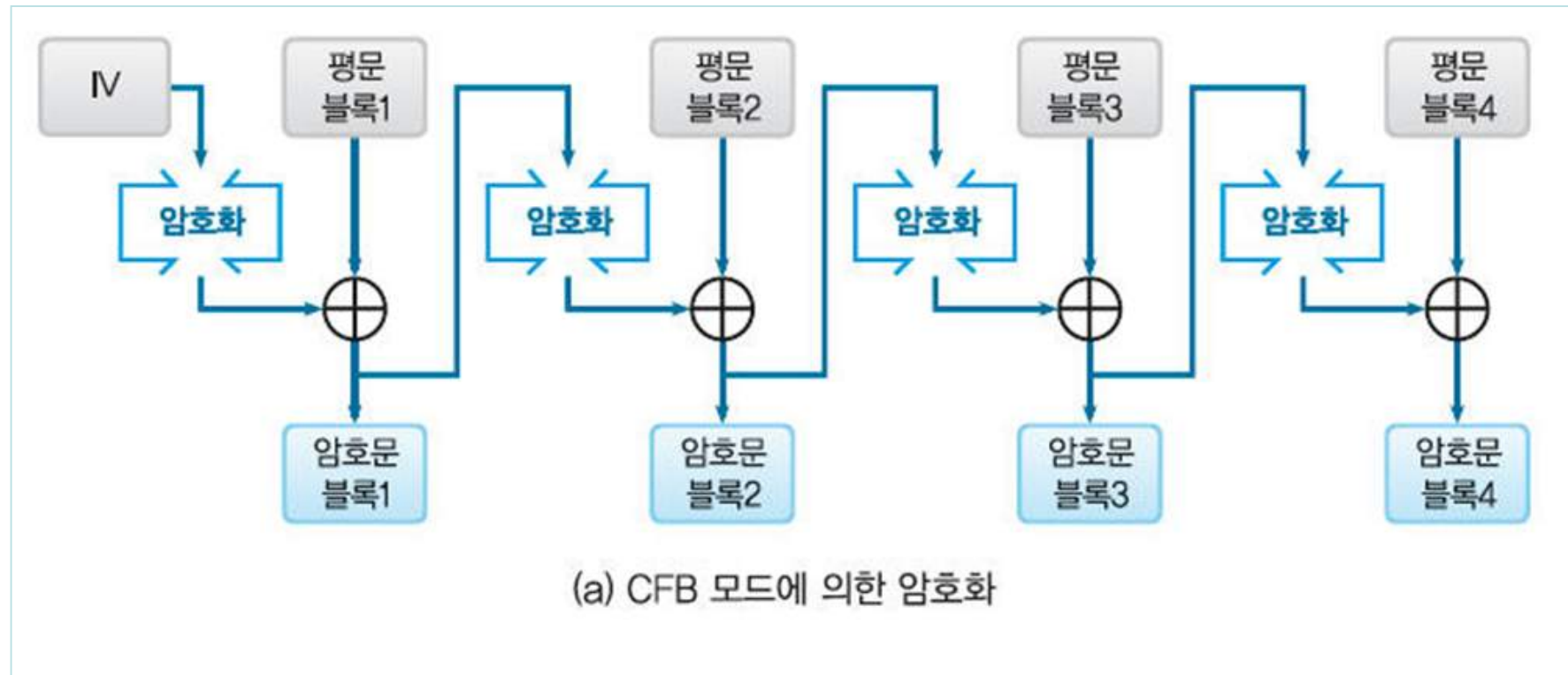
**4.4 CFB 모드의 복호화**

**4.5 CFB 모드에 대한 공격**

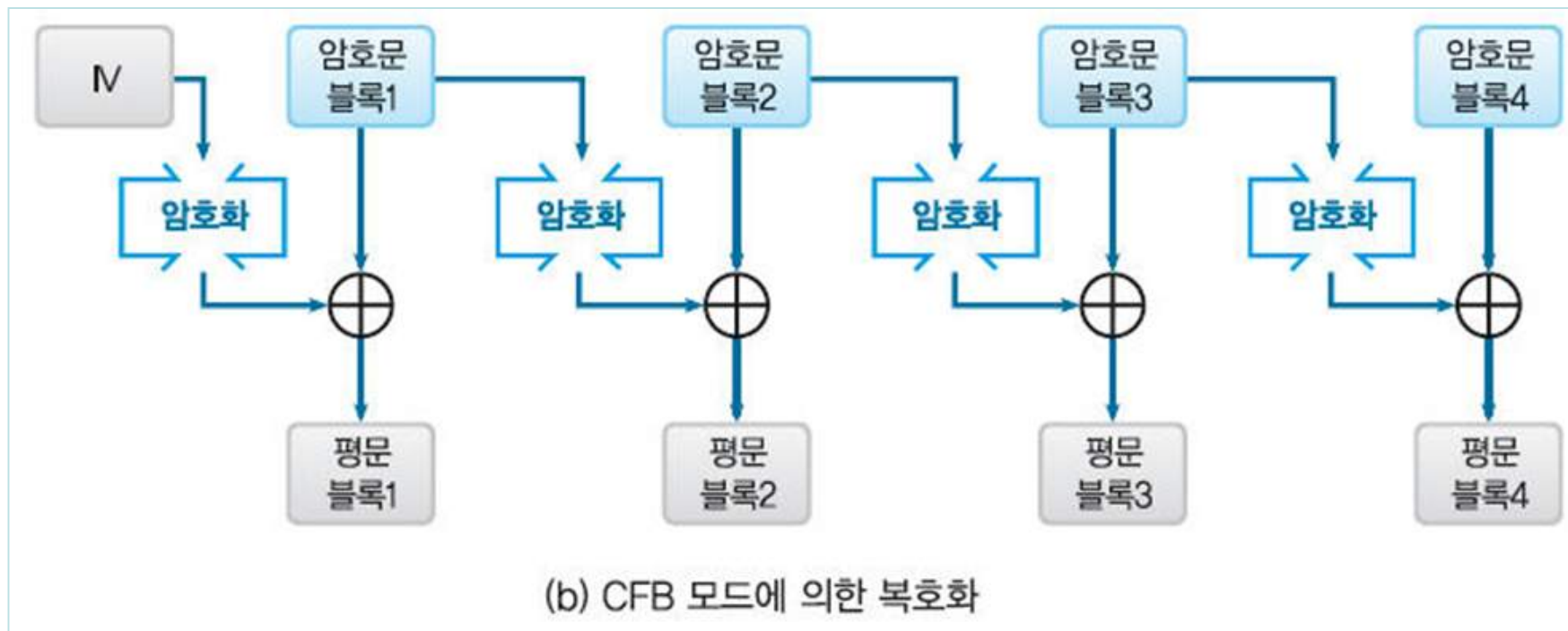
## 4.1 CFB 모드

- CFB 모드
  - Cipher FeedBack 모드(암호 피드백 모드)의 약자
  - CFB 모드에서는 1 단계 앞의 암호문 블록을 암호 알고리즘의 입력으로 사용

# CFB 모드의 암호화



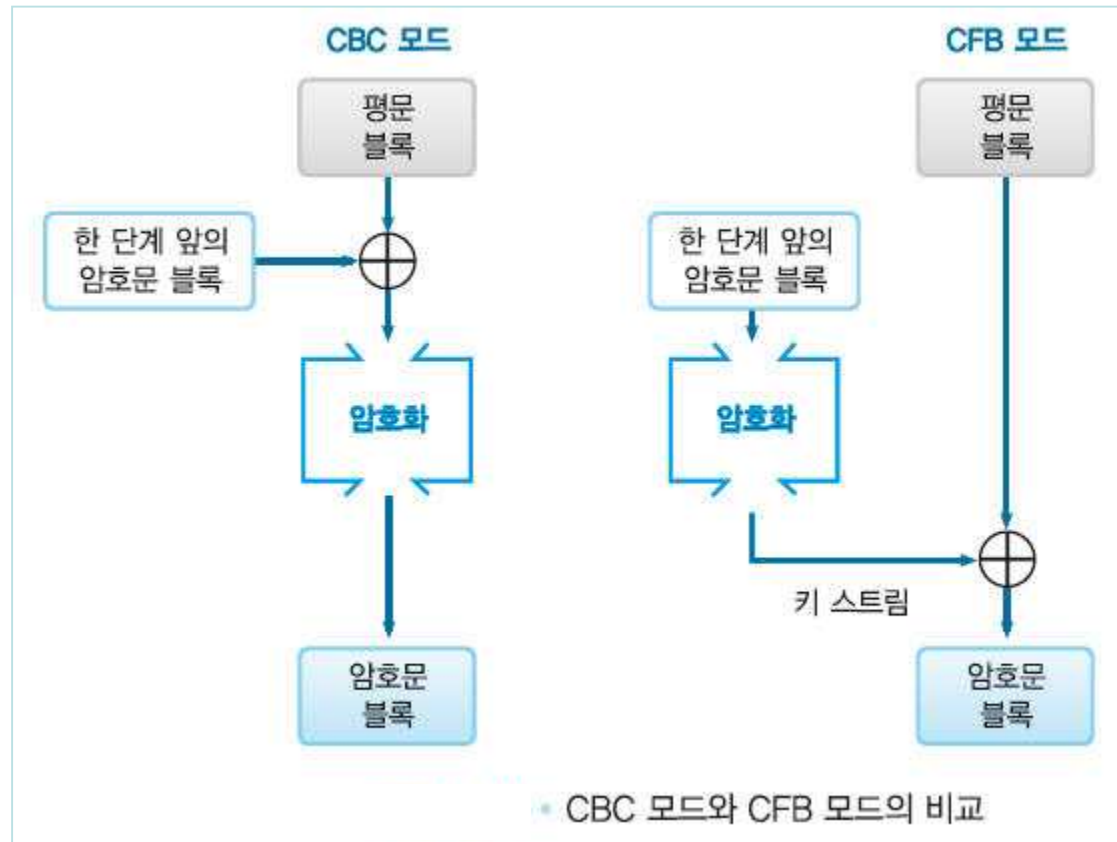
# CFB 모드의 복호화



## 4.2 초기화 벡터

- 초기화 벡터(IV)
  - 최초의 암호문 블록을 만들어낼 때는 1 단계 앞의 출력이 존재하지 않으므로 대신에 IV를 사용

# CBC 모드와 CFB 모드 비교



## 4.3 CFB 모드와 스트림 암호

- 키 스트림(key stream)
  - CFB 모드에서 암호 알고리즘이 생성하는 비트열
  - 키 스트림을 생성하기 위한 의사난수 생성기로서 암호 알고리즘을 이용
  - 초기화 벡터는 의사난수 생성기의 「seed(종자)」에 해당
- CFB 모드는 블록 암호를 써서 생성한 키를 이용하는 스트림 암호

## 4.4 CFB 모드의 복호화

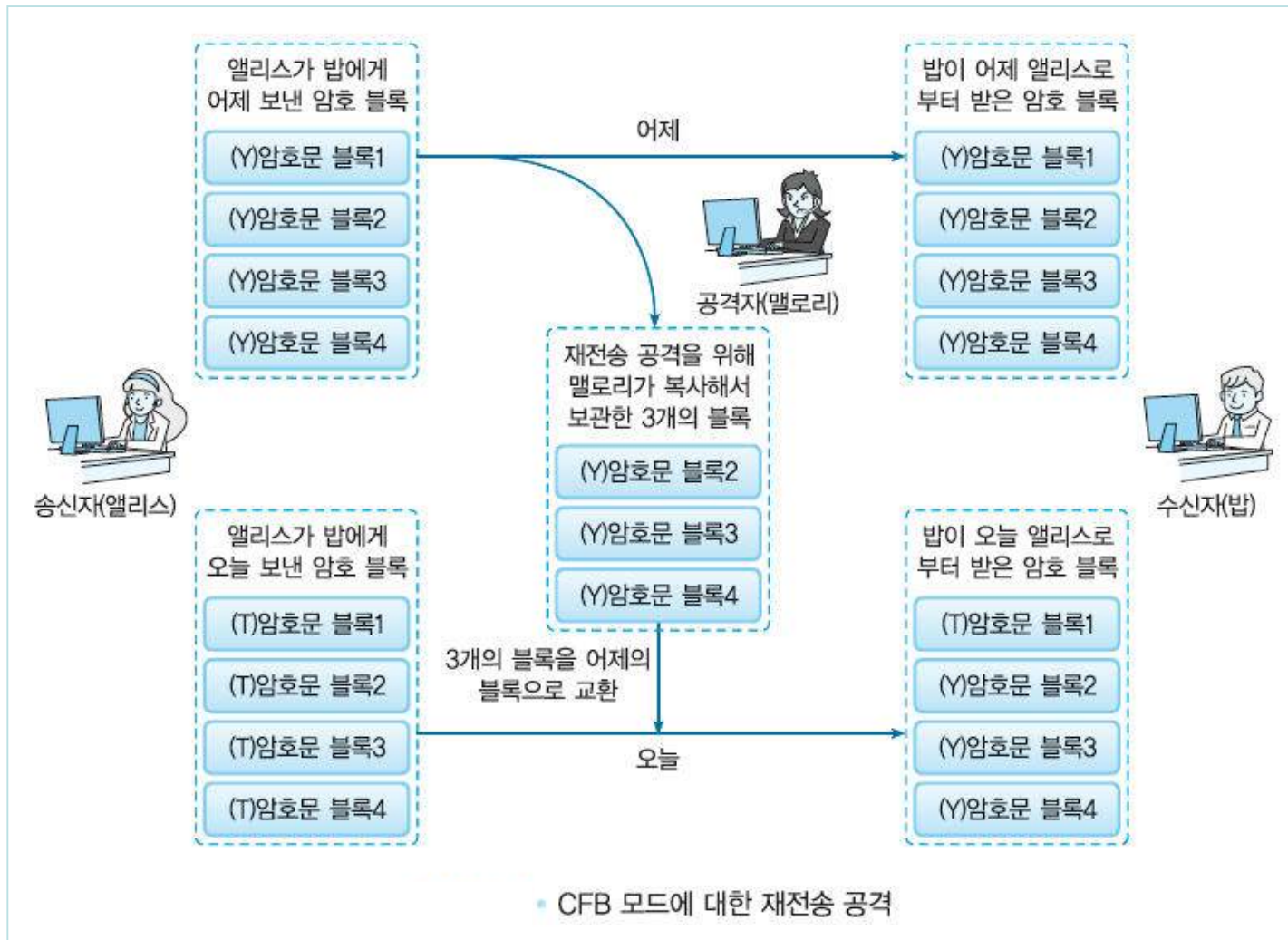
- CFB 모드에서 복호화를 수행할 경우, 블록 암호 알고리즘 자체는 암호화를 수행하고 있다는 것에 주의
- 키 스트림은 암호화에 의해 생성



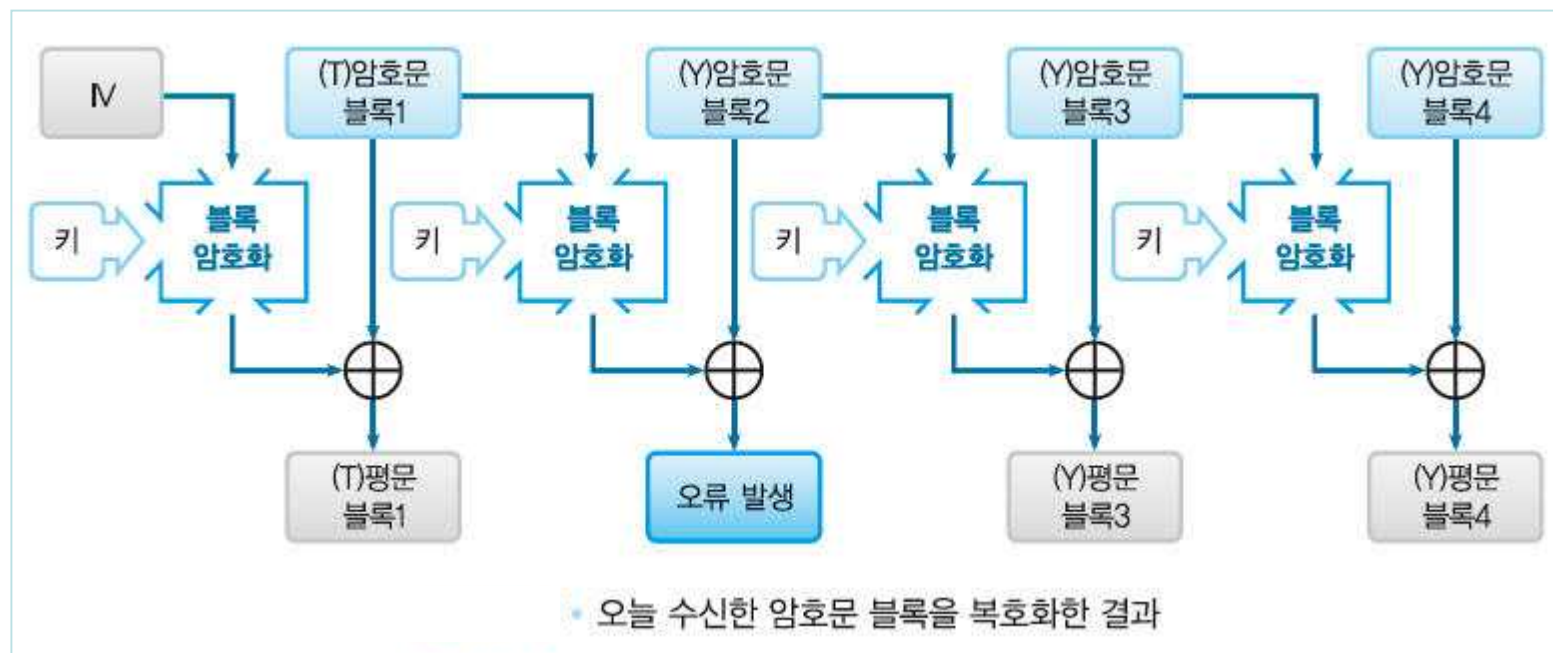
## 4.5 CFB 모드에 대한 공격

- 재전송 공격(replay attack)

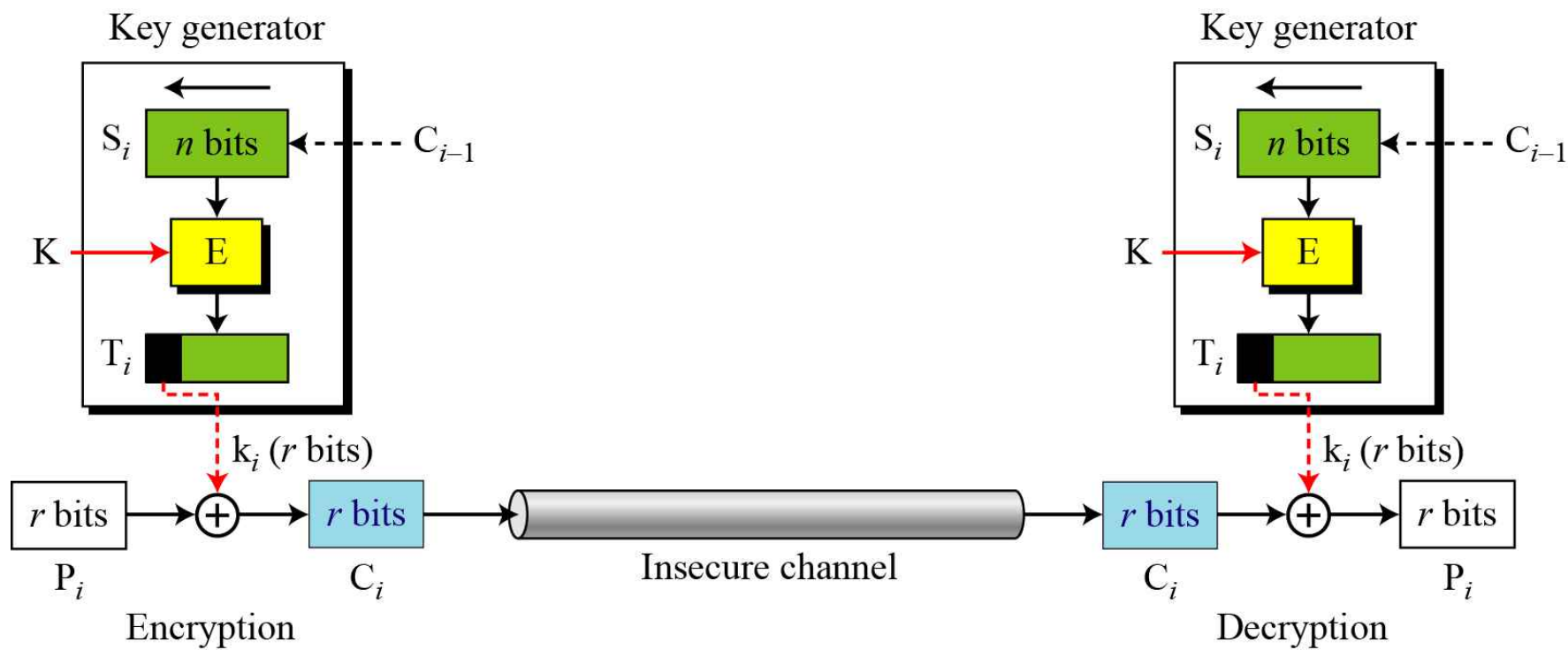
# 재전송 공격(replay attack)



# 오늘 수신한 암호문의 복호화



# 스트림 암호로서의 CFB 모드



## 제5절 OFB 모드

**5.1 OFB 모드란?**

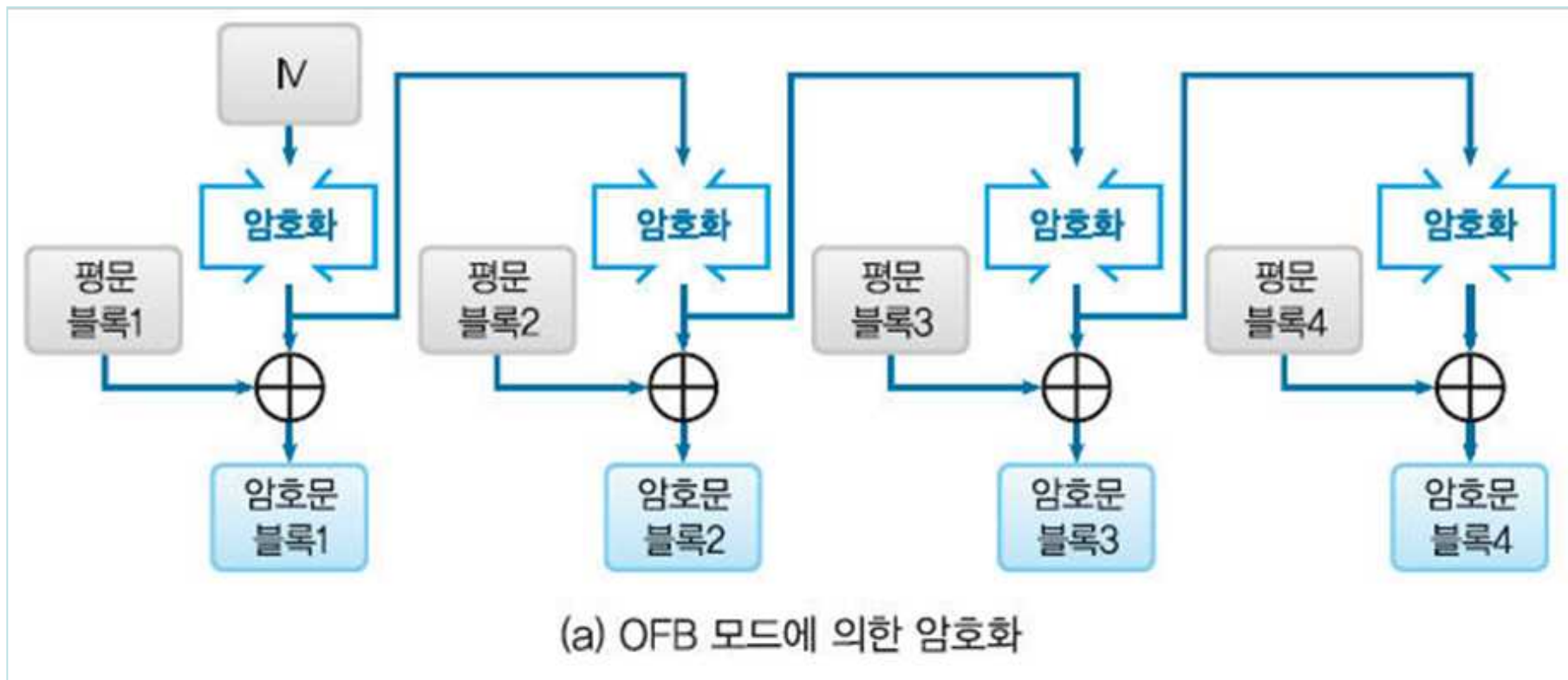
**5.2 초기화 벡터**

**5.3 CFB 모드와 OFB 모드의 비교**

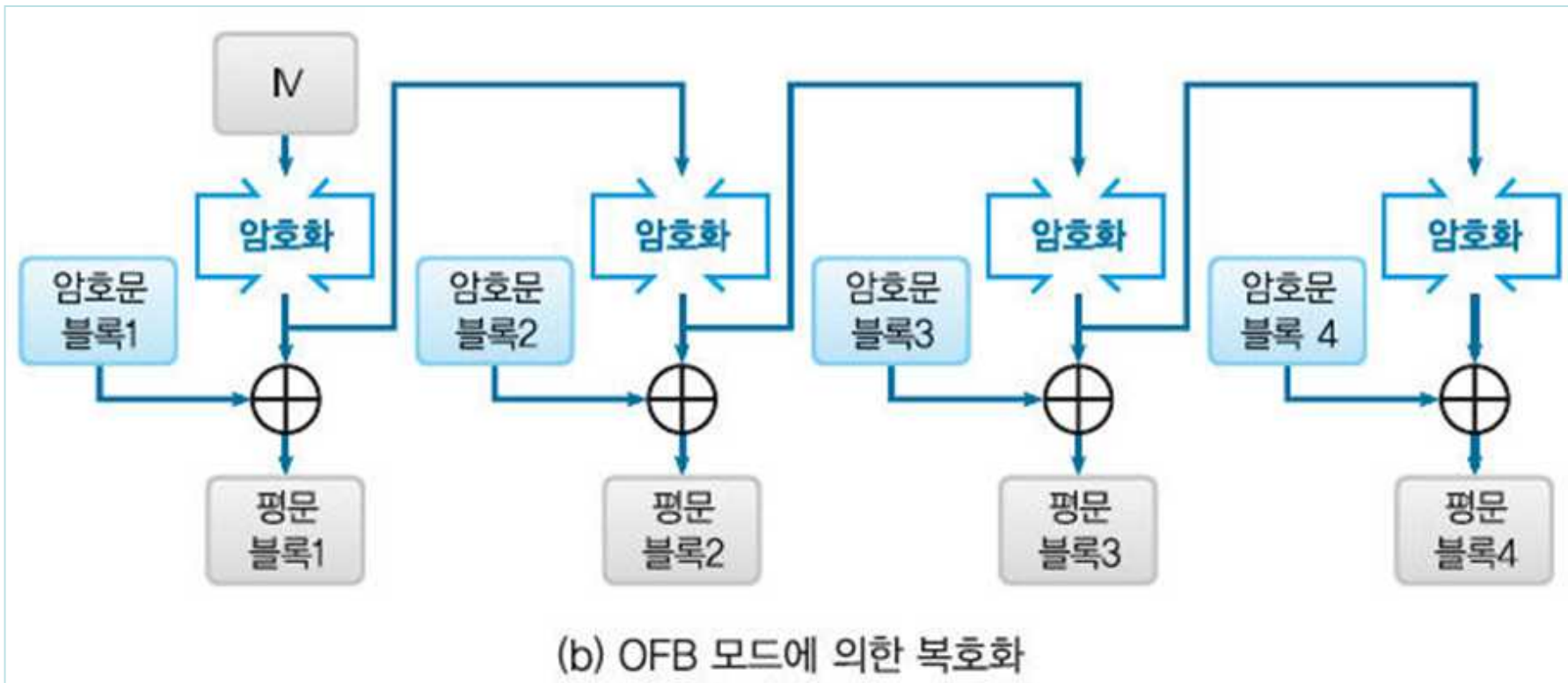
## 5.1 OFB 모드란?

- OFB 모드
  - Output-FeedBack 모드(출력 피드백 모드)의 약자
  - OFB 모드에서는 암호 알고리즘의 출력을 암호 알고리즘의
  - 평문 블록은 암호 알고리즘에 의해 직접 암호화되고 있는 것은 아  
님
  - 「평문 블록」과 「암호 알고리즘의 출력」을 XOR  
해서 「암호문 블록」을 만든다

# OFB 모드에 의한 암호화



# OFB 모드에 의한 복호화





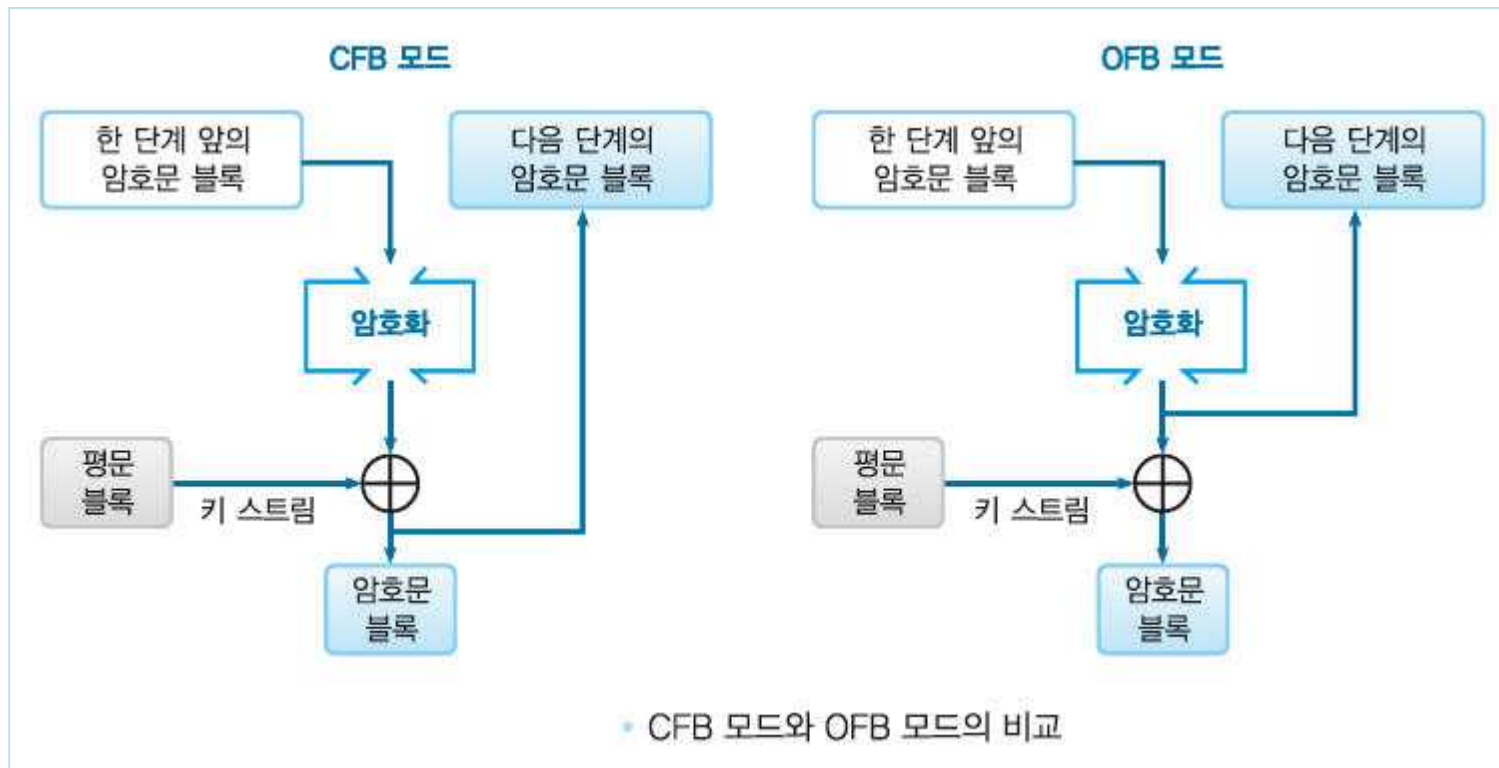
## 5.2 초기화 벡터

- CBC 모드나 CFB 모드와 마찬가지로 초기화 벡터(IV)를 사용
- 초기화 벡터는 암호화 때마다 다른 랜덤 비트열을 이용

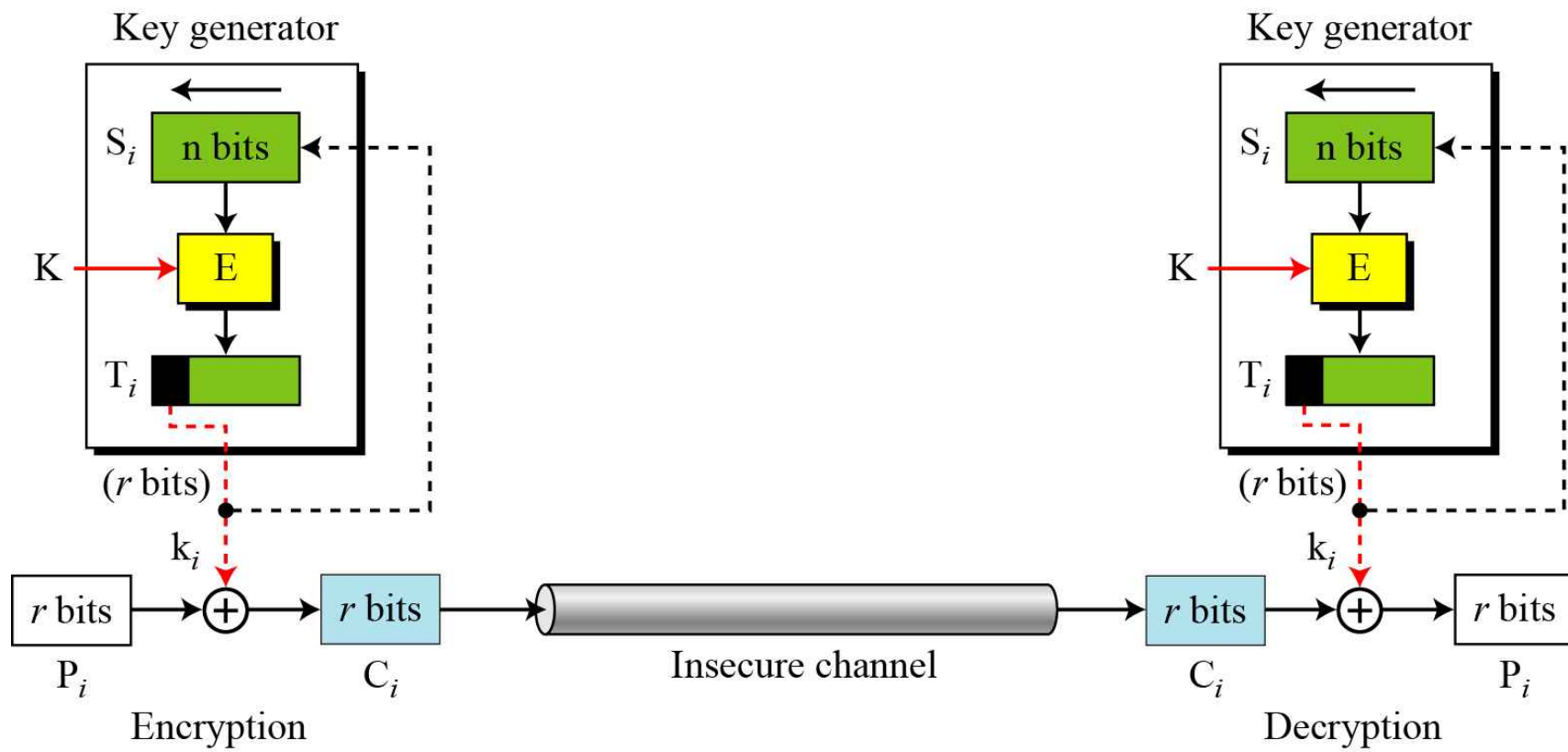
## 5.3 CFB 모드와 OFB 모드의 비교

- OFB 모드와 CFB 모드에서는 암호 알고리즘으로의 입력만 다르다

# CFB 모드와 OFB 모드 비교



# 스트림 암호로서의 OFB 모드



## 제6절 CTR 모드

### 6.1 카운터 만드는 법

### 6.2 OFB 모드와 CTR 모드의 비교

### 6.3 CTR 모드의 특징

### 6.4 오류와 기밀성

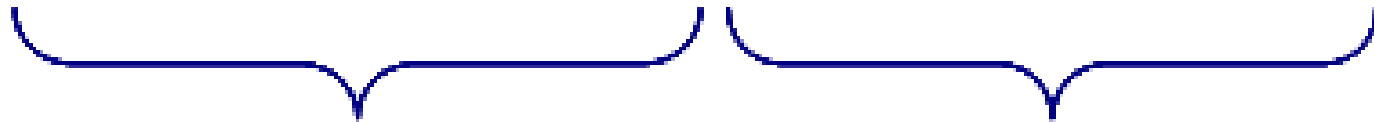
# CTR 모드

- CounTeR 모드의 약자
- CTR 모드는 1씩 증가해 가는 카운터를 암호화해서 키 스트림을 만들어 내는 스트림 암호
- 블록을 암호화할 때마다 1씩 증가해가는 카운터를 암호화해서 키 스트림을 만든다

## 6.1 카운터 만드는 법

- 카운터 초기값
  - 암호화 때마다 다른 값(nonce, 비표)을 기초로 해서 작성

66 1F 98 CD 37 A3 8B 4B 00 00 00 00 00 00 00 01



비표

블록 번호

## 6.1 카운터 만드는 법

- 평문 블록1용의 카운터(초기값)

66 1F 98 CD 37 A3 8B 4B 00 00 00 00 00 00 00 01

- 평문 블록2용의 카운터

66 1F 98 CD 37 A3 8B 4B 00 00 00 00 00 00 00 02

- 평문 블록3용의 카운터

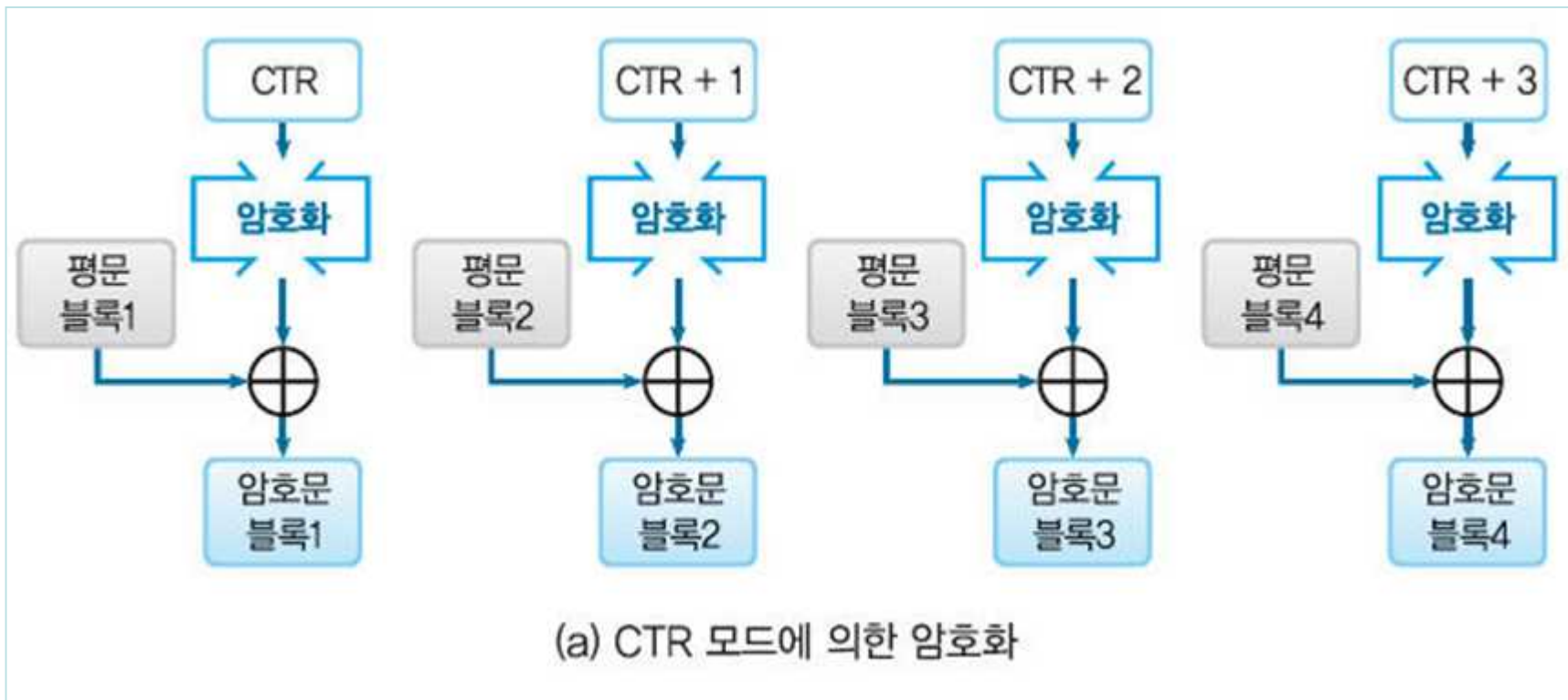
66 1F 98 CD 37 A3 8B 4B 00 00 00 00 00 00 00 03

- 평문 블록4용의 카운터

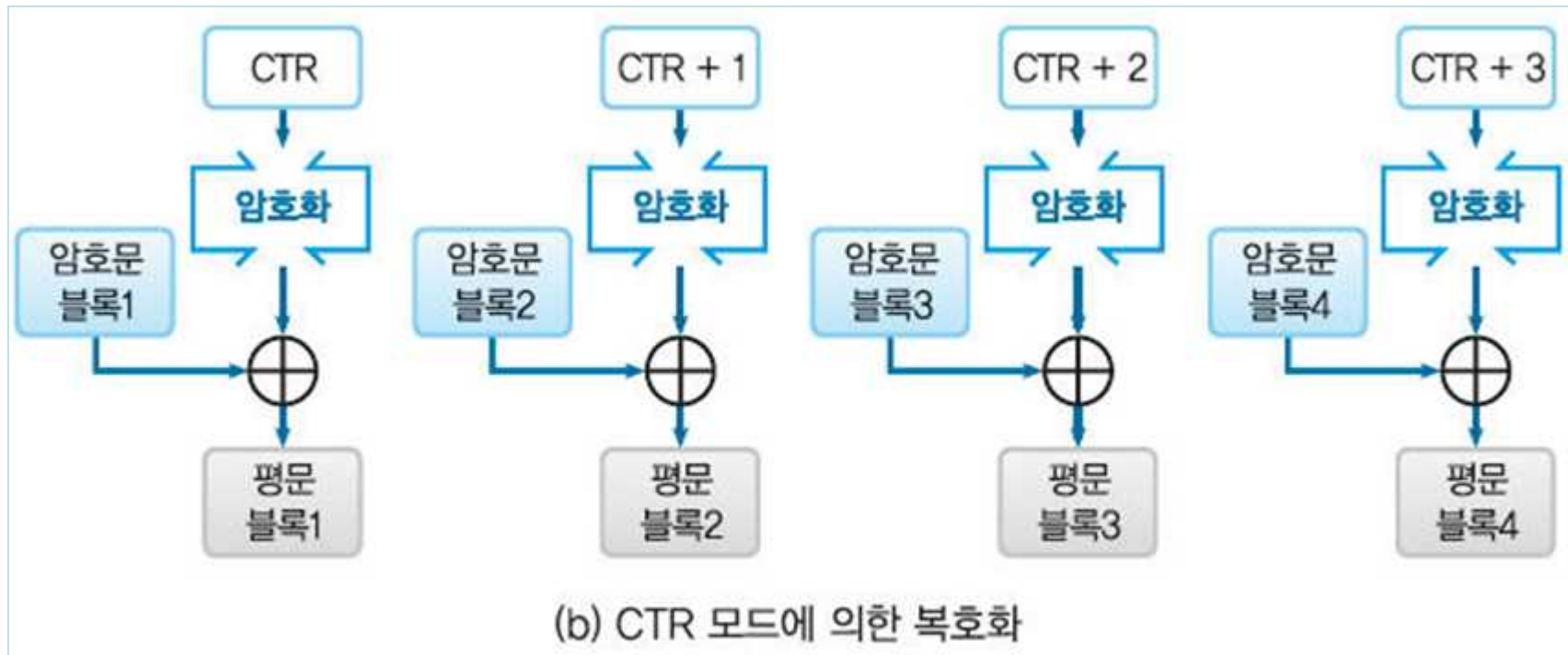
66 1F 98 CD 37 A3 8B 4B 00 00 00 00 00 00 00 04



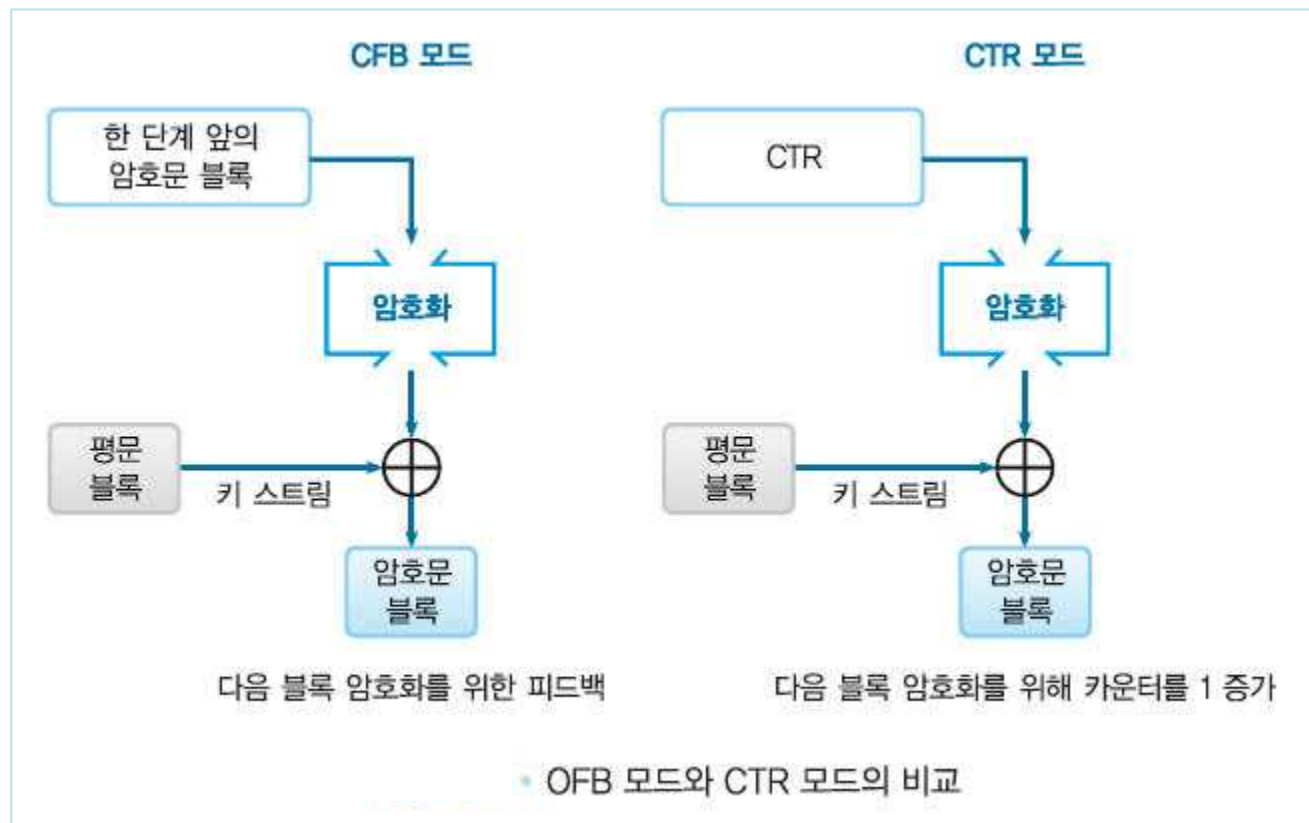
# CTR 모드 암호화



# CTR 모드 복호화



## 6.2 OFB 모드와 CTR 모드의 비교



## 6.3 CTR 모드의 특징

- CTR 모드의 암호화와 복호화는 완전히 같은 구조
- 프로그램으로 구현하는 것이 매우 간단
- OFB 모드와 같은 스트림 암호의 특징
- CTR 모드에서는 블록을 임의의 순서로 암호화 · 복호화할 수 있다
- 병렬 처리가 가능한 시스템에서는 CTR 모드를  
이용하여 자료를 고속으로 처리

## 6.4 오류와 기밀성

- CTR 모드의 암호문 블록에서 1비트의 반전이 발생한다고 가정
  - 복호화를 수행하면, 반전된 비트에 대응하는 평문 블록의 1비트만이 반전 되고, 오류는 확대되지 않는다.
- OFB 모드에서는 키 스트림의 1블록을 암호화한 결과가, 암호화 전의 결과와 우연히 같아졌다고 하면 그 이후 키 스트림은 완전히 같은 값의 반복된다.
  - CTR 모드에서는 그런 걱정은 없음

## 6.5 모드선택

**각 모드의 특징을 정리**

# ECB 모드

	이름	장점	단점	비고
E C B 모 드	Electric CodeBook 전자 부호표 모드	<ul style="list-style-type: none"> <li>간단</li> <li>고속</li> <li>병렬 처리 가능 (암호화, 복호화 양쪽)</li> </ul>	<ul style="list-style-type: none"> <li>평문 속의 반복이 암호문에 반영된다.</li> <li>암호문 블록의 삭제나 교체에 의한 평문의 조작이 가능</li> <li>비트 단위의 에러가 있는 암호문을 복호화하면, 대응하는 블록이 에러가 된다.</li> <li>재전송 공격이 가능</li> </ul>	사용해서는 안된다

# CBC 모드

	이름	장점	단점	비고
C B C 모 드	Cipher Block Chaining 암호 블록 연쇄 모드	<ul style="list-style-type: none"> <li>▪ 평문의 반복은 암호문에 반영되지 않는다.</li> <li>▪ 병렬 처리 가능 (복호화만)</li> <li>▪ 임의의 암호문 블록을 복호화할 수 있다.</li> </ul>	<ul style="list-style-type: none"> <li>▪ 비트 단위의 에러가 있는 암호문을 복호화하면, 1블록 전체와 다음 블록의 대응하는 비트가 에러가 된다.</li> <li>▪ 암호화에서는 병렬 처리를 할 수 없다.</li> </ul>	권장



# CFB 모드

	이름	장점	단점	비고
C F B 모 드	Cipher- FeedBack 암호 피드백 모드	<ul style="list-style-type: none"> <li>패딩이 필요 없다.</li> <li>병렬 처리 가능(복호화만)</li> <li>임의의 암호문 블록을 복호화할 수 있다.</li> </ul>	<ul style="list-style-type: none"> <li>암호화에서는 병렬 처리를 할 수 없다.</li> <li>비트 단위의 에러가 있는 암호문을 복호화하면, 1블록 전체와 다음 블록의 대응하는 비트가 에러가 된다.</li> <li>재전송 공격이 가능</li> </ul>	<p>현재는 사용 안 함.</p> <p>CTR 모드를 사용하는 편이 나음.</p>

# OFB 모드

	이름	장점	단점	비고
0 F B 모 드	Output- FeedBack 출력 피드백 모드	<ul style="list-style-type: none"> <li>▪ 패딩이 필요 없다.</li> <li>▪ 암호화, 복호화의 사전 준비를 할 수 있다.</li> <li>▪ 암호화와 복호화가 같은 구조를 하고 있다.</li> <li>▪ 비트 단위의 에러가 있는 암호문을 복호화하면, 평문의 대응하는 비트만 에러가 된다.</li> </ul>	<ul style="list-style-type: none"> <li>▪ 병렬 처리를 할 수 없다.</li> <li>▪ 적극적 공격자가 암호문 블록을 비트 반전시키면, 대응하는 평문 블록이 비트 반전된다.</li> </ul>	CTR 모드를 사용하는 편이 나음.

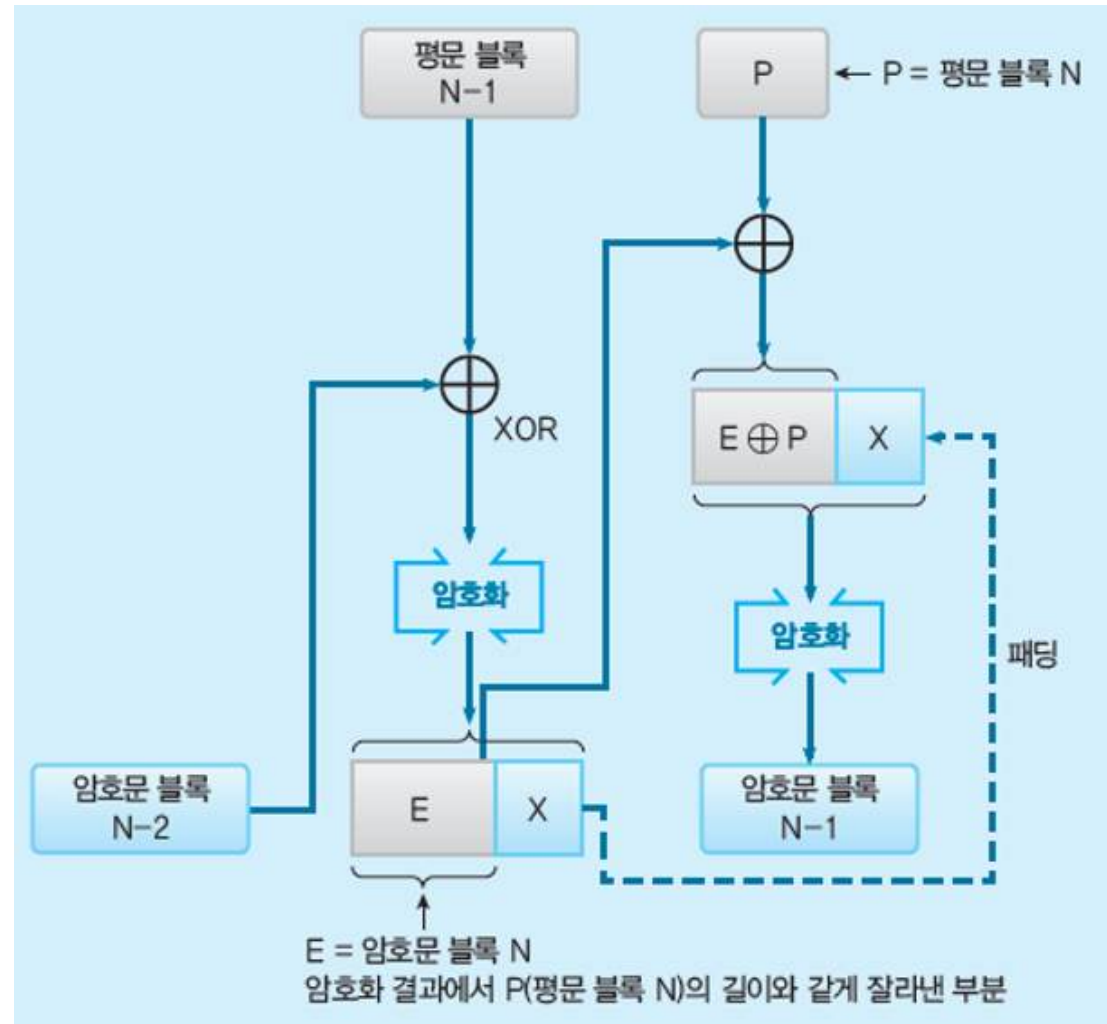
# CTR 모드

	이름	장점	단점	비고
C T R 모 드	CounTeR 카운터 모드	<ul style="list-style-type: none"> <li>패딩이 필요 없다.</li> <li>암호화. 복호화의 사전 준비를 할 수 있다.</li> <li>암호화와 복호화가 같은 구조를 하고 있다.</li> <li>비트 단위의 에러가 있는 암호문을 복호화하면, 평문의 대응하는 비트만 에러가 된다.</li> <li>병렬 처리 가능(암호화. 복호화 양쪽)</li> </ul>	<ul style="list-style-type: none"> <li>적극적 공격자가 암호문 블록을 비트 반전시키면, 대응하는 평문 블록이 비트 반전된다.</li> </ul>	권장

## 칼럼 ( CTS 모드 )

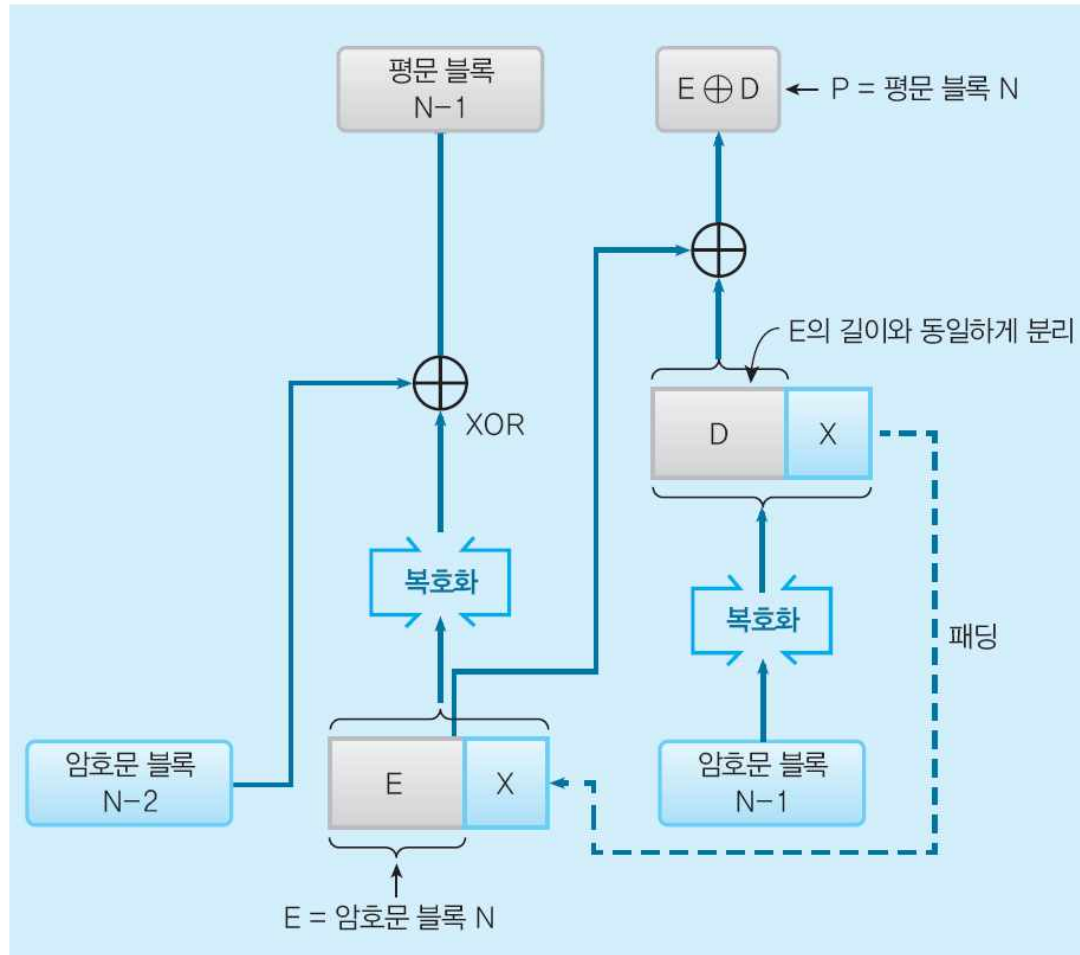
- CTS(Cipher Text Stealing) 모드
- 마지막 블록 한 단계 전의 암호화 블록을 패딩으로 대신 이용
- ECB나 CBC 모드와 조합해 사용
- 마지막 블록을 송신하는 순서를 변경하는 변형된 형태로도 운영
- CBC-CS1, CBC-CS2, CBC-CS3

# CTS 모드 암호화



CTS 모드 암호화

# CTS 모드 복호화



CTS 모드 복호화

## 칼럼 (GCM 모드)

- GCM(Galois / Counter Mode)
- CTR 모드에 인증 기능을 추가한 모드
- CTR 모드로 암호문과 인증자를 생성
- 암호문 위조를 탐지할 수 있음
- 암호화 입력 - 암호키, 평문, Nonce 및 부가 인증 데이터 (ADD) / 출력 - 암호문과 인증값 (tag)
- 복호화 입력 - 암호문, 암호키, Nonce, ADD

## 참고문헌 추가

- 알기쉬운 정보보호 개론, 히로시 유키 지음 (이재광 외 2 공역), 인피니티북스, 2017
- 김기문 외5인, "패딩 오라클 공격에 따른 다양한 패딩방법의 안전성 분석", 정보보호학회 논문지, 제25권 제2호, 2015, pp. 271-278
- 암호학과 네트워크 보안 (cryptography and network security), 포로잔, 베로즈 A, 2008, 한국맥그로힐



**Q & A**  
**Thank You!**