Date

## ⊛ Modes of Operation :

Mainly 5 tyepes of common modes of operation.

1. ECB (Electronic Codebook).
2. CBC (Ciphen Block chaining).
3. CFB (Ciphen Feedback).
4. OFB (Output Feedback).
5. CTR (Counten Mode).

### 1) ECB

→ Each block is encrypted independenthy.

→ Ciphertext = Encrypt (key, Plaintext)

Java implementation :

```
public static string encryptECB (string plaintext,
secretkey ) throws Exception {
Cipher cipher = Cipher.getInstance ("AES/ECB/
                                    PKCS5Padding");
cipher.init (Cipher.ENCRYPT_MODE, key);
return encode (cipher.doFinal (Plaintext.getBytes()));
}
```

output : d4MZ2YB+wRiQ3usC4cCSZg==

Tanzina Fatema, IT21005

Tanzina Fatema, IT21005

## Advantages:

→ Simple to implement.
→ Fast (No chaining Overhead).
→ Supports parallel processing.

### 2) CBC

→ Each plaintext block is XORed with previous ciphertext block before encryption.
→ Needs an IV (Initialization Vector).

### Java implementation:

```
public static string encryptCBC(String plaintext,
SecretKey, IvParameterSpec iv) throws Exception {
Cipher cipher = Cipher.getInstance("AES/CBC/PKCS-
5PPadding");
cipher.init(Cipher.ENCRYPT-Mode, key, iv);
return encode(cipher.doFinal(Plaintext.getBytesU));
}
```

Output: JX3cJh6f3H8bV25D4u9y+w==

Tanzina Fatema, IT21005

Advantages:

→ secure against pattern leakage.

→ widely used in practice.

## 3) CFB

→ Turns block cipher into a self-synchronizing stream cipher.

→ Encrypts IV or previous ciphertext, then XORs with plaintext to get ciphertext.

Java implementation:

```
public static string encryptCFB(String plaintext,
secretkey key, IvparameterSpec iv) throws
Exception {Cipher cipher = cipher.getInstance(
"AES/CFB/PKCS5Padding");
cipher.init(Cipher.ENCRYPT_MODE, key, iv);
return encode(cipher.doFinal(plaintext.get
Bytes()));
}
```

output : 5RU8KHL9JPGLsFOS6ZiwP=
O8s2vGcR6kDXHlj46RBG1E=

Tanzina Fatema, IT21005

## Advantages:

→ Converts block cipher into stream cipher.

→ Good for real time on byte at a time encryption.

### 4) OFB

→ Similar to CFB, but instead of feedback from ciphertext, uses output of block cipher.

→ $O_i = E(K, O_{i-1})$, $C_i = P_i \oplus O_i$

Java implementation:

```
public static String encryptOFB(String plaintext, secret
key key, IvparameterSpec iv) throws Exception{Cipher
cipher = Cipher.getInstance("AES/OFB/PKCS5Padding");
cipher.init(Cipher.ENCRYPT_MODE, key, iv);
return encode(cipher.doFinal(plaintext.getBytes()));
}
```

Output : s7b7dVvCg8S6cFz70F2VnDi2tNPGrFbf3
nOKMmEeyb+M=

Date

Advantages:

→ No error propagation
→ Also acts like a stream cipher.
→ IV reuse doesn't reveal plaintext.

## 5) CTR

→ Uses a counter that increments for each block.
→ Encrypt the counter and XOR with plaintext.
→ $C_i = P_i \oplus E(k, Counter_i)$

Java implementation:

```
public static String encryptCTR(String plaintext,
secretkey key, IvparameterSpec iv) throws Exception
{Cipher cipher = Cipher.getInstance("AES/CTR/
Nopadding");
cipher.init(Cipher.ENCRYPT_MODE, key, iv);
return encode(cipher.doFinal(plaintext.getBytes()));
}
```

Output: zT/jpO3YbZQG7h8Rym9shtEjyo7PvXz4a
EKOcA==

Tanzina Fatema, IT21005

## Advantages:

→ Fully parallelizable
→ Fast and efficient.
→ Good for large data streams.

## RC5

→ It is a symmetric key block cipher designed by Ronald Rivest in 1994.
→ supports variable block size, key size, and number of rounds.

## Java implementation:

```
import org.bouncycastle.jce.provider.BouncyCastle
provider;
import javax.crypto.*; import javax.crypto.spec.*;
import java.security.*; import java.util.*;
public class ShortRC5 {
public static void main(String[]args) throws Exception{
security.addProvider(new BouncyCastleProvider());
SecretKey key = KeyGenerator.getInstance("RC5","BC").
                                        generatKey();
IvparameterSpec iv = new IvparameterSpec(new byte[8]);
.
```

```
Cipher c = Cipher.getInstance("RC5/CBC/PKCS
                              Padding", "BC");
c.init(Cipher.ENCRYPT_MODE, key, iv);
String ct = Base64.getEncoder().encodeToString
        (c.doFinal("Secret Msg".getBytes()));
c.init(Cipher.DECRYPT_MODE, key, iv);
System.out.println("D: " + new String(c.doFinal
    (Base64.getDecoder().decode(ct))));
  }
}
```

Output : D : Secret Msg

Advantages :

→ Flexible design (block size: 32/64/128).
→ Very efficient in both hardware and software.
→ Provides strong mixing of input bits.
→ suitable for embedded systems or constrained devices.

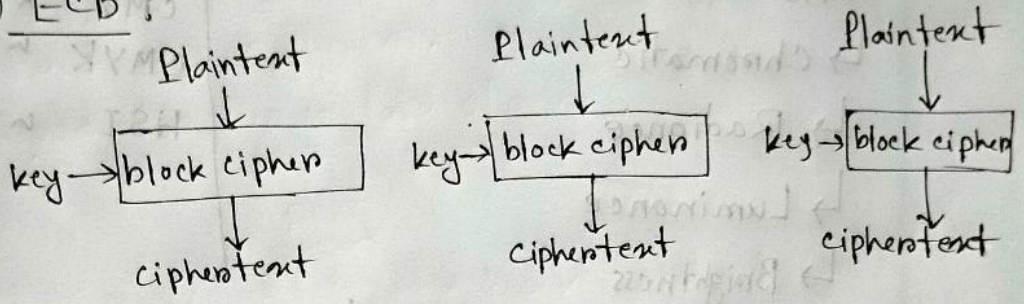
Tanzina Fatema, IT21005

Date :

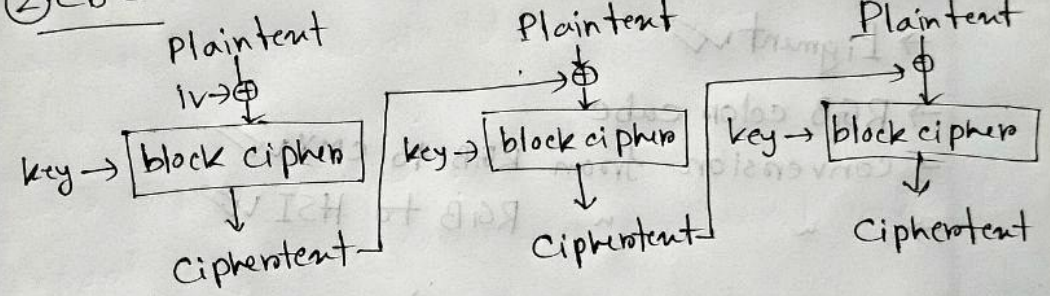
Tanzina Fatema, IT21005

## Block diagram :

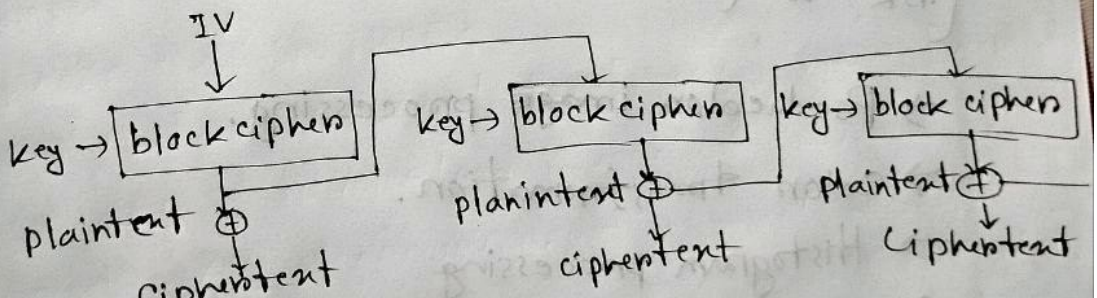
### (1) ECB :




### (2) CBC :



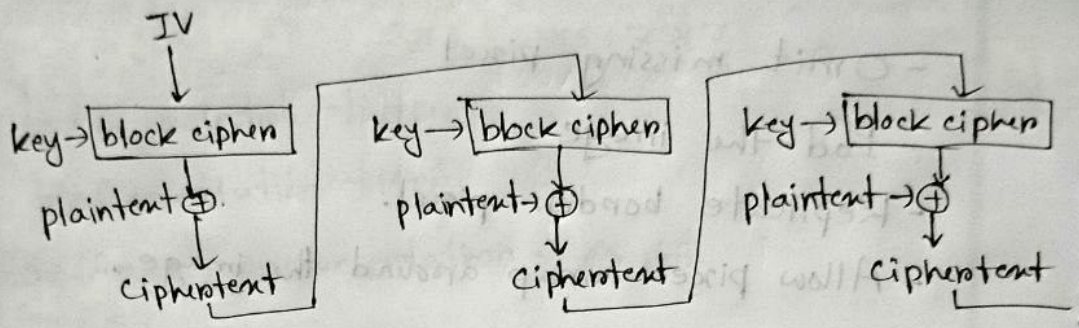

### (3) ~~CFB~~ OFB :



Tanzina Fatema, IT21005

Date

④ ~~OFB~~ CFB:




⑤ CTR: