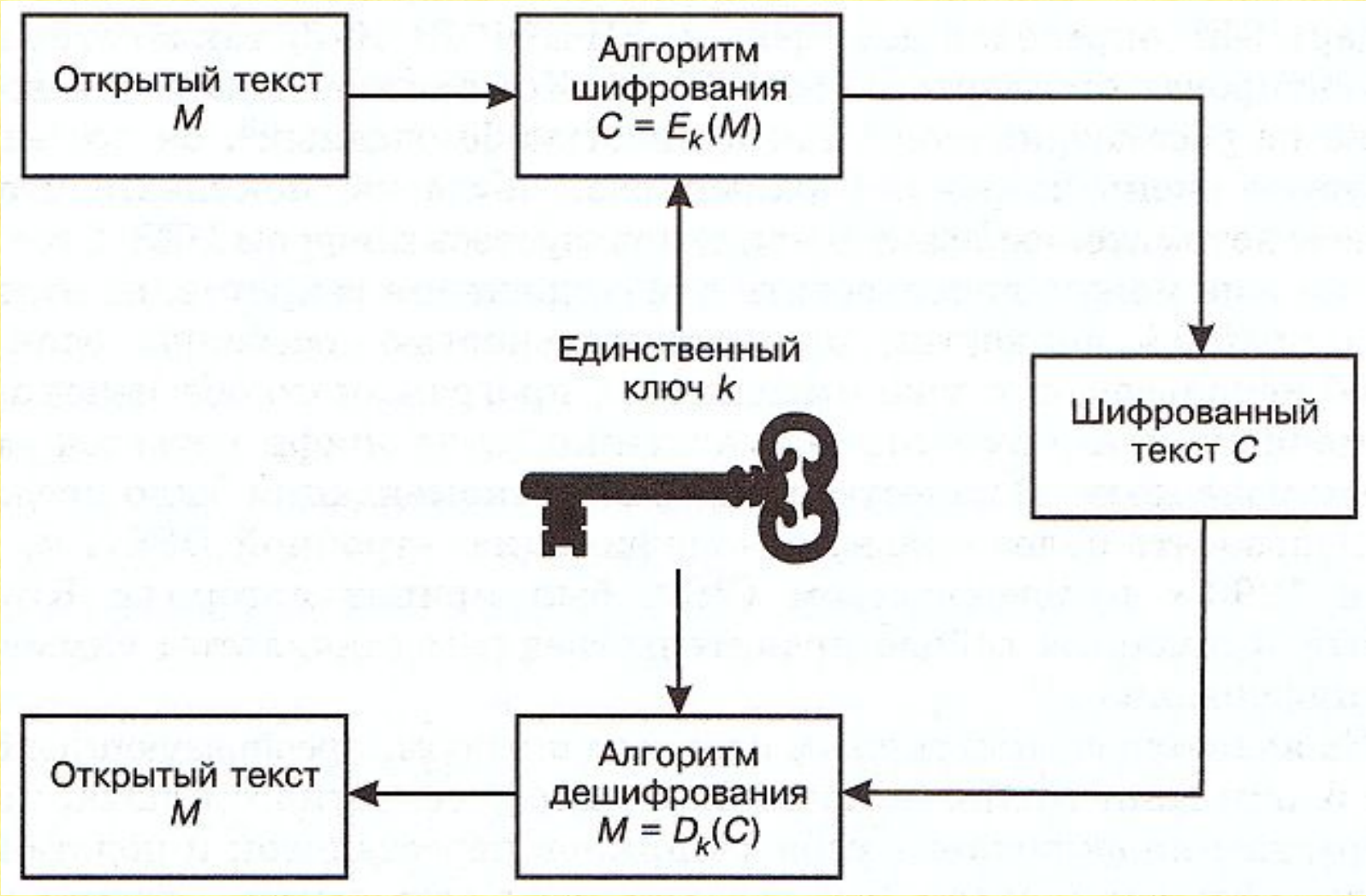
SymmetricAlgorithm

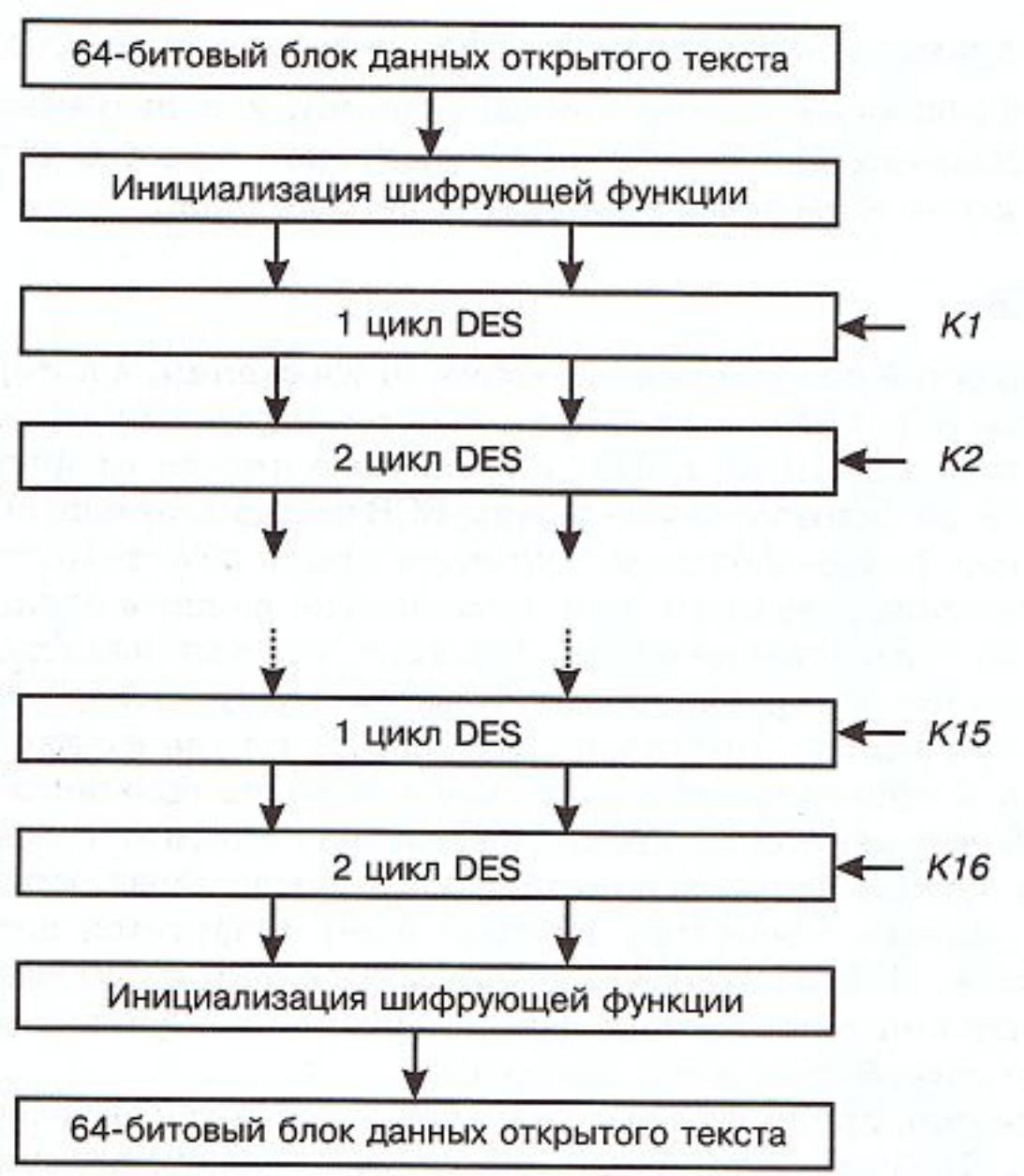
Петелин А.Е.



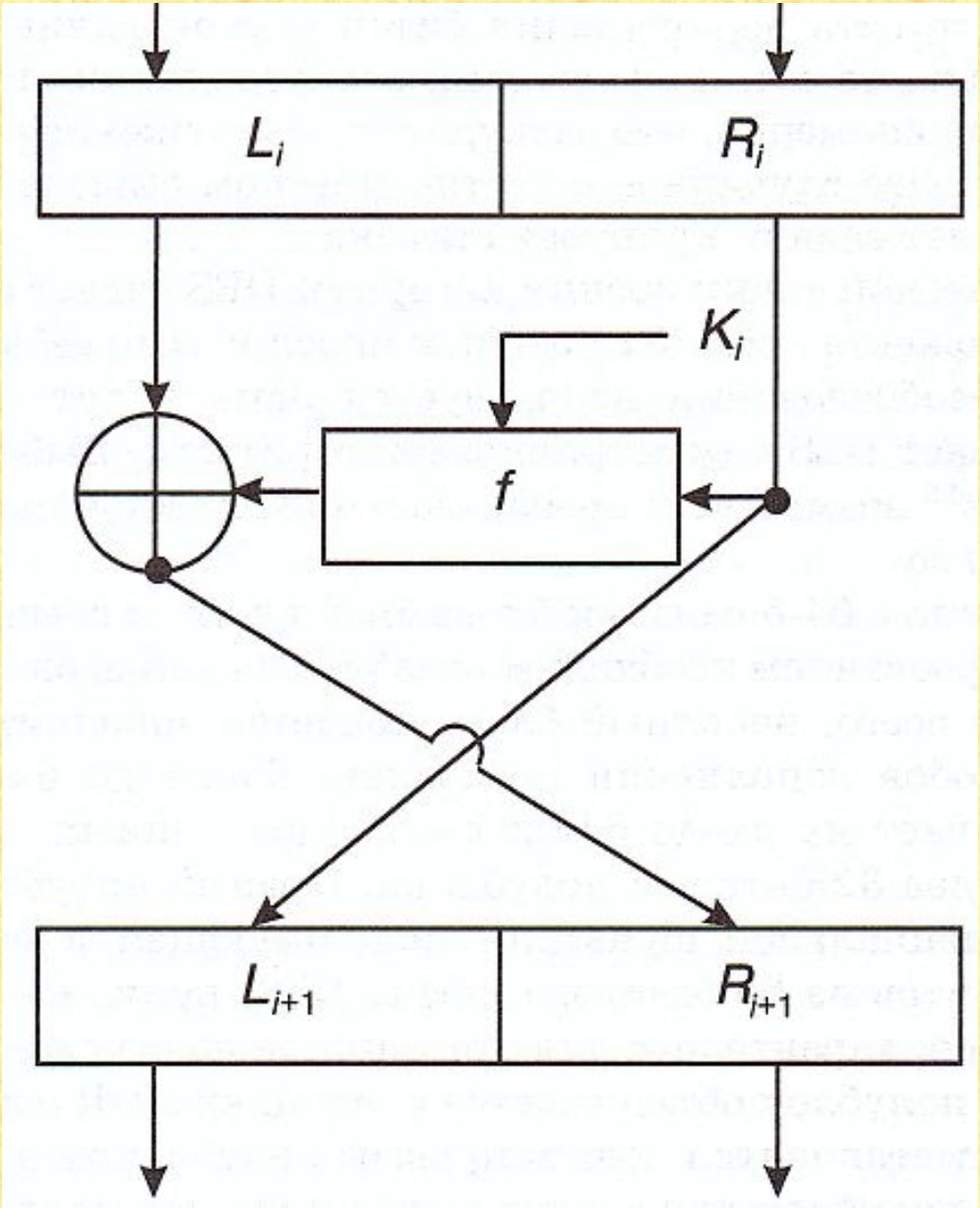
Шифрование: $C = E_k(M)$

Дешифрование: $M = D_k(C)$

Схема работы алгоритма DES



Один цикл алгоритма DES



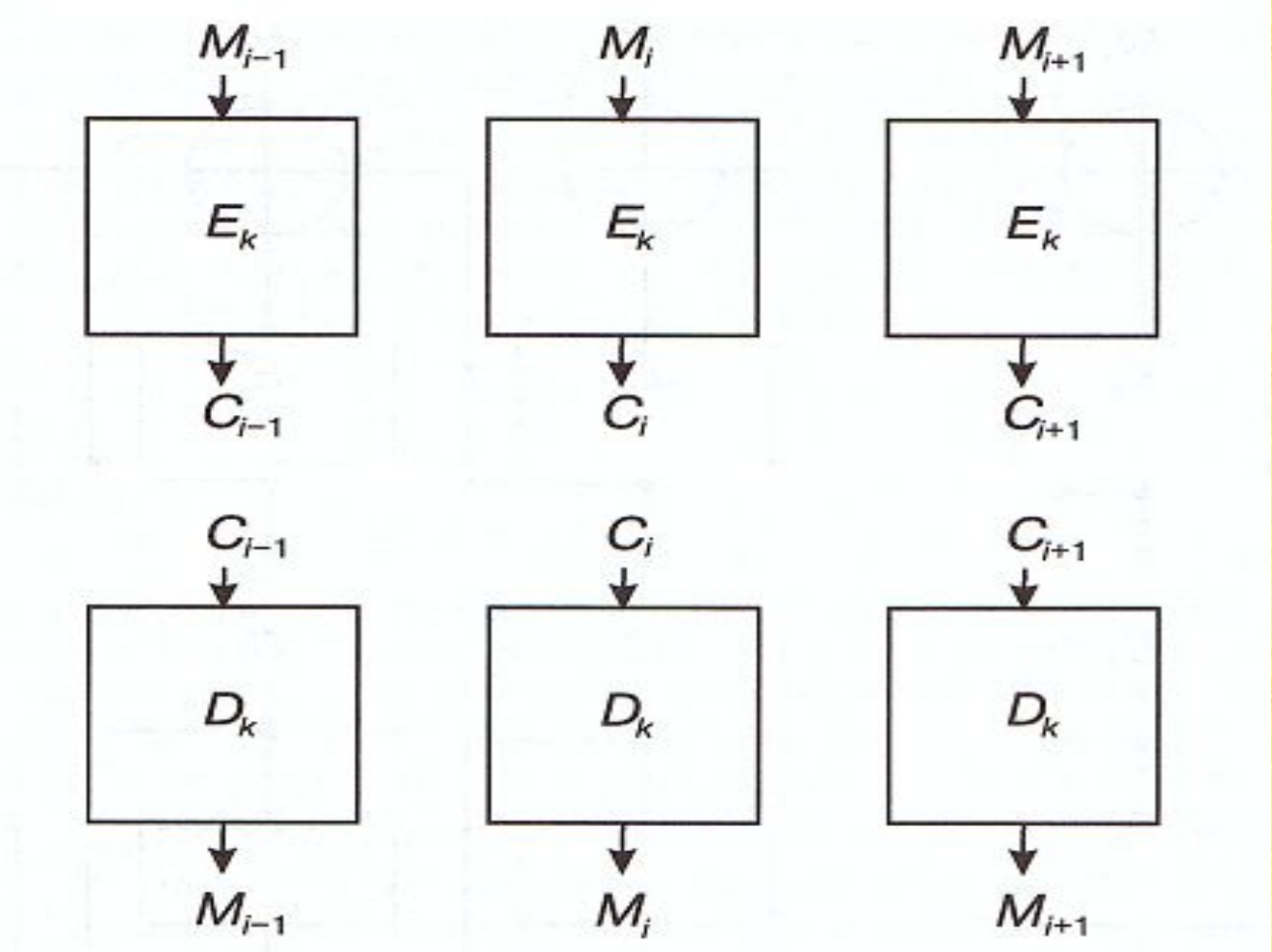
Операционные режимы

- электронная шифровальная книга (Electronic Codebook – ECB);
- сцепление шифрованных блоков (Cipher Block Chaining – CBC);
- шифрованная обратная связь (Cipher Feedback – CFB);
- обратная связь по выходу (Output Feedback – OFB);
- проскальзывание шифрованного текста (Cipher Text Stealing – CTS).

Используемые сокращения

- M_i – i-й 64-битовый блок открытого текста;
- C_i – i-й 64-битовый блок шифрованного текста;
- E_k – функция шифрования DES, использующая ключ k;
- D_k – функция дешифрования DES, использующая ключ k;
- IV – 64-битовый вектор инициализации, используемый в некоторых режимах для имитации предыдущего блока данных в ситуации, когда обрабатывается самый первый блок.

Режим ECB



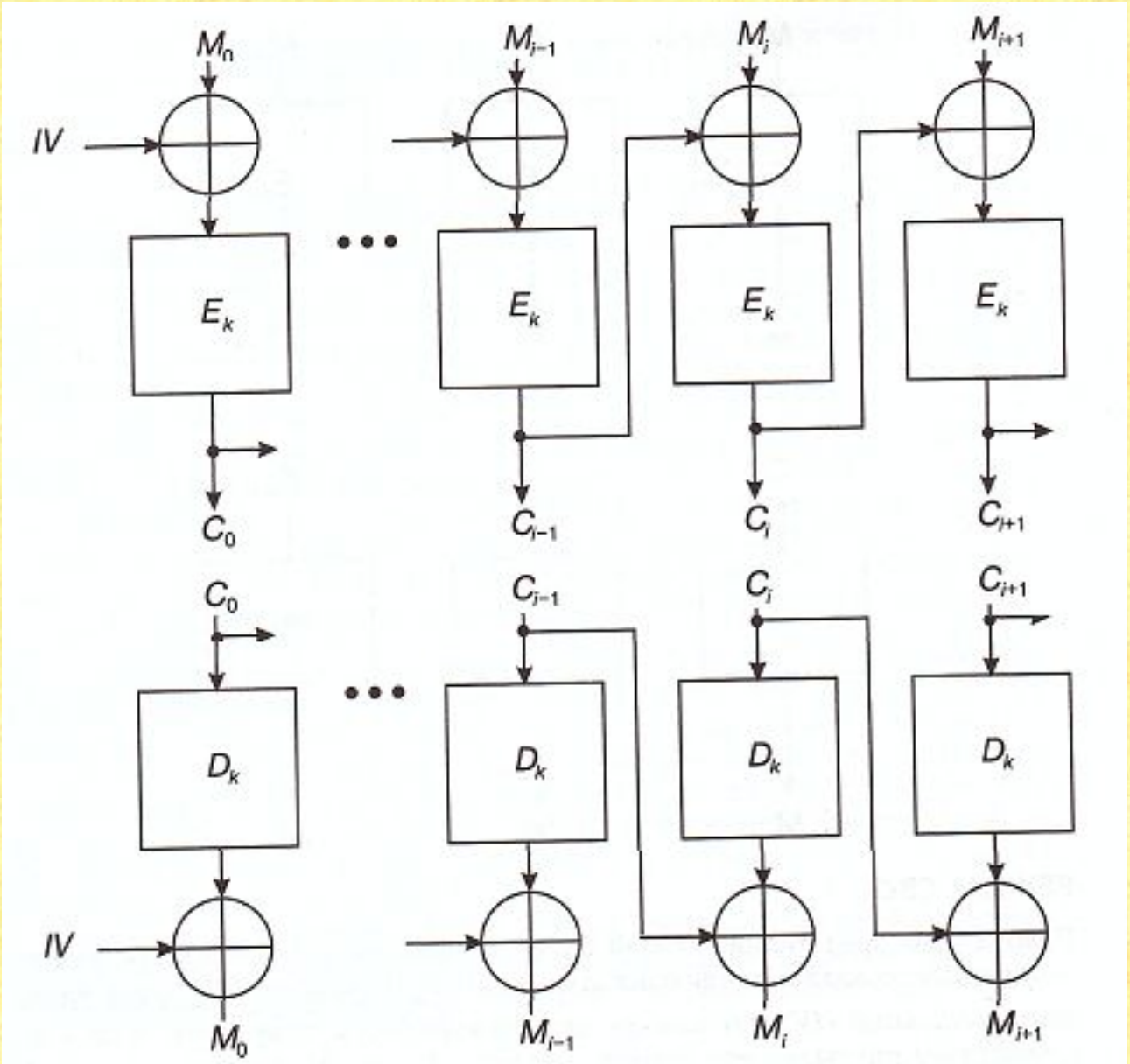
Шифрование в режиме ECB:

$$C_i = E_k(M_i).$$

Дешифрование в режиме ECB:

$$M_i = D_k(C_i).$$

Режим CBC



Шифрование в режиме CBC:

$$C_0 = E_k(M_0 \oplus IV),$$

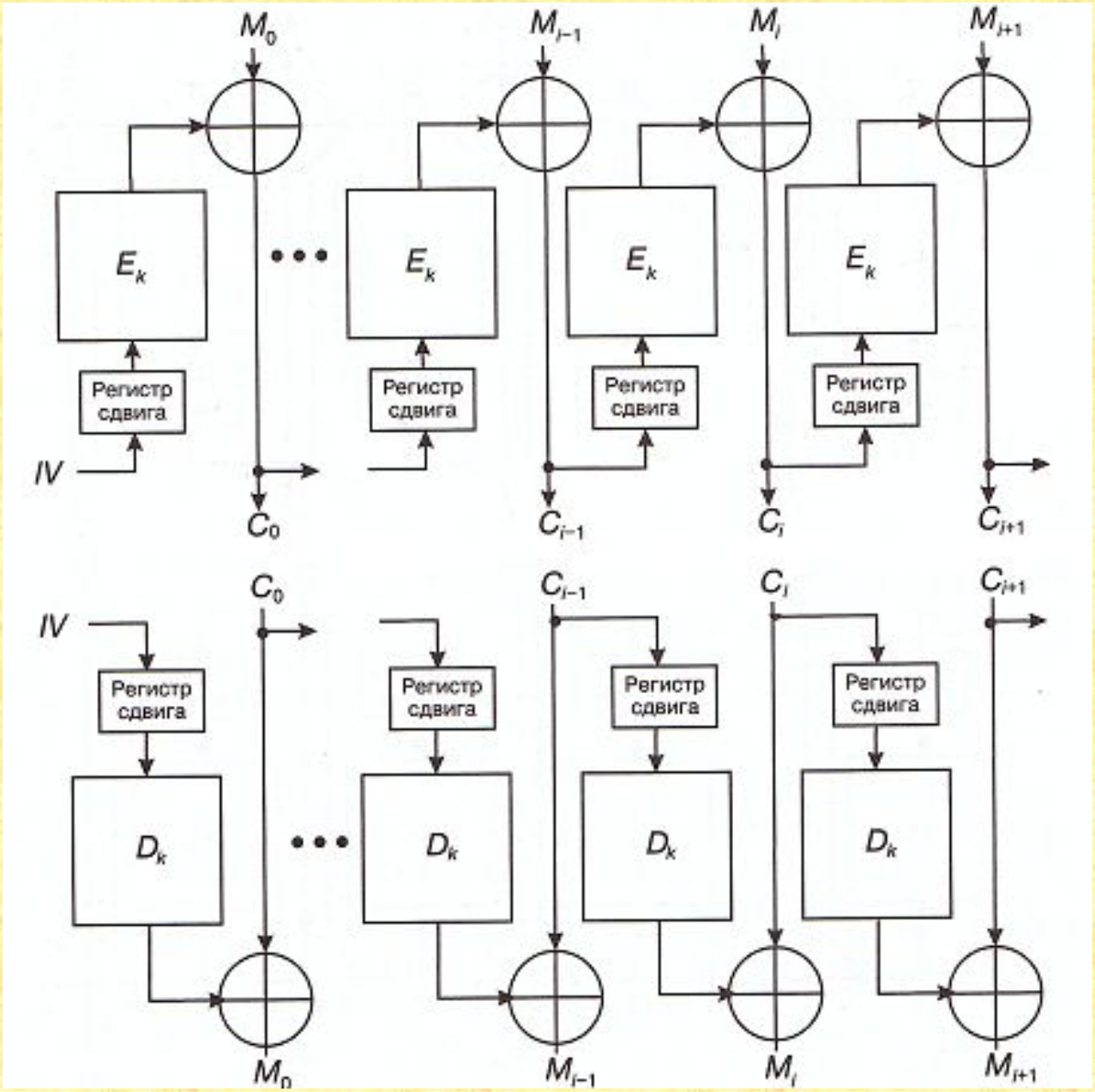
$$C_i = E_k(M_i \oplus C_{i-1}).$$

Дешифрование в режиме CBC:

$$M_0 = D_k(C_0) \oplus IV,$$

$$M_i = D_k(C_i) \oplus C_{i-1}.$$

Режим OFB



Шифрование в режиме CFB:

$$C_0 = M_0 \oplus E_k(IV),$$

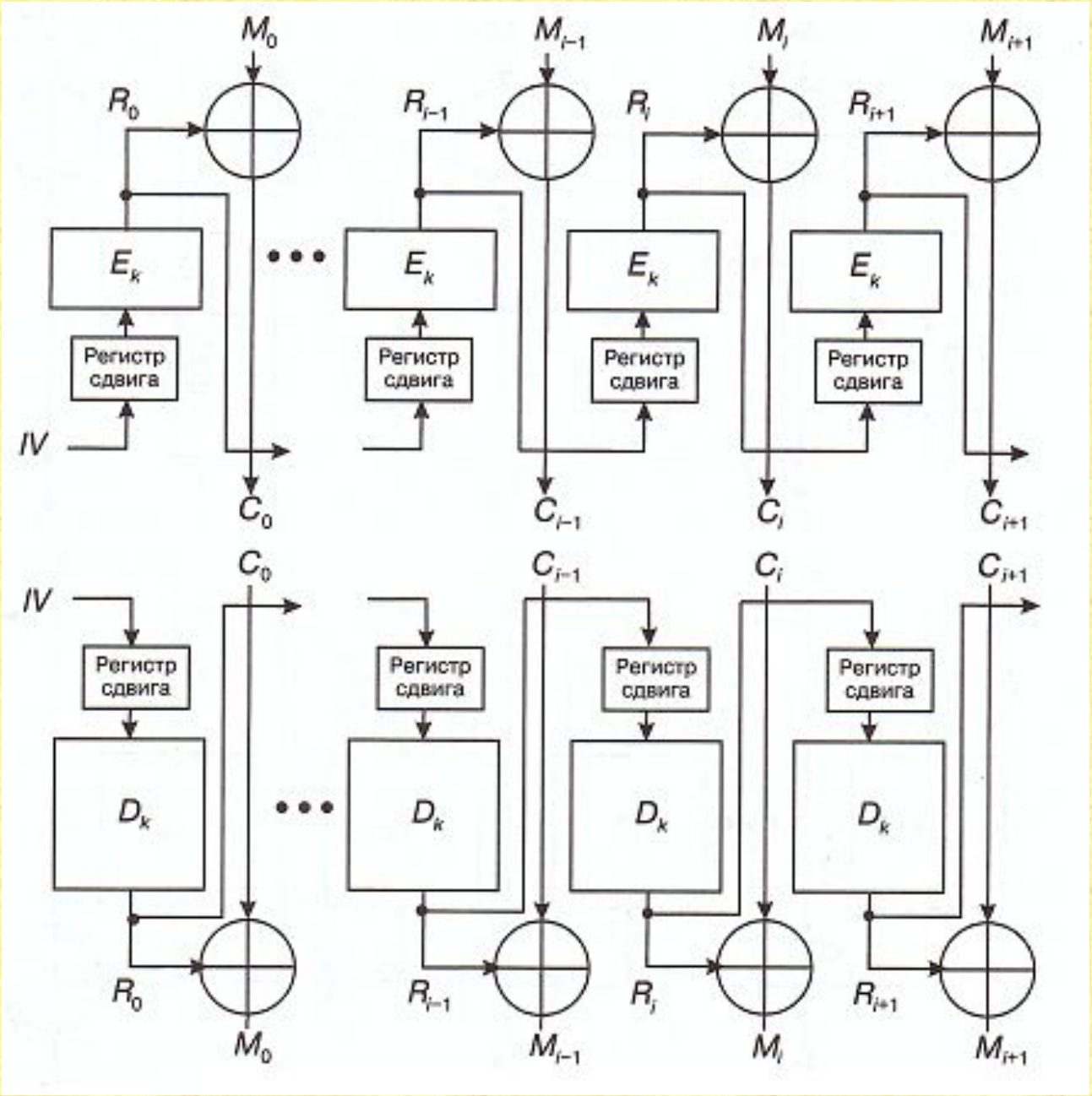
$$C_i = M_i \oplus E_k(C_{i-1}).$$

Дешифрование в режиме CFB:

$$M_0 = C_0 \oplus D_k(IV),$$

$$M_i = C_i \oplus D_k(C_{i-1}).$$

Режим CFB



Шифрование в режиме OFB:

$$C_0 = M_0 \oplus E_k(IV),$$

$$C_i = M_i \oplus E_k(R_{i-1}),$$

Дешифрование в режиме OFB:

$$M_0 = C_0 \oplus D_k(IV),$$

$$M_i = C_i \oplus D_k(R_{i-1}),$$

Алгоритмы шифрования

1. DES;
2. TripleDES;
3. Rijndael;
4. RC2

Дополнения блоков

1. PKCS7;
2. Zeros;
3. None.

Криптографические потоки

```
public CryptoStream(
    Stream stream,
    ICryptoTransform transform,
    CryptoStreamMode mode
);
```

Пример шифрования

```
//Шифрование
SymmetricAlgorithm sa = TripleDES.Create();
sa.GenerateKey();
key = sa.Key;
sa.Mode = CipherMode.ECB;
sa.Padding = PaddingMode.PKCS7;
MemoryStream ms = new MemoryStream();
CryptoStream cs = new CryptoStream(ms, sa.CreateEncryptor(),
                                   CryptoStreamMode.Write);
byte[] plainbytes = Encoding.Default.GetBytes(text.Text);
cs.Write(plainbytes, 0, plainbytes.Length);
cs.Close();
cipherbytes = ms.ToArray();
ms.Close();
//Вывод зашифрованного текста
str = Encoding.Default.GetString(cipherbytes);
textBox2.Text = str;
```

Пример дешифрования

```
//Дешифрование
te = new Byte[str.Length];
te = Encoding.Default.GetBytes(str);
SymmetricAlgorithm sa2 = TripleDES.Create();
sa2.Key = key;
sa2.Mode = CipherMode.ECB;
sa2.Padding = PaddingMode.PKCS7;
MemoryStream ms2 = new MemoryStream(te);
CryptoStream cs2 = new CryptoStream(ms2, sa2.CreateDecryptor(),
                                    CryptoStreamMode.Read);
byte[] plainbytes2 = new Byte[te.Length];
cs2.Read(plainbytes2, 0, te.Length);
cs2.Close();
ms2.Close();
textBox1.Text = Encoding.Default.GetString(plainbytes2);
```