**Block Cipher Mode of Operation**

Block cipher mode of operation is an algorithm that uses a [block cipher](https://en.wikipedia.org/wiki/Block_cipher) to provide [information security](https://en.wikipedia.org/wiki/Information_security) such as [confidentiality](https://en.wikipedia.org/wiki/Confidentiality) or [authenticity](https://en.wikipedia.org/wiki/Authentication).A block cipher by itself is only suitable for the secure cryptographic transformation (encryption or decryption) of one fixed-length group of [bits](https://en.wikipedia.org/wiki/Bit) called a [block](https://en.wikipedia.org/wiki/Block_(data_storage)). A mode of operation describes how to repeatedly apply a cipher's single-block operation to securely transform amounts of data larger than a block.Encryption algorithms are divided into two categories based on the input type, as a block cipher and stream cipher. **Block cipher** is an encryption algorithm that takes a fixed size of input say *b* bits and produces a ciphertext of *b* bits again. If the input is larger than *b* bits it can be divided further. Most modes require a unique binary sequence, often called an [initialization vector](https://en.wikipedia.org/wiki/Initialization_vector) (IV), for each encryption operation. The IV has to be non-repeating and, for some modes, random as well. The initialization vector is used to ensure distinct [ciphertexts](https://en.wikipedia.org/wiki/Ciphertext" \o "Ciphertext) are produced even when the same [plaintext](https://en.wikipedia.org/wiki/Plaintext) is encrypted multiple times independently with the same [key](https://en.wikipedia.org/wiki/Key_(cryptography)). Block ciphers may be capable of operating on more than one [block size](https://en.wikipedia.org/wiki/Block_size_(cryptography)), but during transformation the block size is always fixed. Block cipher modes operate on whole blocks and require that the last part of the data be [padded](https://en.wikipedia.org/wiki/Padding_(cryptography)) to a full block if it is smaller than the current block size.There are, however, modes that do not require padding because they effectively use a block cipher as a [stream cipher](https://en.wikipedia.org/wiki/Stream_cipher).

**Initialization of vector(IV):**

An initialization vector (IV) or starting variable (SV) is a block of bits that is used by several modes to randomize the encryption and hence to produce distinct ciphertexts even if the same plaintext is encrypted multiple times, without the need for a slower re-keying process.

An initialization vector has different security requirements than a key, so the IV usually does not need to be secret. For most block cipher modes it is important that an initialization vector is never reused under the same key.Many block cipher modes have stronger requirements, such as the IV must be [random](https://en.wikipedia.org/wiki/Random) or [pseudorandom](https://en.wikipedia.org/wiki/Pseudorandom). Some block ciphers have particular problems with certain initialization vectors, such as all zero IV generating no encryption

**Padding:**

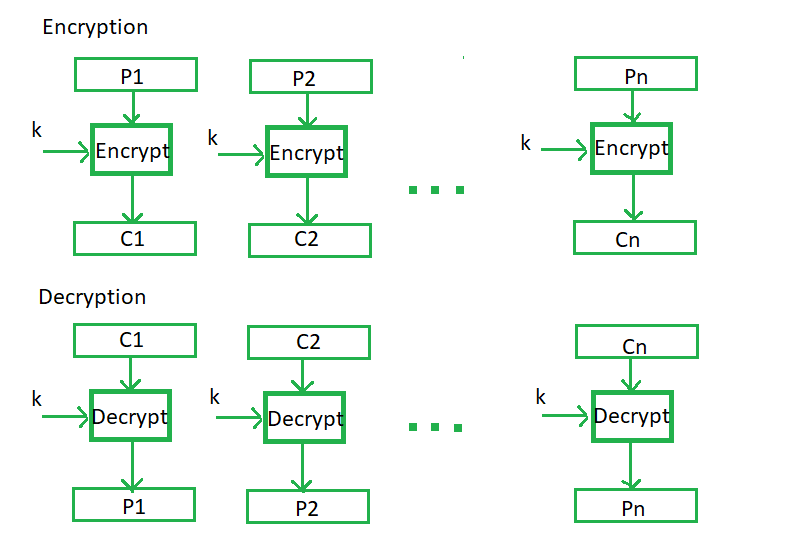
A [block cipher](https://en.wikipedia.org/wiki/Block_cipher) works on units of a fixed [size](https://en.wikipedia.org/wiki/Block_size_(cryptography)) (known as a *block size*), but messages come in a variety of lengths. So some modes (namely [ECB](https://en.wikipedia.org/wiki/Block_cipher_modes_of_operation#ECB) and [CBC](https://en.wikipedia.org/wiki/Block_cipher_modes_of_operation#CBC)) require that the final block be padded before encryption. Several [padding](https://en.wikipedia.org/wiki/Padding_(cryptography)) schemes exist. The simplest is to add [null bytes](https://en.wikipedia.org/wiki/Null_character) to the [plaintext](https://en.wikipedia.org/wiki/Plaintext) to bring its length up to a multiple of the block size, but care must be taken that the original length of the plaintext can be recovered; this is trivial, for example, if the plaintext is a [C](https://en.wikipedia.org/wiki/C_(programming_language)) style [string](https://en.wikipedia.org/wiki/Literal_string) which contains no null bytes except at the end. Slightly more complex is the original [DES](https://en.wikipedia.org/wiki/Data_Encryption_Standard) method, which is to add a single one [bit](https://en.wikipedia.org/wiki/Bit), followed by enough zero [bits](https://en.wikipedia.org/wiki/Bit) to fill out the block; if the message ends on a block boundary, a whole padding block will be added. Most sophisticated are CBC-specific schemes such as [ciphertext stealing](https://en.wikipedia.org/wiki/Ciphertext_stealing" \o "Ciphertext stealing) or [residual block termination](https://en.wikipedia.org/wiki/Residual_block_termination), which do not cause any extra ciphertext, at the expense of some additional complexity.

CFB, OFB and CTR modes do not require any special measures to handle messages whose lengths are not multiples of the block size, since the modes work by [XORing](https://en.wikipedia.org/wiki/Exclusive_or" \o "Exclusive or) the plaintext with the output of the block cipher. The last partial block of plaintext is XORed with the first few bytes of the last [keystream](https://en.wikipedia.org/wiki/Keystream) block, producing a final ciphertext block that is the same size as the final partial plaintext block. This characteristic of stream ciphers makes them suitable for applications that require the encrypted ciphertext data to be the same size as the original plaintext data, and for applications that transmit data in streaming form where it is inconvenient to add padding bytes.

For different applications and uses, there are several modes of operations for a block cipher.

**Electronic Code Book (ECB):**

Electronic code book is the easiest block cipher mode of functioning. It is easier because of direct encryption of each block of input plaintext and output is in form of blocks of encrypted ciphertext. Generally, if a message is larger than *b* bits in size, it can be broken down into a bunch of blocks and the procedure is repeated.



**Advantages of using ECB:–**

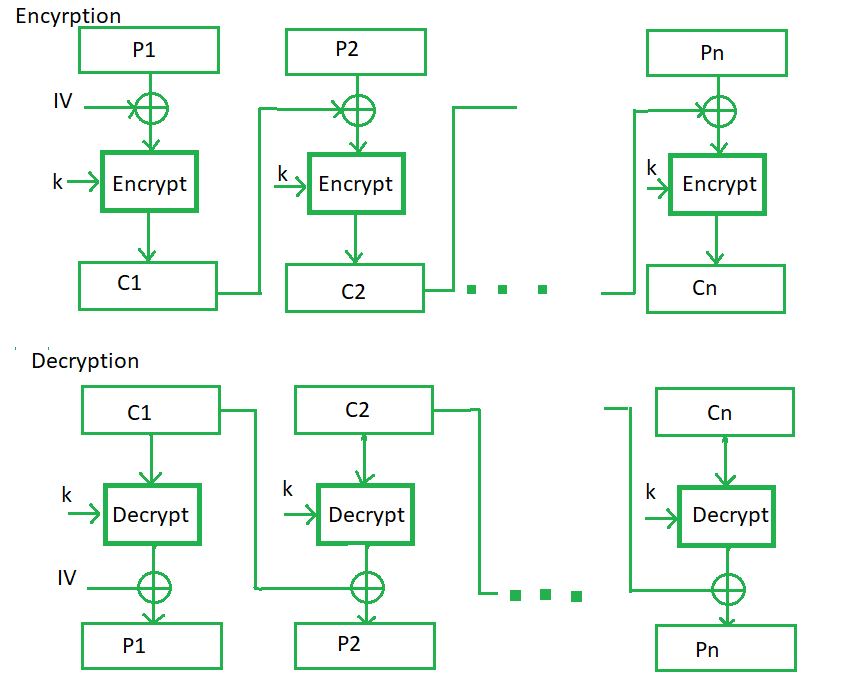
* Parallel encryption of blocks of bits is possible, thus it is a faster way of encryption.
* Simple way of the block cipher.

**Disadvantages of using ECB:-**

* Prone to cryptanalysis since there is a direct relationship between plaintext and ciphertext.

**Cipher Block Chaining(CBC):**

Cipher block chaining or CBC is an advancement made on ECB since ECB compromises some security requirements. In CBC, the previous cipher block is given as input to the next encryption algorithm after XOR with the original plaintext block. In a nutshell here, a cipher block is produced by encrypting an XOR output of the previous cipher block and present plaintext block.



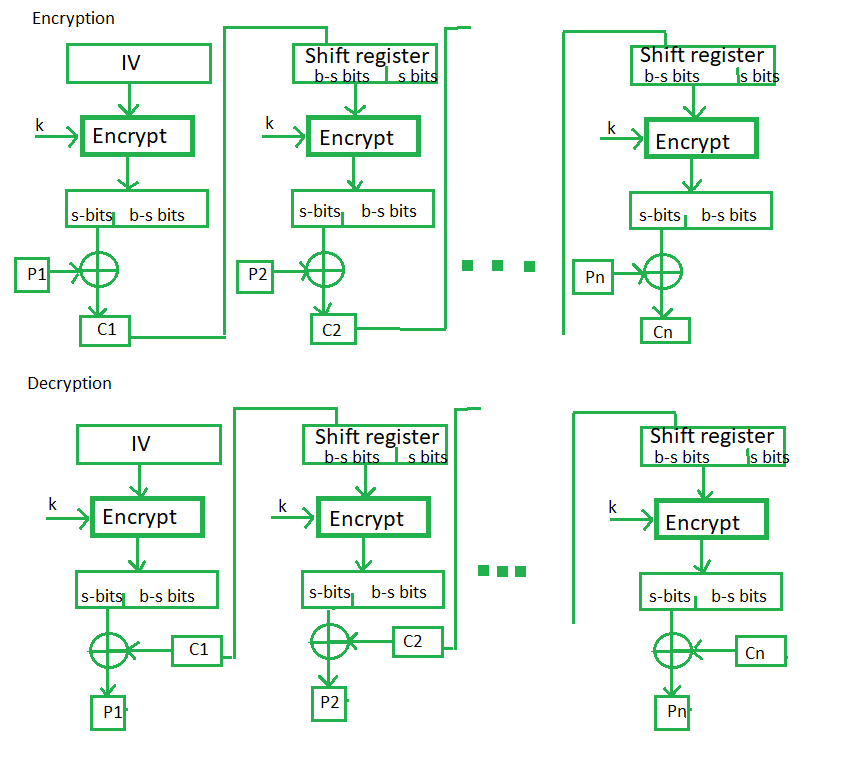
**Advantages of CBC:** –

* CBC works well for input greater than *b* bits.
* CBC is a good authentication mechanism.
* Better resistive nature towards cryptanalysis than ECB.

**Disadvantages of CBC:** –

* Parallel encryption is not possible since every encryption requires a previous cipher.

**CipherFeedbackMode(CFB)**:  
In this mode the cipher is given as feedback to the next block of encryption with some new specifications: first, an initial vector IV is used for first encryption and output bits are divided as a set of *s*and *b-s* bits.The left-hand side *s*bits are selected along with plaintext bits to which an XOR operation is applied. The result is given as input to a shift register having b-s bits to lhs,s bits to rhs and the process continues. The encryption and decryption process for the same is shown below, both of them use encryption algorithms.



**Advantages of CFB**: –

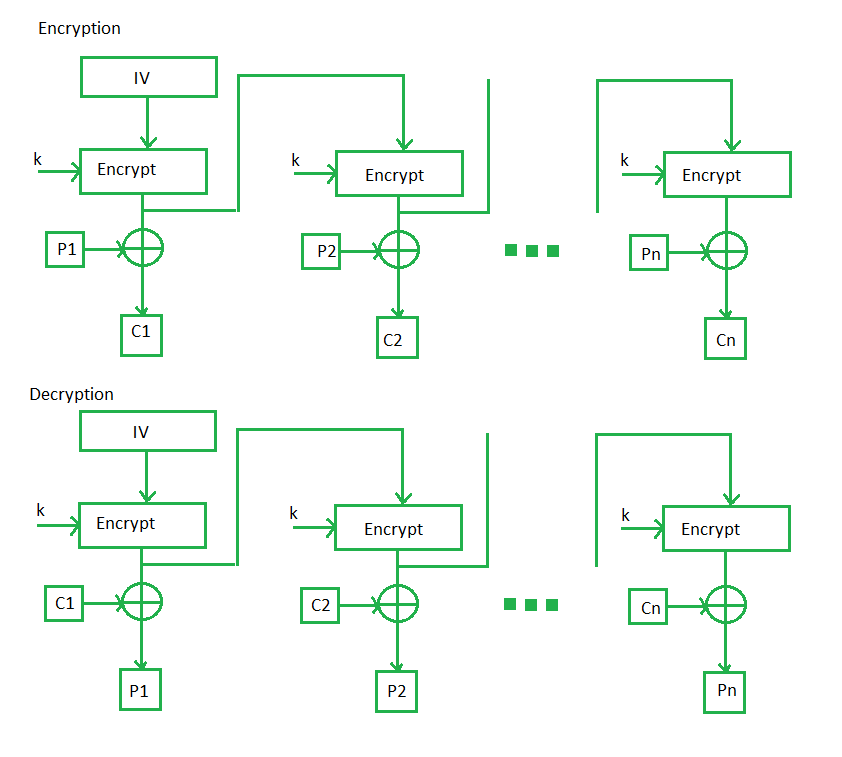
* Since, there is some data loss due to the use of shift register, thus it is difficult for applying cryptanalysis.

**Disadvantages**:-

* The encryption cannot tolerate block losses,nor can multiple blocks be encrypted in parallel

**OutputFeedbackMode:**

The output feedback mode follows nearly the same process as the Cipher Feedback mode except that it sends the encrypted output as feedback instead of the actual cipher which is XOR output. In this output feedback mode, all bits of the block are sent instead of sending selected *s* bits. The Output Feedback mode of block cipher holds great resistance towards bit transmission errors. It also decreases the dependency or relationship of the cipher on the plaintext.



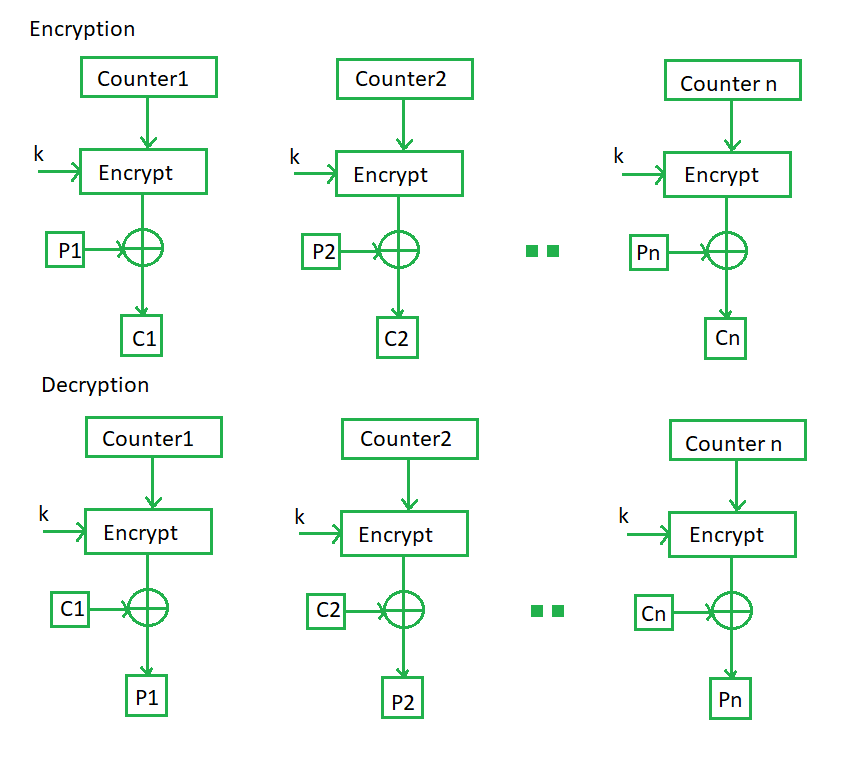
**Advantages of OFB**: –

* In the case of CFB, a single bit error in a block is propagated to all subsequent blocks. This problem is solved by OFB as it is free from bit errors in the plaintext block.

**Disadvantages**-

* Insensitivity to transmisiion errors and applicability to bulk encryption of multiple users transmissons

**CounterMode**  
The Counter Mode or CTR is a simple counter-based block cipher implementation. Every time a counter-initiated value is encrypted and given as input to XOR with plaintext which results in ciphertext block. The CTR mode is independent of feedback use and thus can be implemented in parallel.



**Advantages of Counter**:-

* Since there is a different counter value for each block, the direct plaintext and ciphertext relationship is avoided. This means that the same plain text can map to different ciphertext.
* Parallel execution of encryption is possible as outputs from previous stages are not chained as in the case of CBC.

**Disadvantages**:-

* It requires a synchronous counter at the sender and receiver in this mode

**Conclusion:-**

It defines how the different blocks of a multi-block plaintext should be encrypted and decrypted. By agreeing on a block cipher mode of operation the sender and recipient of a message ensure that they do things the same way and that the data decrypts correctly.

Block cipher have the advantage of high diffusion and strong tamper resistance without detection. They have disadvantage of slower encryption speed since the entire block must be captured for encryption/decryption. Block cipher also breed errors since a mistake in just one symbol could alter the whole block