

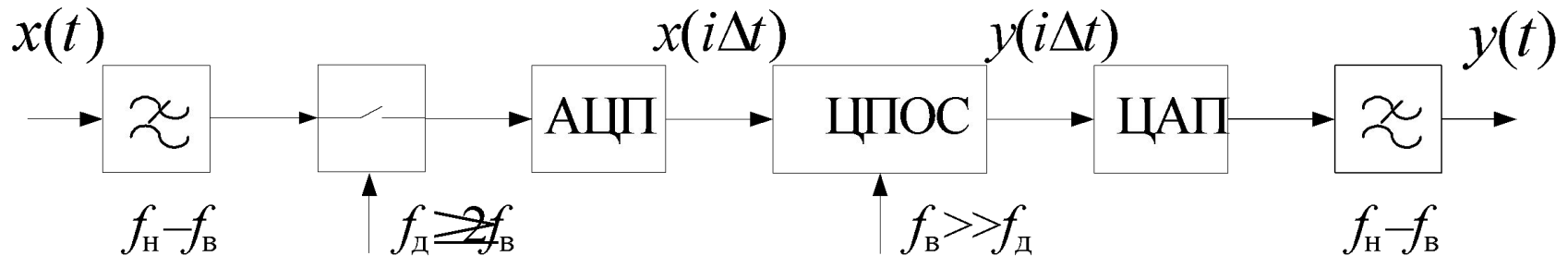
# Основные определения

- **Микропроцессор** —устройство, отвечающее за выполнение арифметических, логических операций и операций управления, записанных в машинном коде, реализованный в виде одной микросхемы или комплекта из нескольких специализированных микросхем.
- **Процессор цифровой обработки сигналов** – это микропроцессор, особенностью работы которого является поточный характер обработки больших объемов данных в реальном масштабе времени и, как правило, с интенсивным обменом данных с внешними устройствами.
- **Реальный масштаб времени**– это такой режим работы устройства, при котором регистрация и арифметическая обработка (а при необходимости и анализ, визуализация, сохранение, систематизация, синтез и передача по каналам связи) данных производится без потерь информации, поступающей от ее источника.

## Типовые задачи, решаемые ПЦОС

- фильтрация сигналов;
- свертка двух сигналов (смешение сигналов);
- вычисление значений авто и кросс-корреляционной функции двух сигналов;
- усиление, нормализация или преобразование сигналов;
- прямое и обратное Фурье-преобразование;

# Структурная схема цифровой обработки сигналов



АЦП- аналого-цифровой преобразователь

ЦПОС- цифровой процессор обработки сигналов

ЦАП- цифро-аналоговый преобразователь

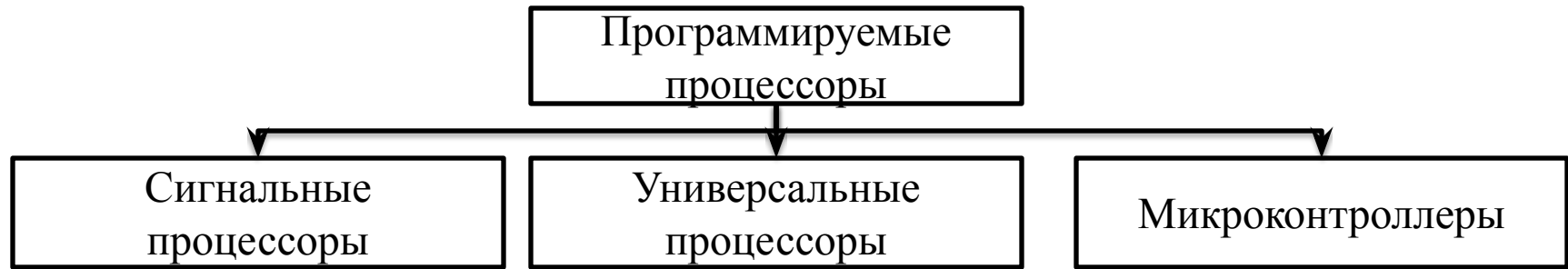
$$f_B = 4 \text{ кГц} \Rightarrow f_d = 8 \text{ кГц} (T_d = 125 \text{ мкс})$$

$$\text{Интервал времени } \Delta t = 4T_d = 500 \text{ мкс}$$

Количество операций, выполняемых ЦПОС за  $\Delta t : N = 2500$

$$T_T = \frac{\Delta t}{N} = \frac{500 \cdot 10^{-6}}{2500} = 200 \text{ нс} \Rightarrow f_T = T_T^{-1} = 5 \text{ МГц}$$

# Классификация процессоров



Если основная задача устройства связана с обработкой данных– то это процессор, если же с управлением– то это контроллер.

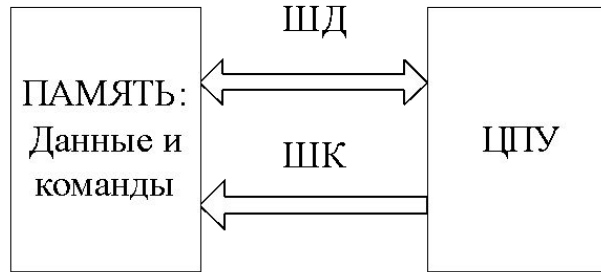
**Универсальные процессоры** используют все передовые технологии, не взирая на стоимость, габариты и расходуемую энергию. В технике связи применяются для управления дорогостоящими системами связи.

**Микроконтроллеры** используются для управления дешевыми малогабаритными устройствами связи. При их изготовлении учитывают стоимость, габариты и потребляемую энергию.

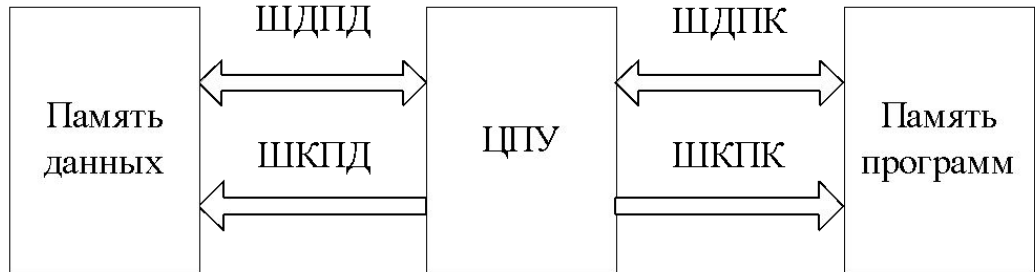
**Сигнальные процессоры** характеризуются упрощенным набором команд, наличием встроенного умножителя. На основе ПЦОС создаются устройства в которых требуется реальный масштаб времени выполнения практически любых арифметических задач.

# Архитектура процессоров

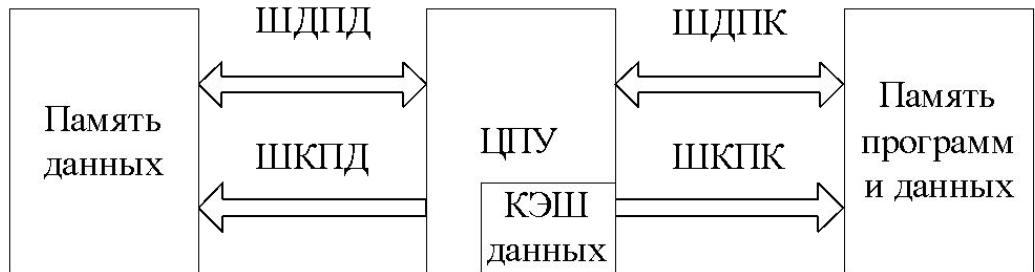
*Фон Неймана*



*Гарвардская архитектура*



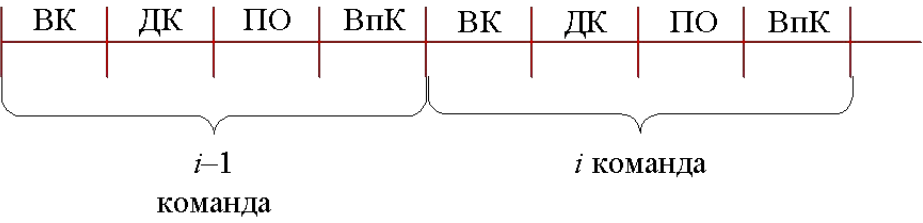
*Модифицированная гарвардская архитектура*



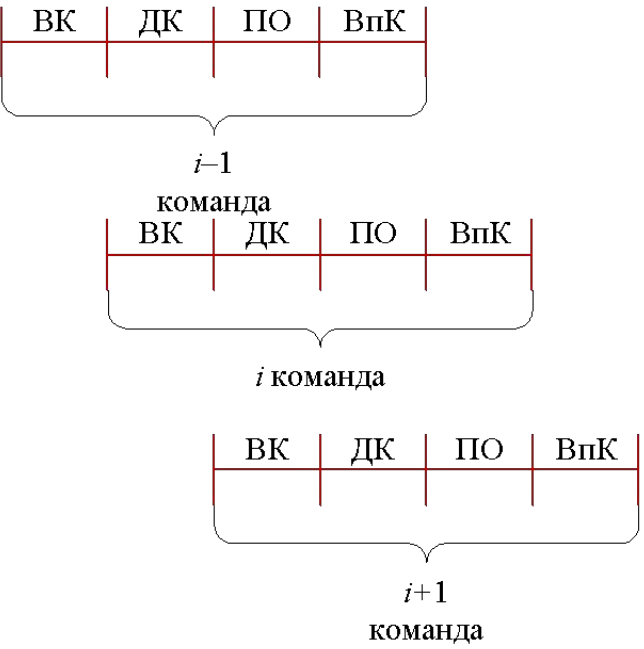
# Структура команды

Выборка команды (ВК)	Декодирование команды (ДК)	Подготовка операндов (ПО)	Выполнение команды (ВпК)
----------------------	----------------------------	---------------------------	--------------------------

## Неконвейерное выполнение команд



## Конвейерное выполнение команд



# Системы счисления

$$R_n = a_{L-1} \cdot n^{L-1} + a_{L-2} \cdot n^{L-2} + \dots + a_1 \cdot n^1 + a_0 \cdot n^0$$

$a_i \in [0 \dots n]$  – цифры

$n$  – основание системы счисления

двоичная –  $a = 0, 1$

восьмиричная –  $a = 0, 1 \dots 7$

десятичная –  $a = 0, 1 \dots 9$

шестнадцатиричная –  $a = 0, 1 \dots 9, A, B, C, D, E, F$

# Дополнительный код

Кодирование положительных чисел

+25 0 11001 прямой код

0 11001 дополнительный код

Кодирование отрицательных чисел  $-B = \bar{B} + 1$

-25 1 11001 – прямой код

а) 1 00110 – обратный код

б) 1 00110

+ 0 00001

1 00111 – дополнительный код

Кодирование отрицательных чисел

а) 1 00111 – дополнительный код

б) 1 11000

в) 1 11000

+ 0 00001

1 11001 – прямой код

8 -25 = -17

0 01000

1 10000

+ 1 00111

+ 0 00001

1 01111 – дополнительный код

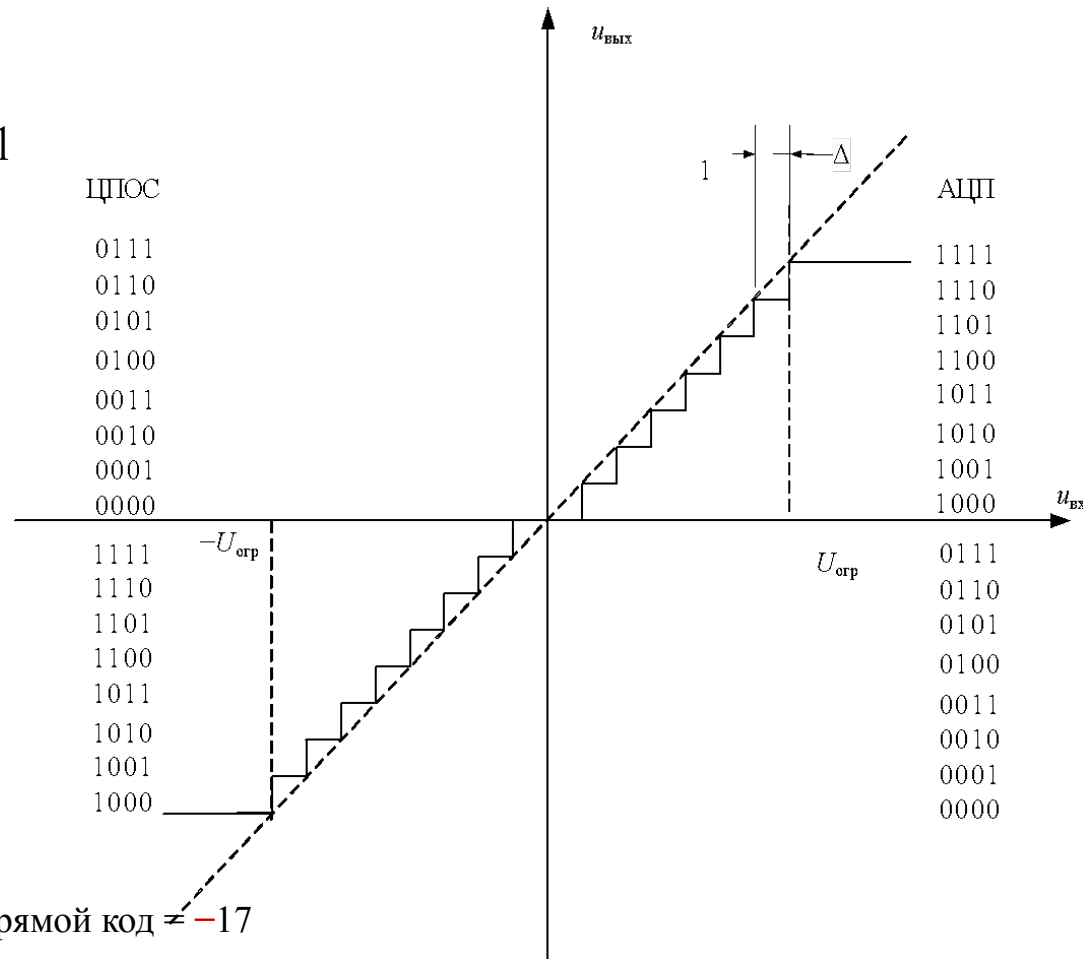
1 10001 – прямой код  $\neq -17$

31 -25 = +6

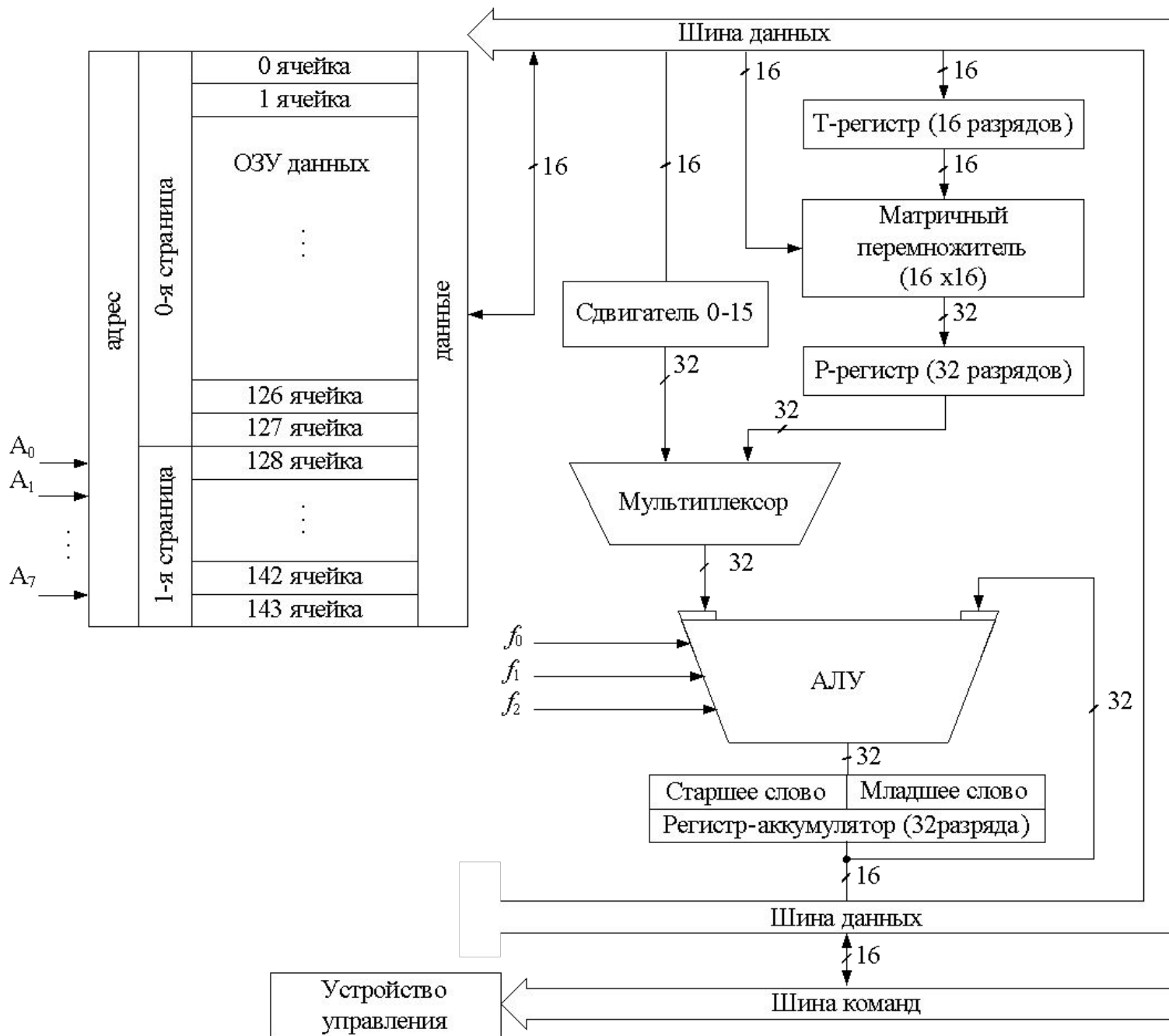
0 11111

+ 1 00111

1| 0 00110 – дополнительный/прямой код



# Сигнальный процессор TMS 32010

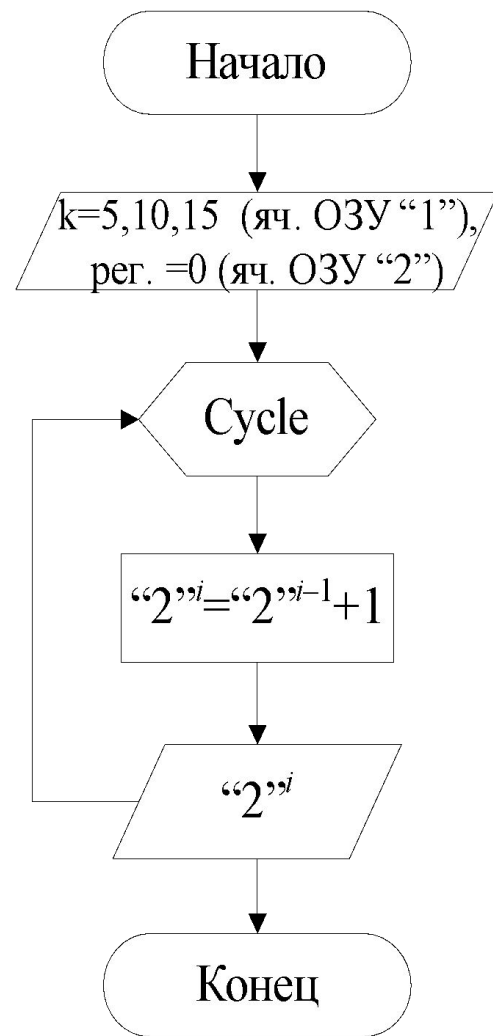
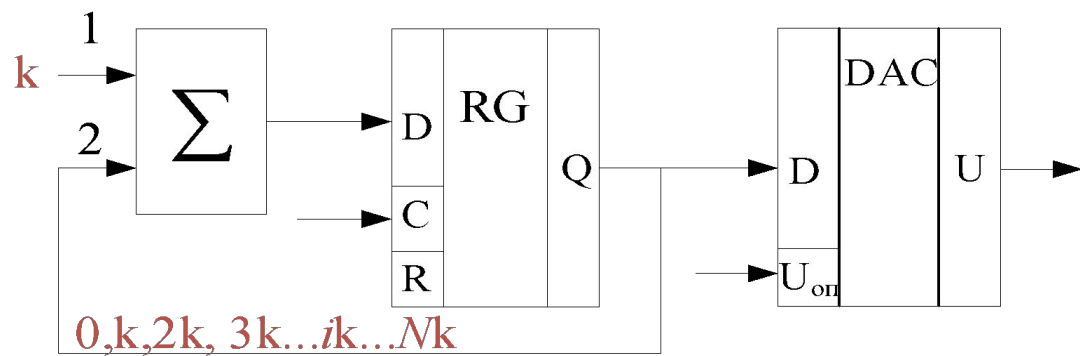




# Команды сигнального процессоров

- ADD** Прибавить к содержимому аккумулятора со сдвигом
- AND** Логическое умножение с содержимым аккумулятора
- APAC** Прибавить содержимое регистра Р к содержимому аккумулятора
- B** Безусловный переход
- BNZ** Переход при значении содержимого аккумулятора  $\neq 0$
- BZ** Переход при значении содержимого аккумулятора  $= 0$
- IN** Считать данные из порта
- LAC** Загрузить в аккумулятор со сдвигом
- LACK** Загрузить в аккумулятор константой
- LDPK** Загрузить в указатель на страницу памяти (константа)
- LT** Загрузить в Т-регистр
- MAC** Выполнить умножение и сохранить результат операции в аккумуляторе
- MPY** Выполнить умножение
- MPYK** Выполнить умножение с константой
- OR** Логическое «ИЛИ» с содержимым аккумулятора
- OUT** Вывести данные в порт
- PAC** Загрузить в аккумулятор содержимое Р-регистра
- SACH** Сохранить старшее слово аккумулятора со сдвигом
- SACL** Сохранить младшее слово аккумулятора со сдвигом
- SPAC** Вычесть содержимое Р-регистра из аккумулятора
- SUB** Вычесть из содержимого аккумулятора со сдвигом
- SUBK** Вычесть константу из содержимого аккумулятора
- TBLR** Чтение таблицы
- TBLW** Запись таблицы
- XOR** Выполнить исключительное «ИЛИ» с содержимым аккумулятора
- ZAC** Обнулить аккумулятор

# Формирование пилообразного напряжения методом работы накапливающего сумматора



.text

ldpk 0

lack 5 (10,15)

sac1 1

zac

Cycle lac 2

add 1

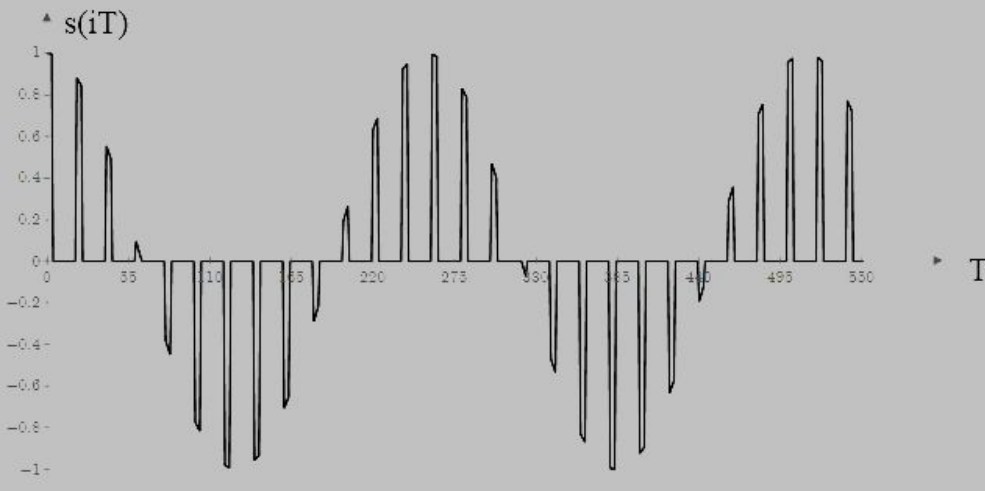
sac1 2

out 2,pa0

b Cycle

.end

# Формирование синусоидального сигнала



```
Cycle    LAC Fi
ADD DelFi
SACL Fi
LACK 0FFh
ADD One,8
AND Fi
SACL Fi
LAC One,7
ADD Fi
TBLR SinFi
LAC One,15
XOR SinFi
SACL SinFi
OUT SinFi, PA0
B Cycle
.end
```

- One .set 1
- DelFi .set 2
- Fi .set 3
- SinFi .set 4
- .text
- LDPK 0
- LACK 1
- SACL One
- LACK 30
- SACL DelFi
- ZAC
- SACL Fi

# Формирование сигнала с частотной модуляцией

- \*fm.asm
- One .set 1
- DelFi .set 2
- Fi .set 3
- SinFi .set 4
- Inf .set 5
- Count .set 6
- .text
- LDPK 0
- LACK 1
- SACL One
- A LAC One,7
- SACL Count
- LAC Inf
- AND One
- XOR One
- SACL Inf
- BNZ F
- LACK 8
- SACL DelFi
- B LC
- FLACK 16

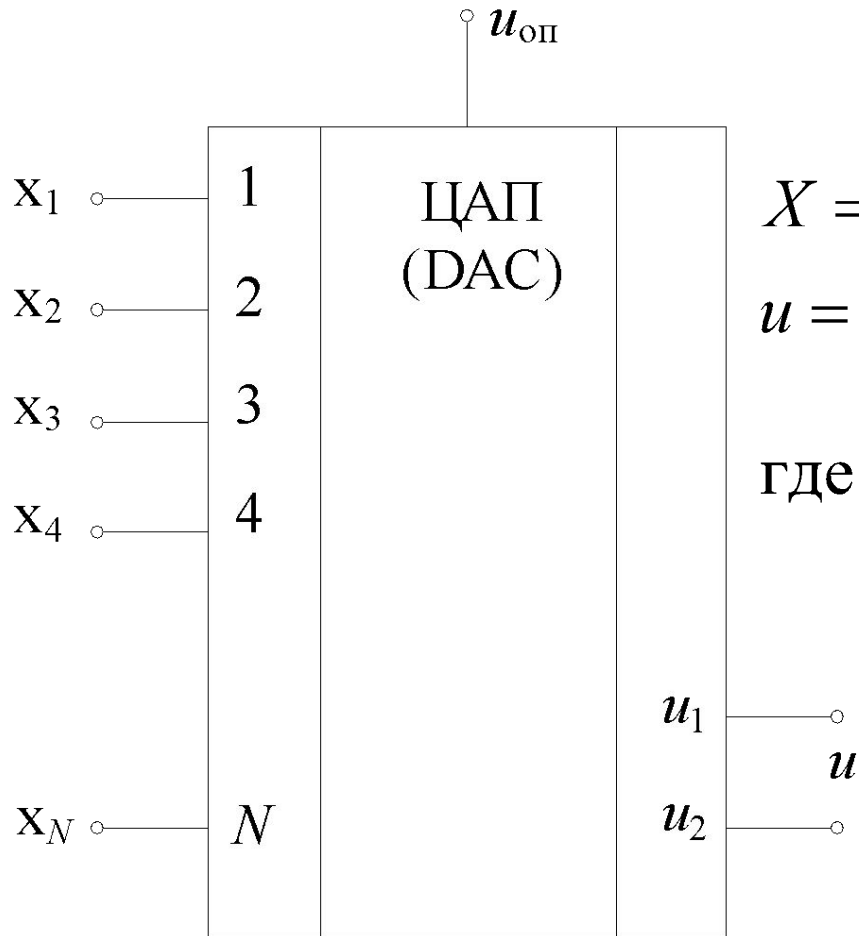
```
SACL DelFi
LC LAC Fi
ADD DelFi
SACL Fi
LACK 11111111B
ADD One,8
AND Fi
ADD One,7
TBLR SinFi
LAC One,15
XOR SinFi
SACL SinFi
OUT SinFi,PA0
LAC Count
SUB One
SACL Count
BNZ LC
B A
.end
```

# Формирование сигнала с частотной модуляцией

- \*phm.asm
- One .set 1
- DelFi .set 2
- Fi .set 3
- SinFi .set 4
- Inf .set 5
- Inf0 .set 6
- Count .set 7
- .text
- LDPK 0
- LACK 1
- SACL One
- LACK 8
- SACL DelFi
- A LAC One,7
- SACL Count
- LAC Inf
- AND One
- XOR One
- SACL Inf
- BZ F
- 

```
LAC Fi
ADD One,8
SACL Fi
B LC
F LAC Fi
SUB One,8
SACL Fi
LC LAC Fi
ADD DelFi
SACL Fi
LACK 11111111B
ADD One,8
AND Fi
ADD One,7
TBLR SinFi
LAC One,15
XOR SinFi
SACL SinFi
OUT SinFi,PA0
LAC Count
SUB One
SACL Count
BNZ LC
B A
.end
```

# Цифро-аналоговый преобразователь

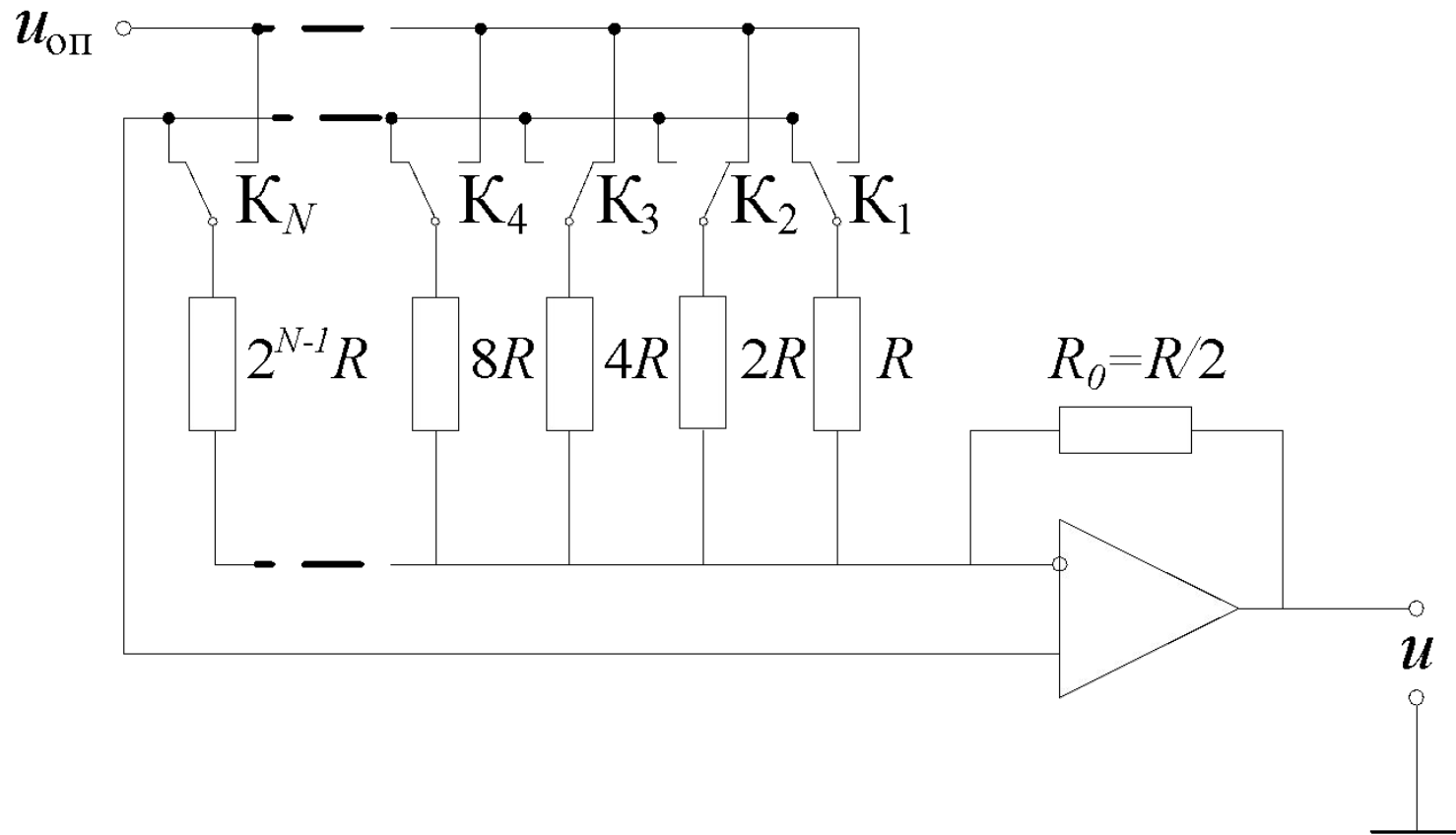


$$X = \{x_1, x_2, x_3, \dots, x_N\},$$

$$u = u_{\text{оп}} D,$$

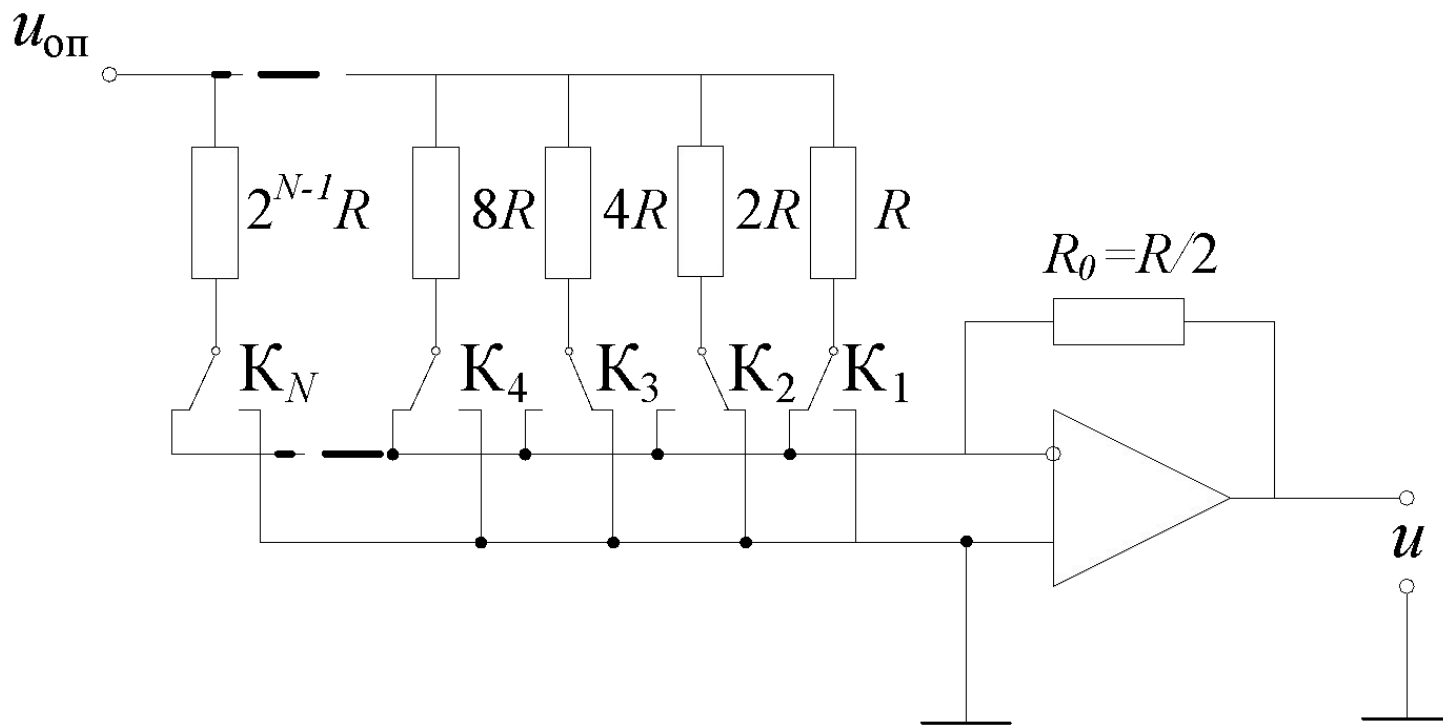
$$\text{где } D = \frac{x_1}{2} + \frac{x_2}{2^2} + \frac{x_3}{2^3} + \dots + \frac{x_N}{2^N}, \quad x_n = \begin{cases} 0 \\ 1 \end{cases}$$

# Цифро-аналоговый преобразователь с коммутацией напряжений

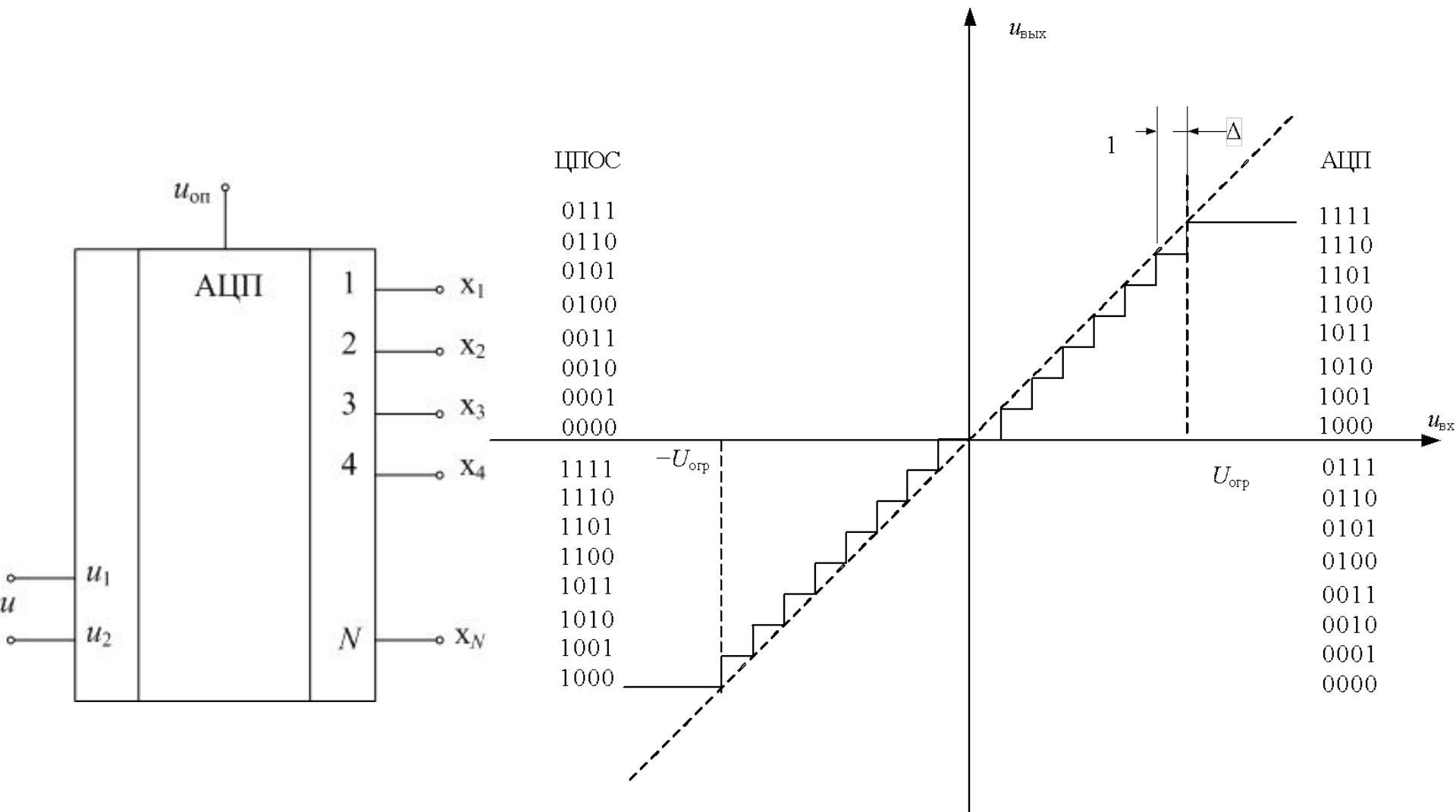




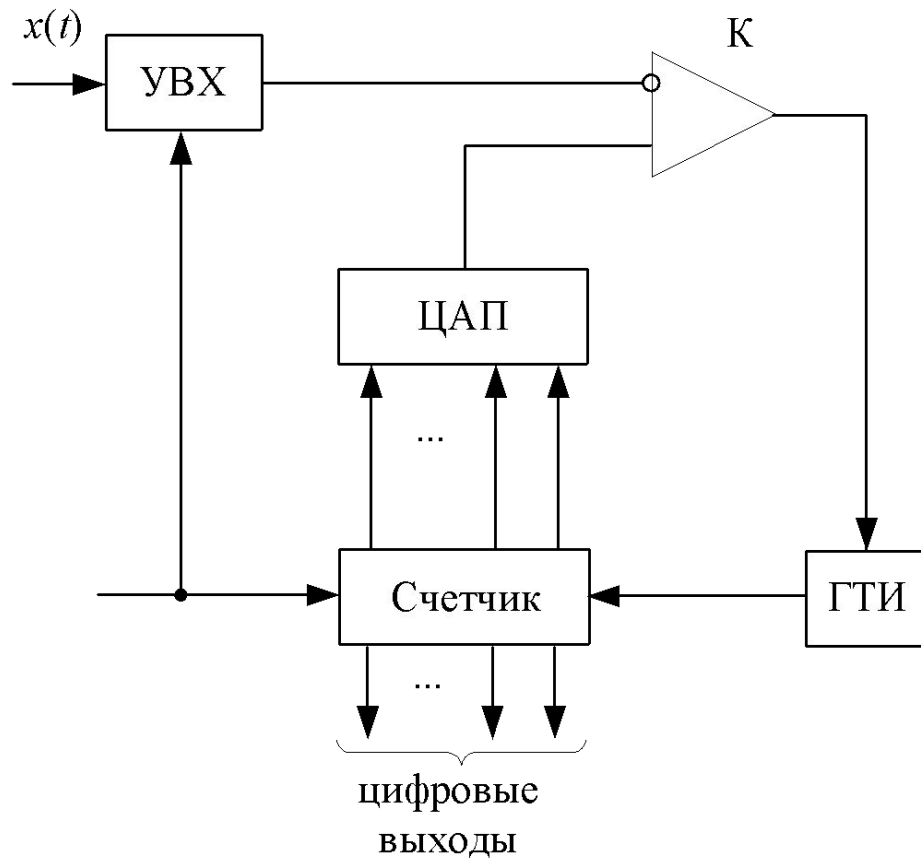
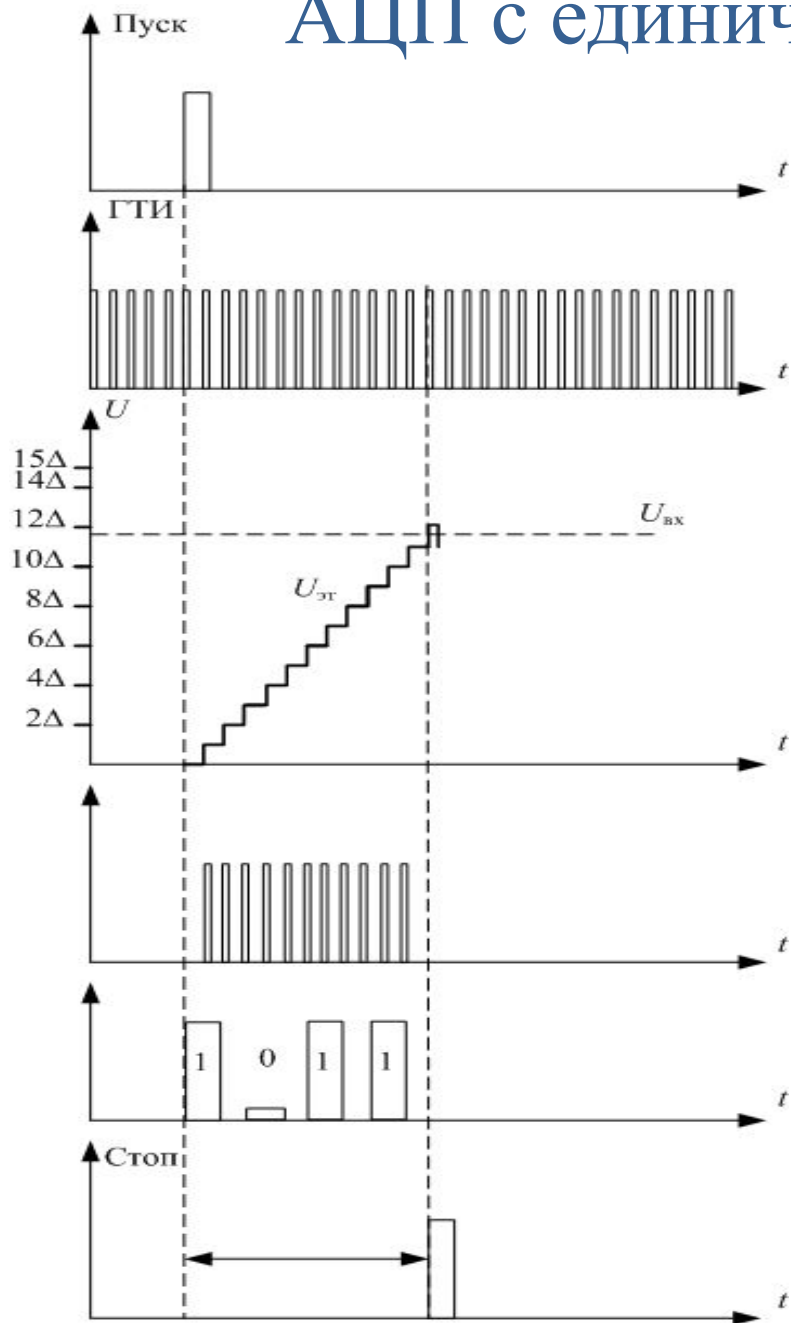
# Цифро-аналоговый преобразователь с КОММУТАЦИЕЙ ТОКОВ



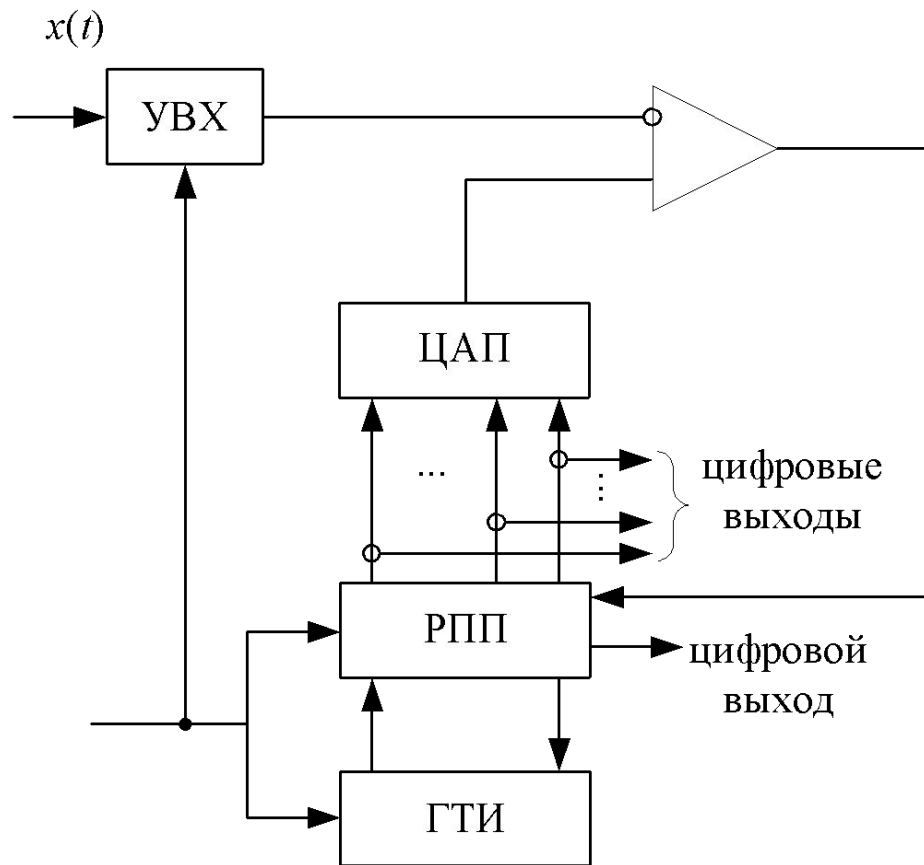
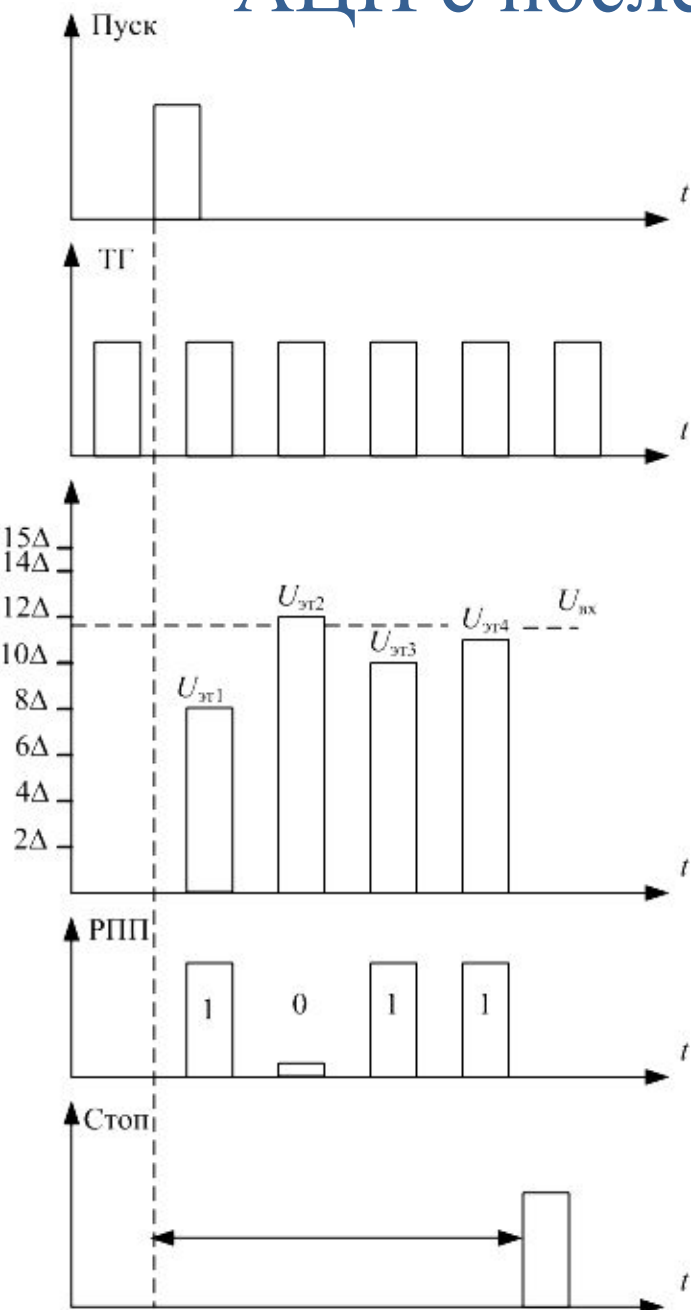
# Аналого-цифровой преобразователь



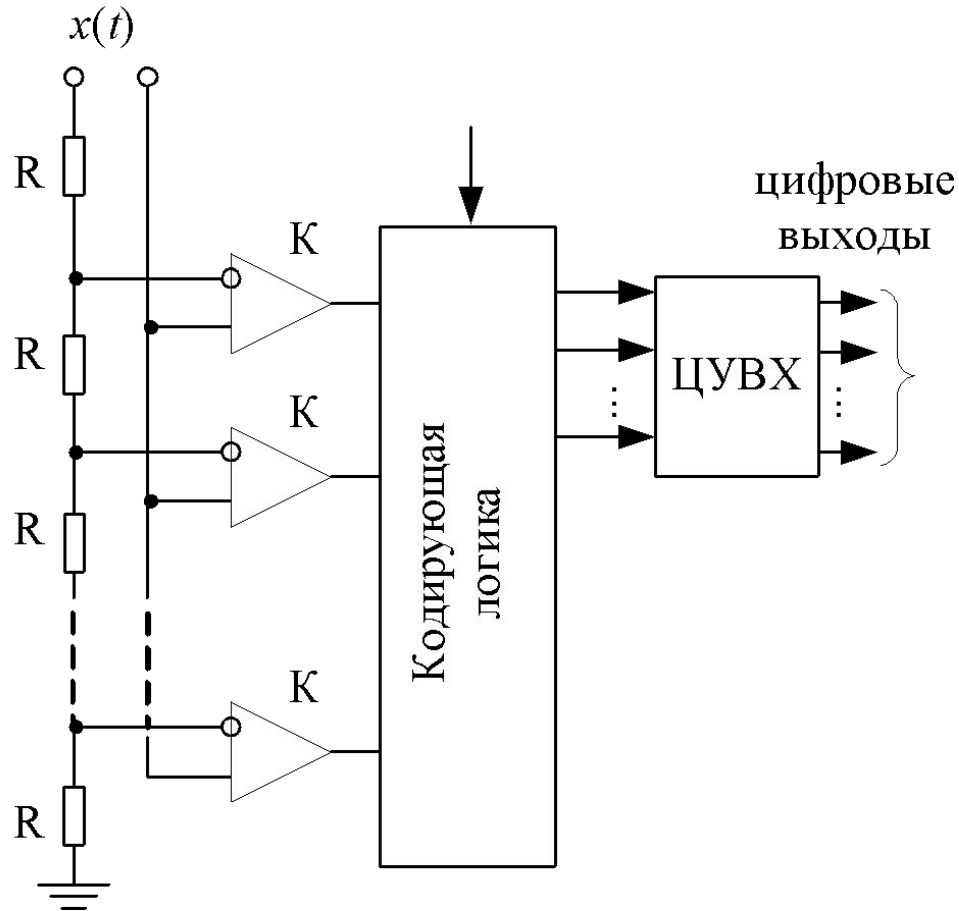
# АЦП с единичными приближениями



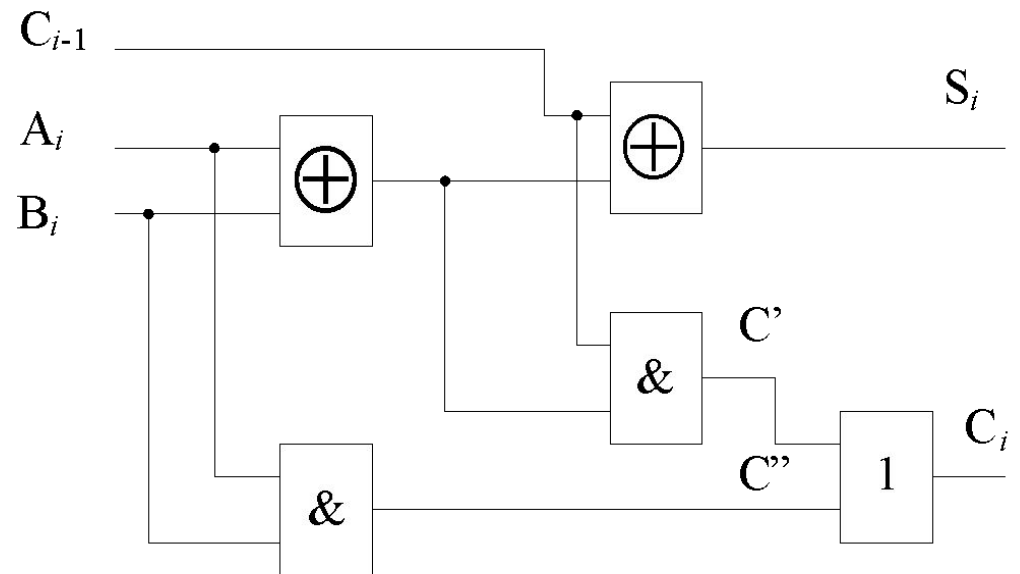
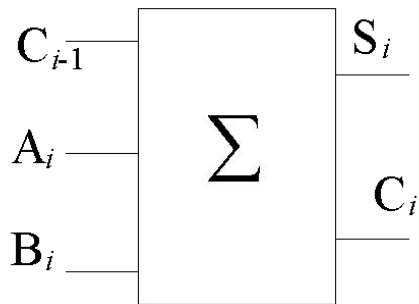
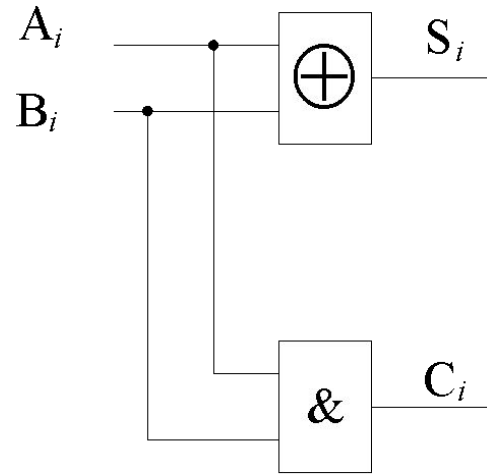
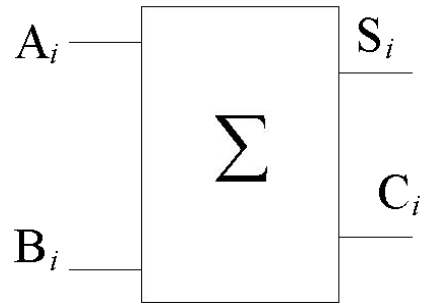
# АЦП с последовательных приближений



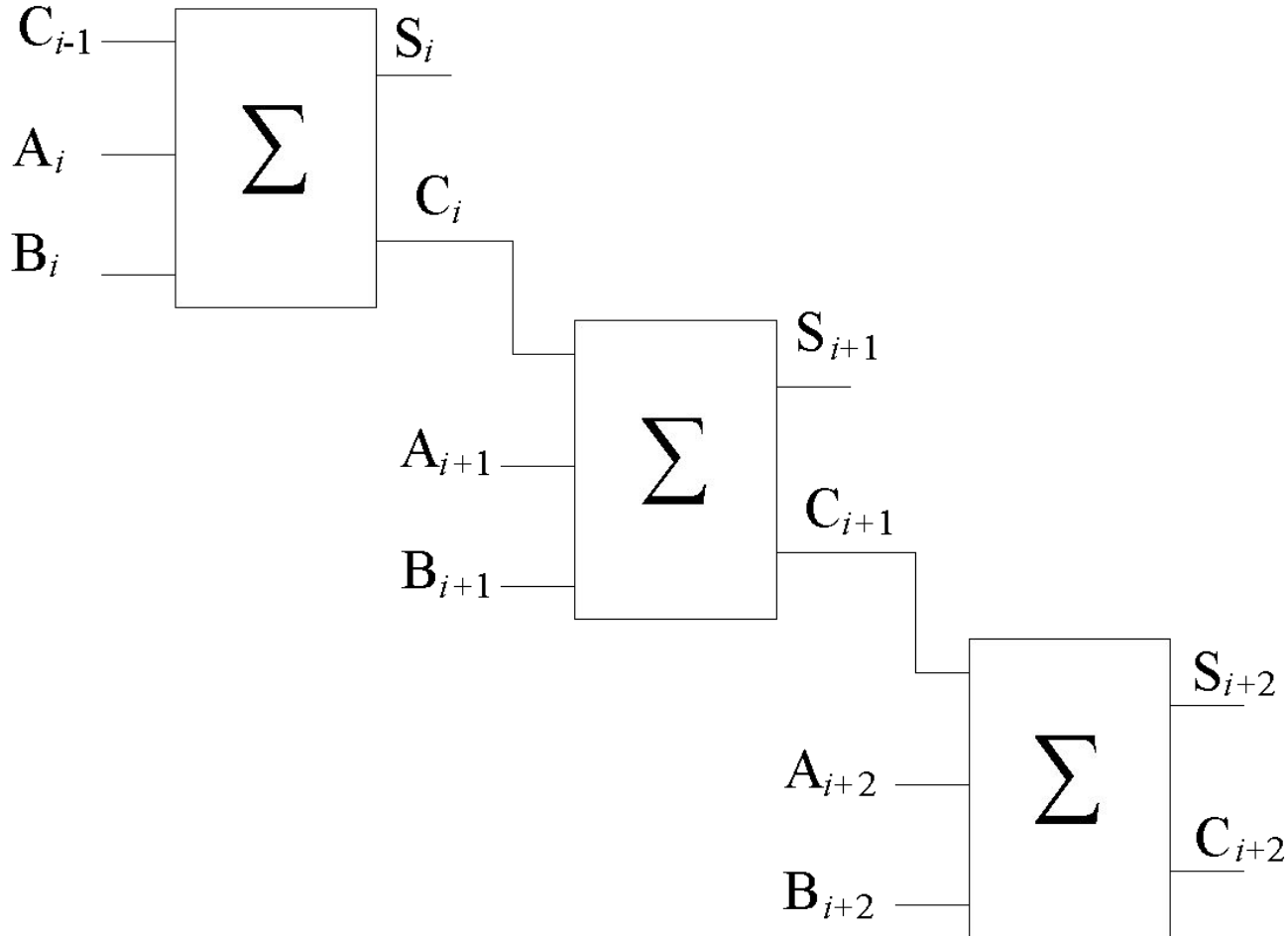
# Аналого-цифровой преобразователь непосредственного сравнения



# Устройства арифметического сложения

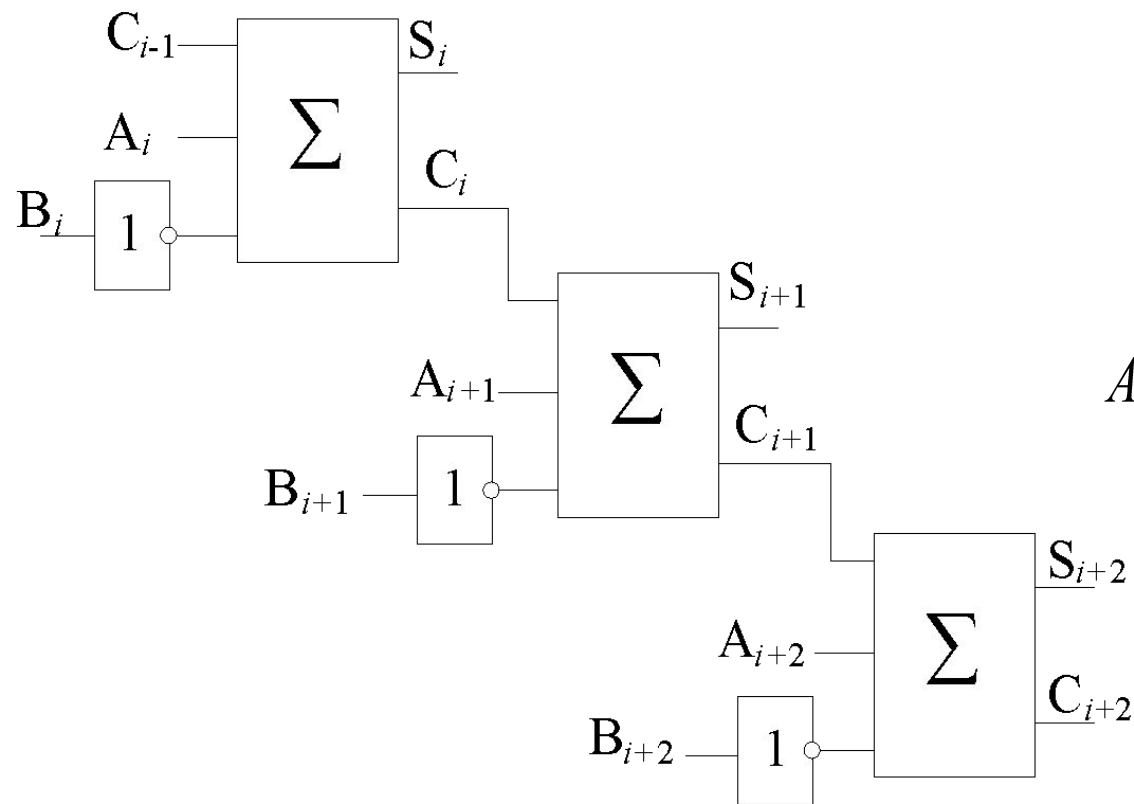


$A_i$	0	0	1	0	0	1	1	1
$B_i$	0	0	0	1	1	0	1	1
$C_{i-1}$	0	1	0	0	1	1	0	1
$S_i$	0	1	1	0	0	0	0	1
$C_i$	0	0	0	1	1	1	1	1



# Устройство вычитания

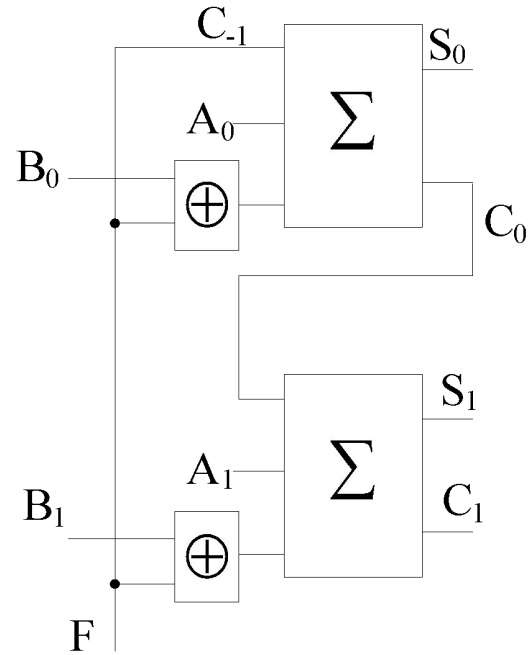
$A_i$	0	0	1	0	0	1	1	1
$B_i$	0	0	0	1	1	0	1	1
$C_{i-1}$	0	1	0	0	1	1	0	1
$S_i$	1	0	0	0	1	1	1	0
$C_i$	0	1	1	0	0	1	0	1



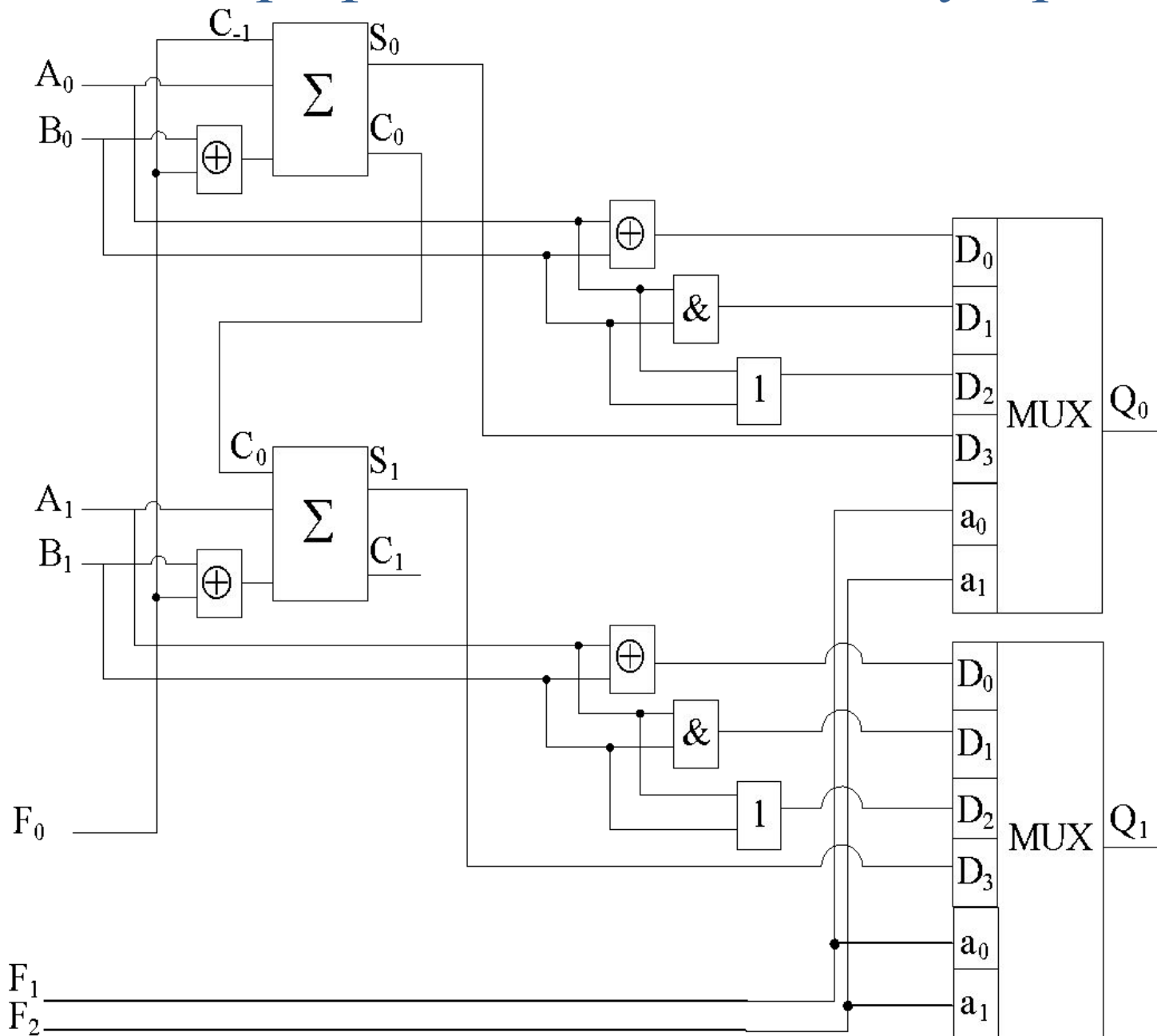
$$A - B = A + (-B) = A + \bar{B} + 1$$



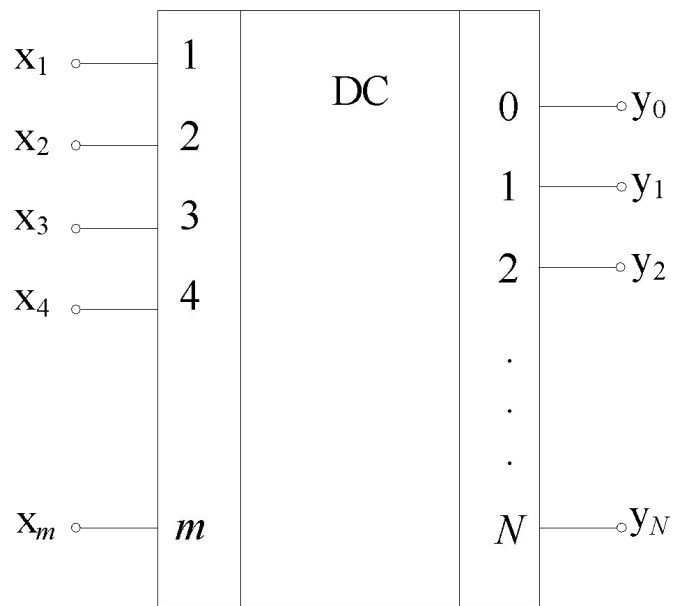
# Универсальное арифметическое устройство



# Арифметико-логическое устройство



# Дешифратор



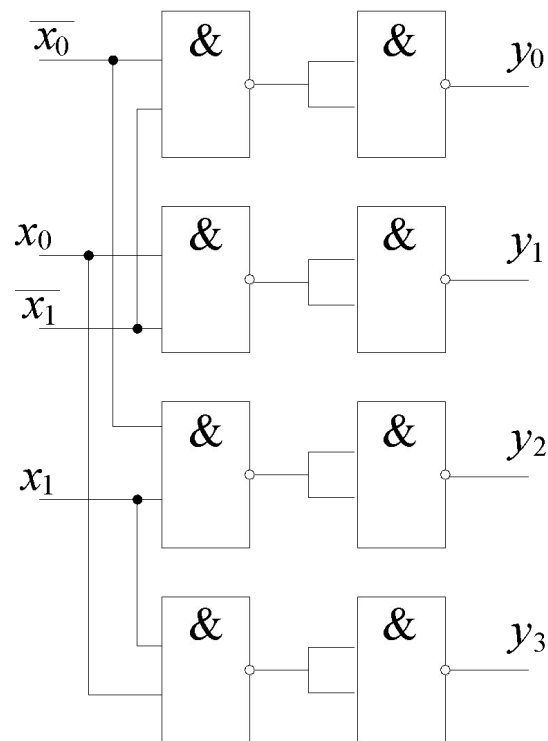
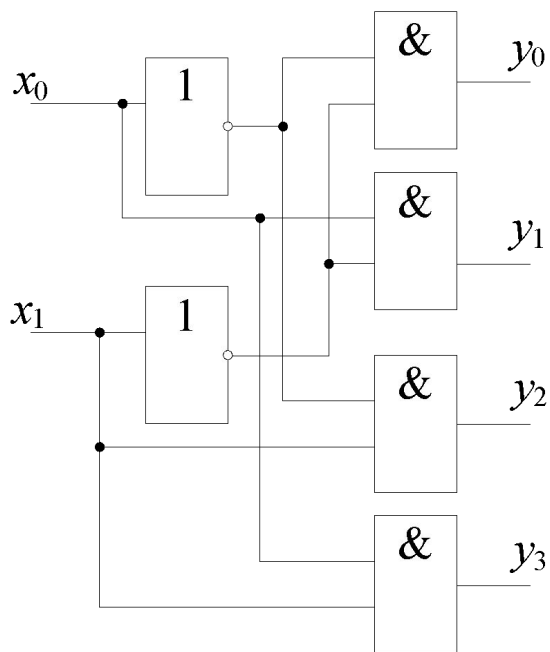
$$N + 1 = 2^m$$

$$y_0 = \bar{x}_1 \bar{x}_2 = \overline{\overline{\bar{x}_1 \bar{x}_2}}$$

$$y_1 = x_1 \bar{x}_2 = \overline{\overline{x_1 \bar{x}_2}}$$

$$y_2 = \bar{x}_1 x_2 = \overline{\overline{\bar{x}_1 x_2}}$$

$$y_3 = x_1 x_2 = \overline{\overline{x_1 x_2}}$$



$$y'_0 = \bar{x}_1 \bar{x}_2 \bar{x}_3 = \overline{\overline{x_1 x_2 x_3}}$$

$$y'_1 = x_1 \bar{x}_2 \bar{x}_3 = \overline{\overline{x_1 x_2 x_3}}$$

$$y'_2 = \bar{x}_1 x_2 \bar{x}_3 = \overline{\overline{x_1 x_2 x_3}}$$

⊠

$$y'_7 = x_1 x_2 x_3 = \overline{\overline{x_1 x_2 x_3}}$$

$$y''_0 = \bar{x}_4 \bar{x}_5 \bar{x}_6 = \overline{\overline{x_4 x_5 x_6}}$$

$$y''_1 = x_4 \bar{x}_5 \bar{x}_6 = \overline{\overline{x_4 x_5 x_6}}$$

$$y''_2 = \bar{x}_4 x_5 \bar{x}_6 = \overline{\overline{x_4 x_5 x_6}}$$

⊠

$$y''_7 = x_4 x_5 x_6 = \overline{\overline{x_4 x_5 x_6}}$$

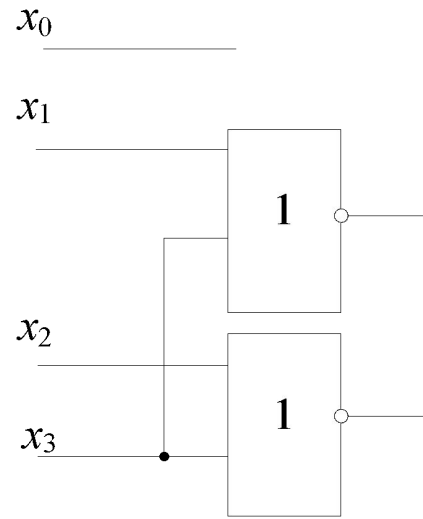
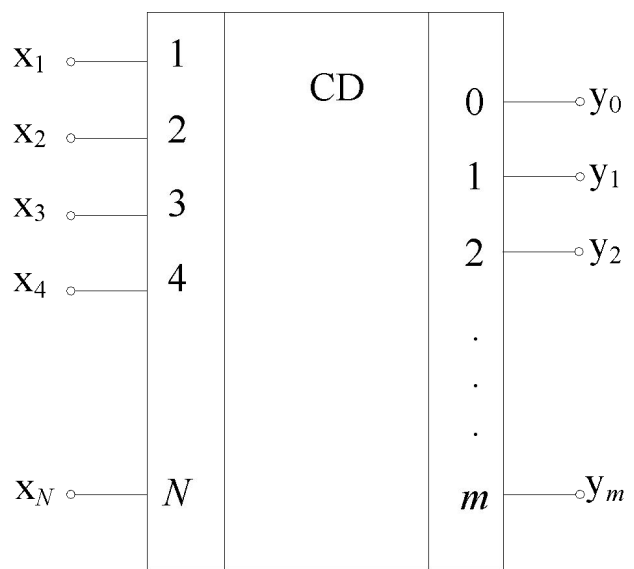
$$y_0 = y'_0 y''_0$$

$$y_1 = y'_1 y''_1$$

⊠

$$y_{63} = y'_7 y''_7$$

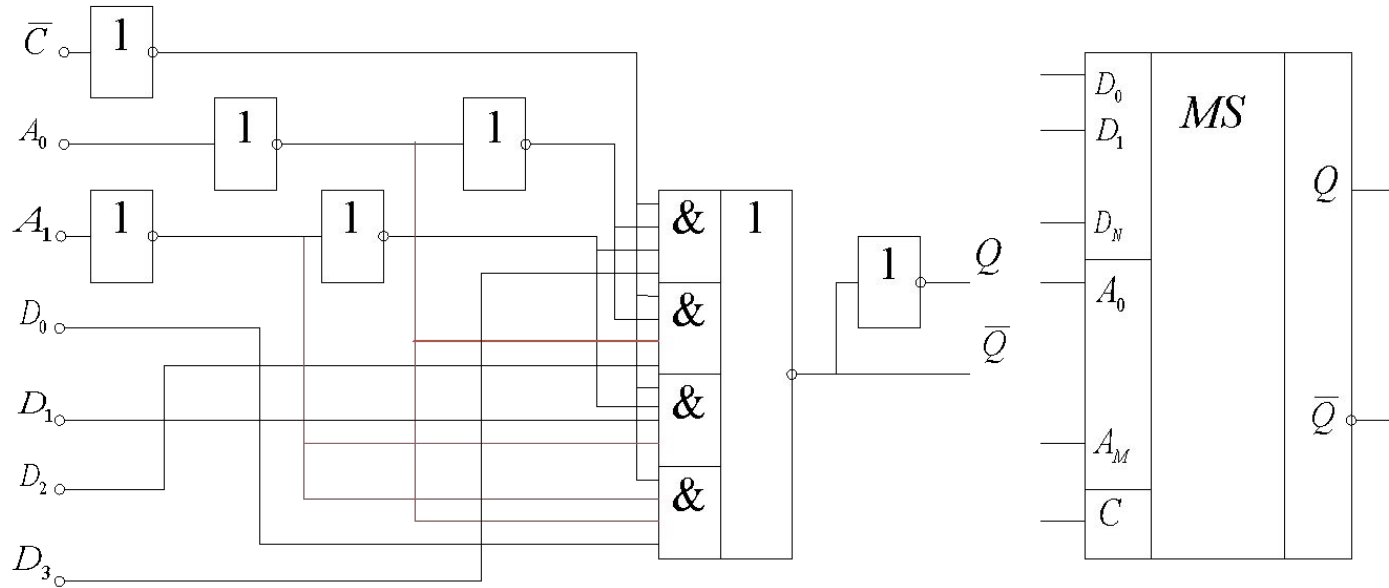
# Шифратор



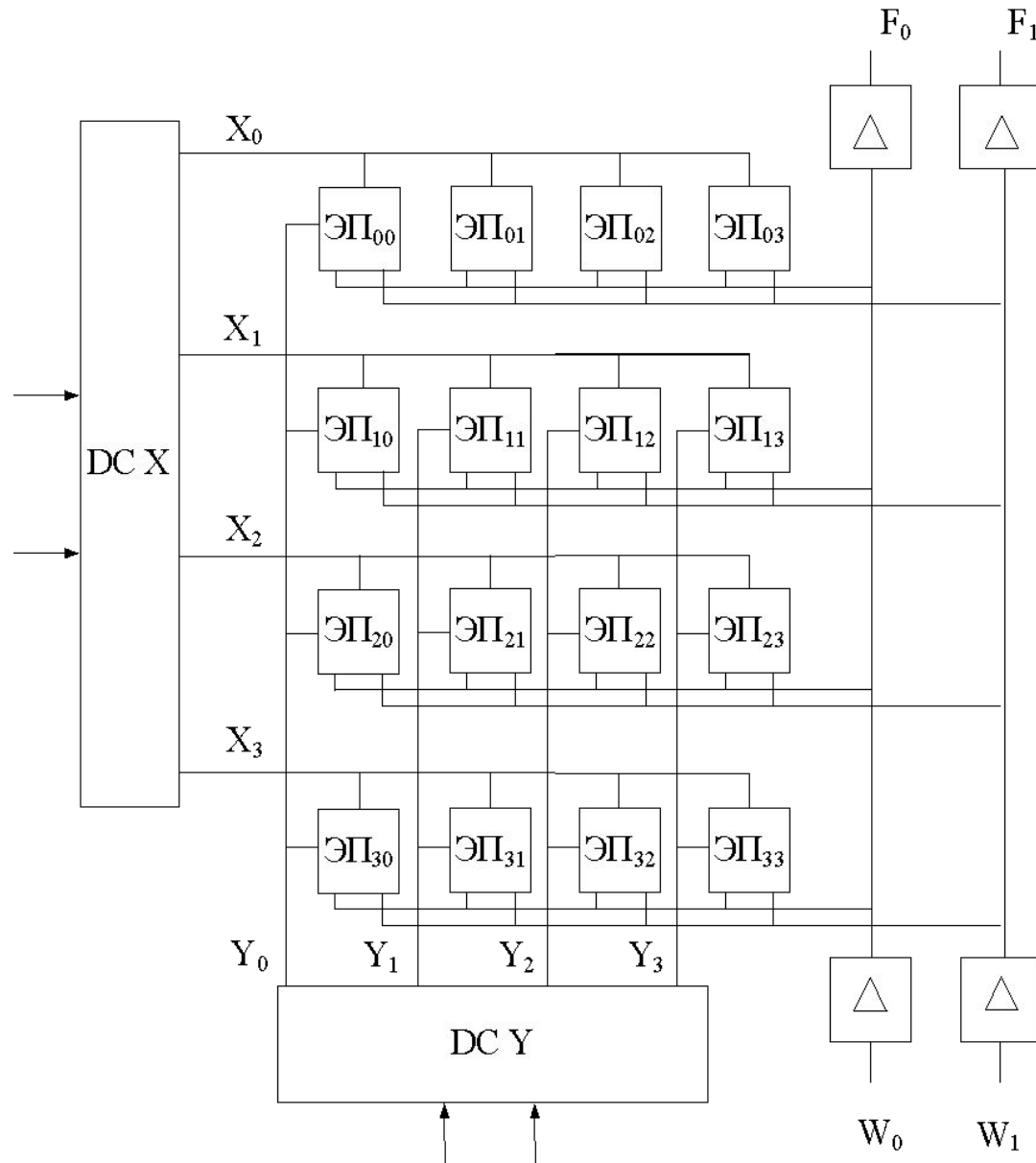
$$y_1 = x_1 + x_3$$

$$y_2 = x_2 + x_3$$

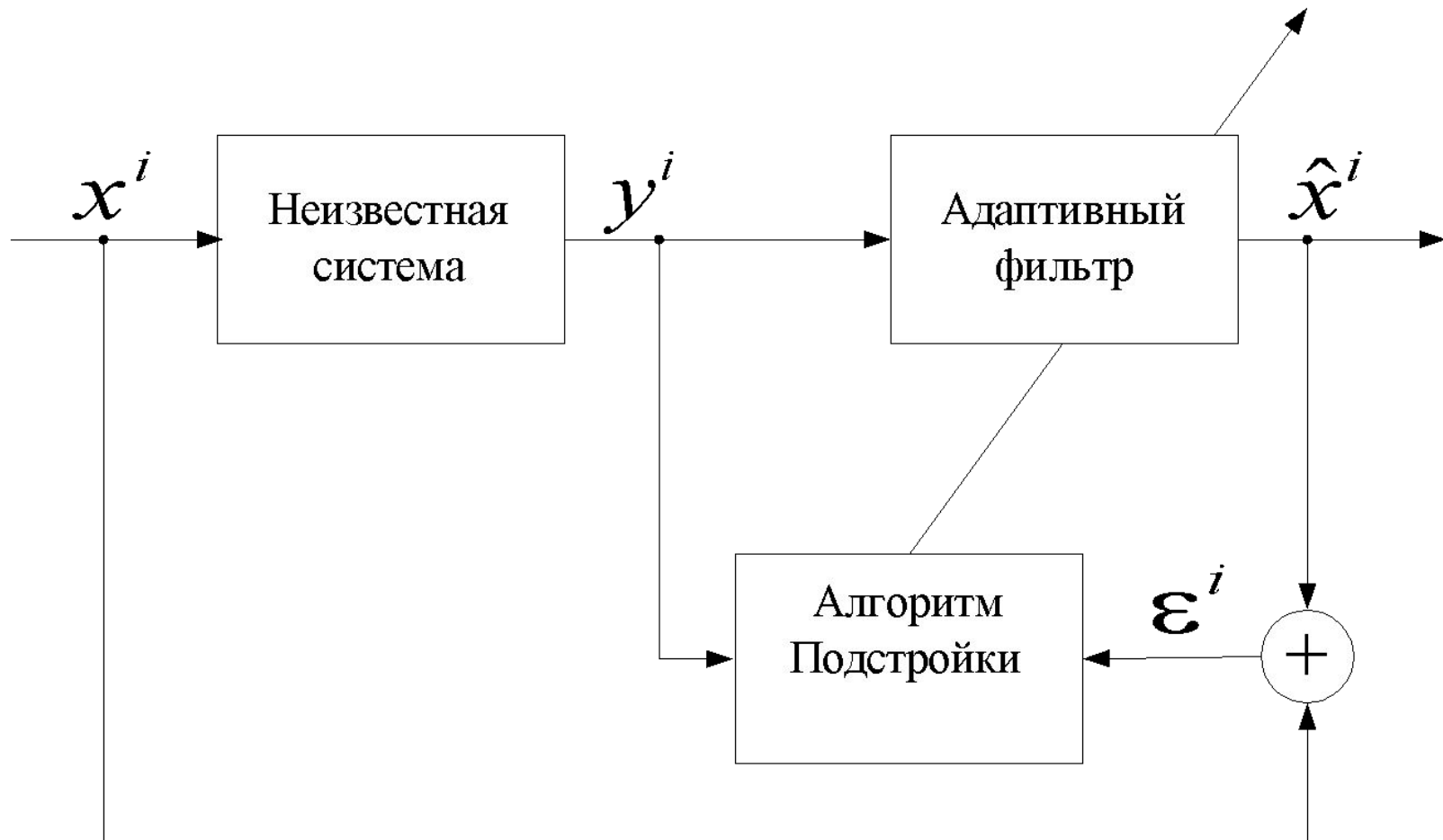
# Мультиплексор



# Запоминающие устройства



# АДАПТИВНАЯ СИСТЕМА



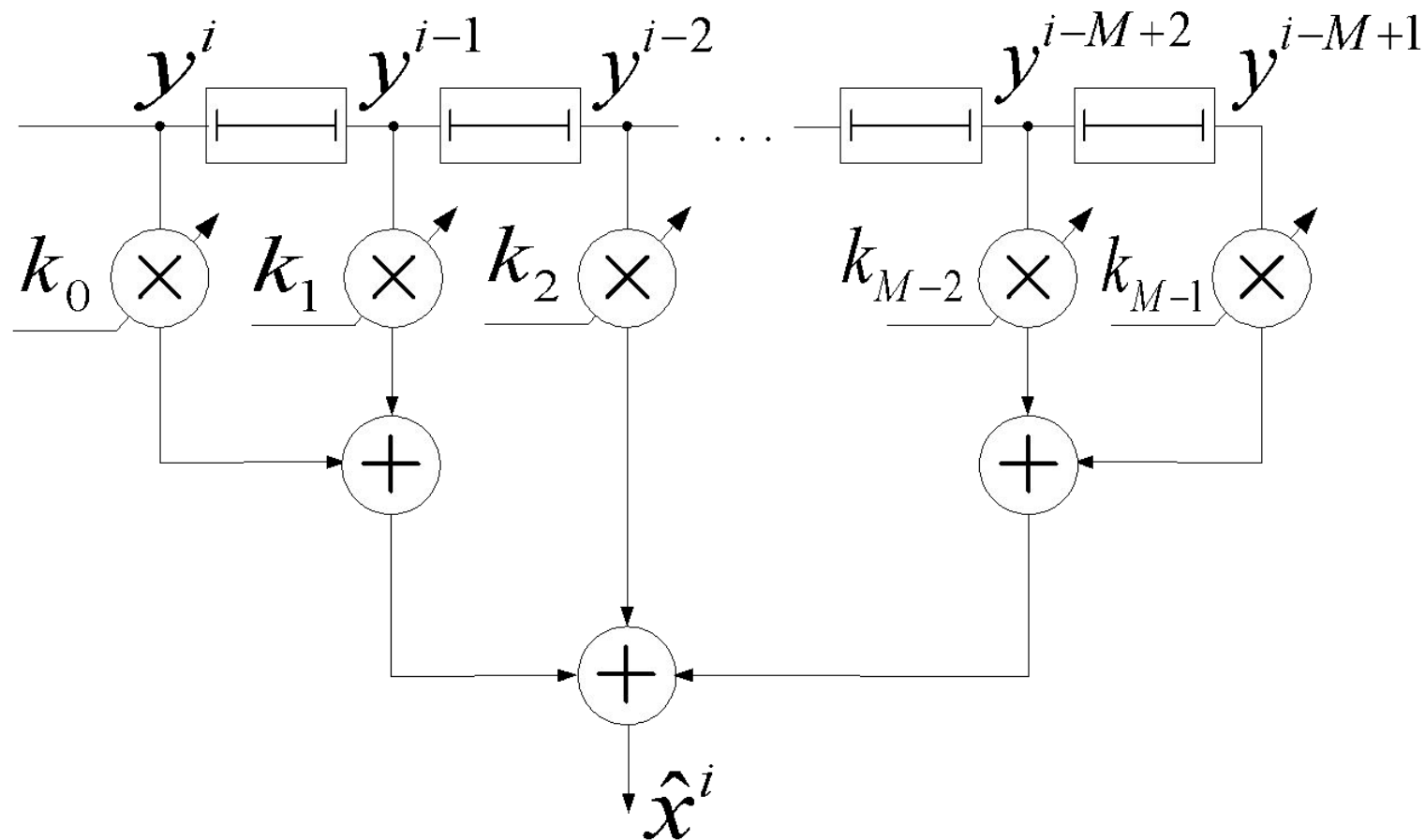
$$y^i = h^a \otimes x^i + n^i = \sum_{a=0}^{A-1} h^a x^{i-a} + n^i$$

$$\hat{x}^i = k_{\text{a.}\phi.}^m \otimes y^i = \sum_{m=0}^{M-1} k^m y^{i-m}$$

$$\varepsilon^i = x^i - \hat{x}^i$$

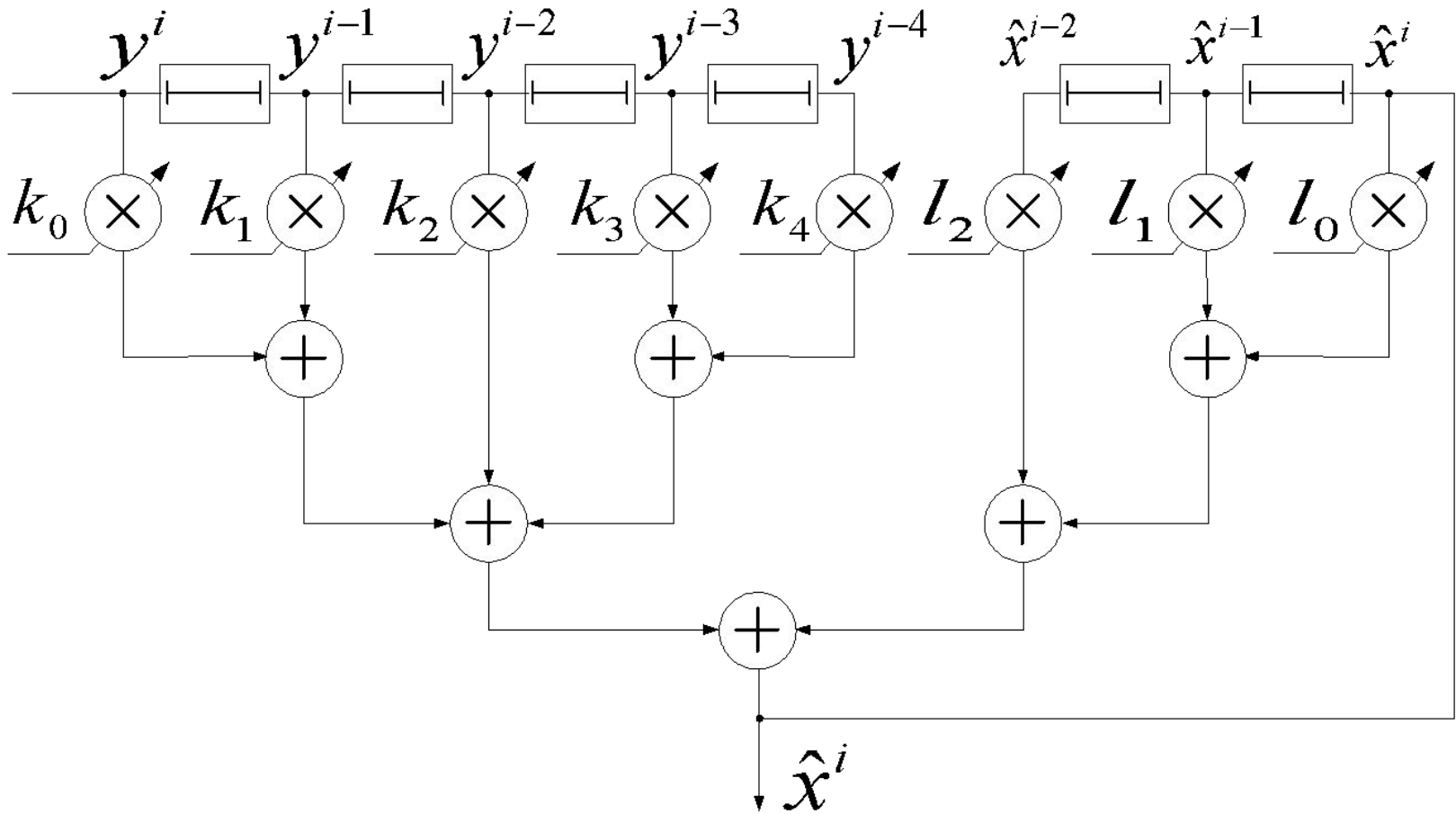


# Адаптивный нерекурсивный фильтр



$$\hat{x}^i = \sum_{m=0}^{M-1} k^m y^{i-m} = \mathbf{k} \mathbf{y}^i$$

# Адаптивный рекурсивный фильтр



$$\hat{x}^i = \sum_{m=0}^{M-1} k^m y^{i-m} - \sum_{n=0}^{N-1} l^n \hat{x}^{i-n} = \mathbf{k} \mathbf{y}^i - \mathbf{l} \hat{\mathbf{x}}^i$$

$$\boldsymbol{\varepsilon}^i = x^i - \hat{x}^i = x^i - \sum_{m=0}^{M-1} k^m x^{i-m} = x^i - (\mathbf{k}^i)^T \mathbf{y}^i$$

$$(\boldsymbol{\varepsilon}^i)^2 = (x^i)^2 + (\mathbf{k}^i)^T \mathbf{y}^i (\mathbf{y}^i)^T \mathbf{k}^i - 2x^i (\mathbf{y}^i)^T \mathbf{k}^i$$

$$E[(\boldsymbol{\varepsilon}^i)^2] = E[(x^i)^2] + (\mathbf{k}^i)^T E[\mathbf{y}^i (\mathbf{y}^i)^T] \mathbf{k}^i - 2E[x^i (\mathbf{y}^i)^T] \mathbf{k}^i$$

$$\xi = E[(\boldsymbol{\varepsilon}^i)^2] = E[(x^i)^2] + (\mathbf{k}^i)^T \mathbf{R} \mathbf{k}^i - 2\mathbf{P}^T \mathbf{k}^i$$

$$\nabla = \frac{\partial \xi}{\partial \mathbf{k}^i} = 2\mathbf{R}(\mathbf{k}^i) - 2\mathbf{P} \quad \nabla = 0 = 2\mathbf{R}\mathbf{k}^* - 2\mathbf{P}$$

$$\mathbf{k}^* = \mathbf{R}^{-1} \mathbf{P}$$

$$\xi_{\min} = E[(x^i)^2] - \mathbf{P}^T \mathbf{k}^*$$

## Другое представление градиента

$$\xi = \xi_{\min} + (\mathbf{k}^i - \mathbf{k}^*)^T \mathbf{R}(\mathbf{k}^i - \mathbf{k}^*)$$

$$\xi = \xi_{\min} + (\mathbf{v}^i)^T \mathbf{R} \mathbf{v}^i$$

$$\nabla = \frac{\partial \xi}{\partial \mathbf{v}^i} = 2\mathbf{R} \mathbf{v}^i$$

## Декорреляция сигнала ошибки и ВХОДНОГО СИГНАЛА

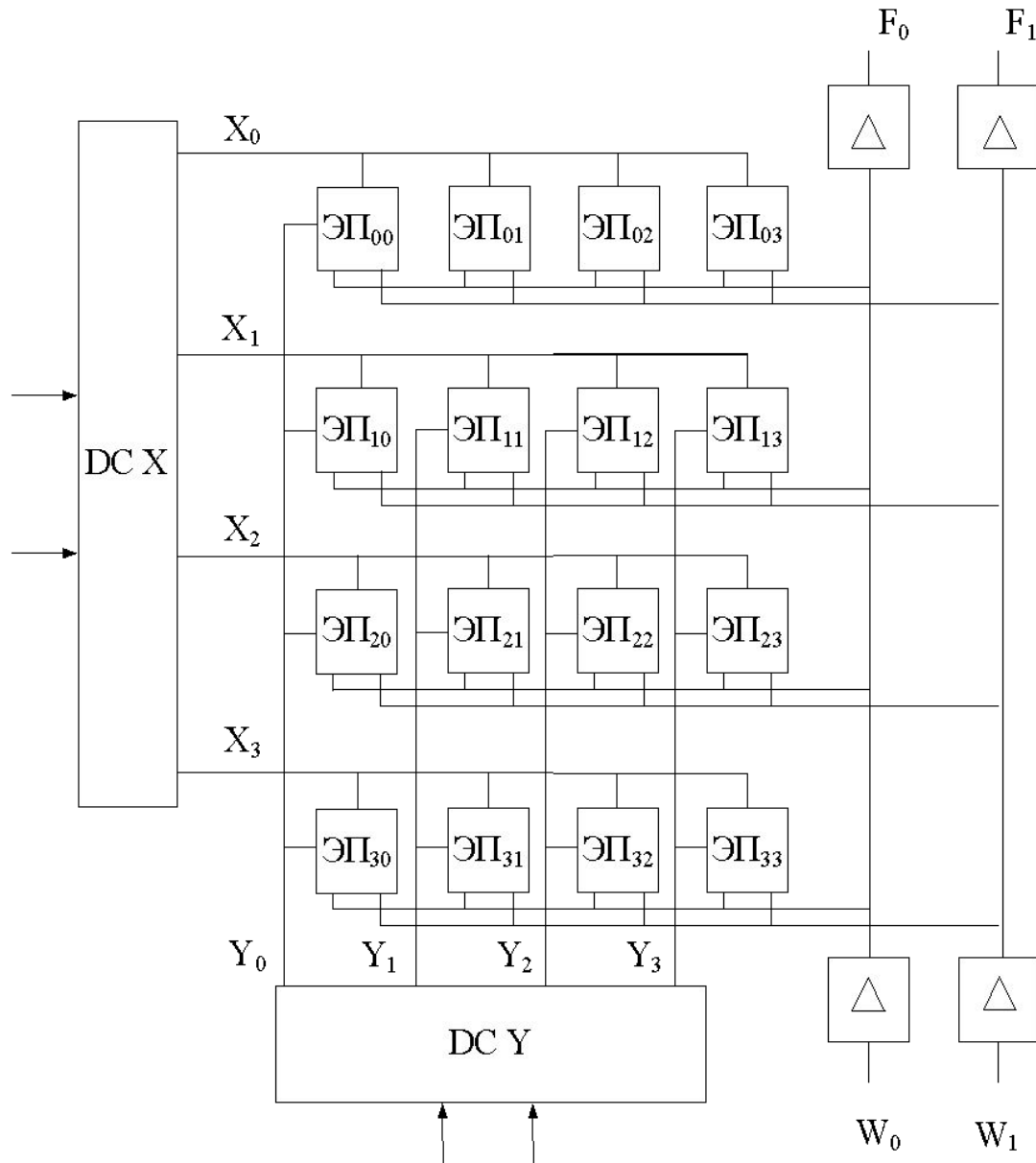
$$\mathbf{k}^i = \mathbf{k}^*$$

$$\varepsilon^i = x^i - (\mathbf{k}^i)^T \mathbf{y}^i$$

$$\varepsilon^i \mathbf{y}^i = x^i \mathbf{y}^i - \mathbf{y}^i (\mathbf{y}^i)^T \mathbf{k}^i$$

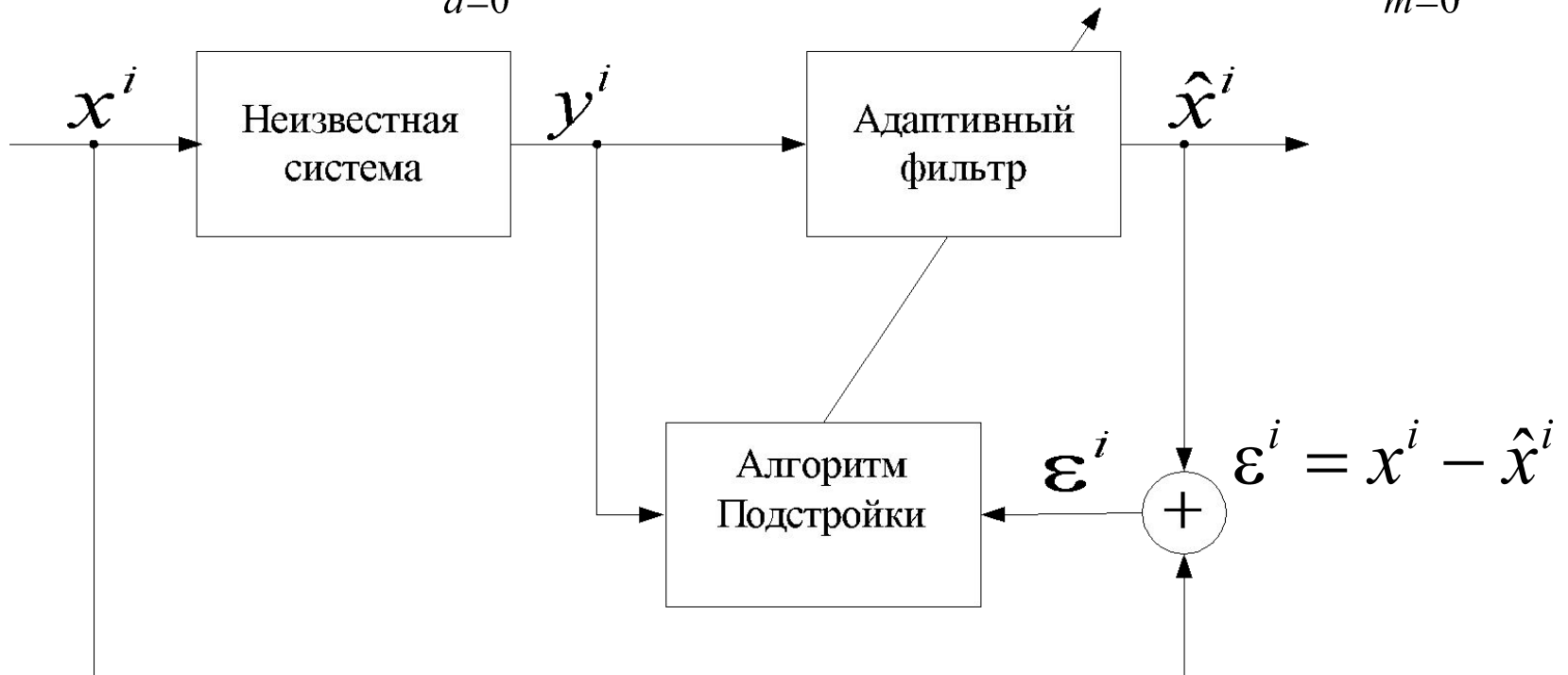
$$E[\varepsilon^i \mathbf{y}^i] = E[x^i \mathbf{y}^i] - E[\mathbf{y}^i (\mathbf{y}^i)^T] \mathbf{k}^i$$

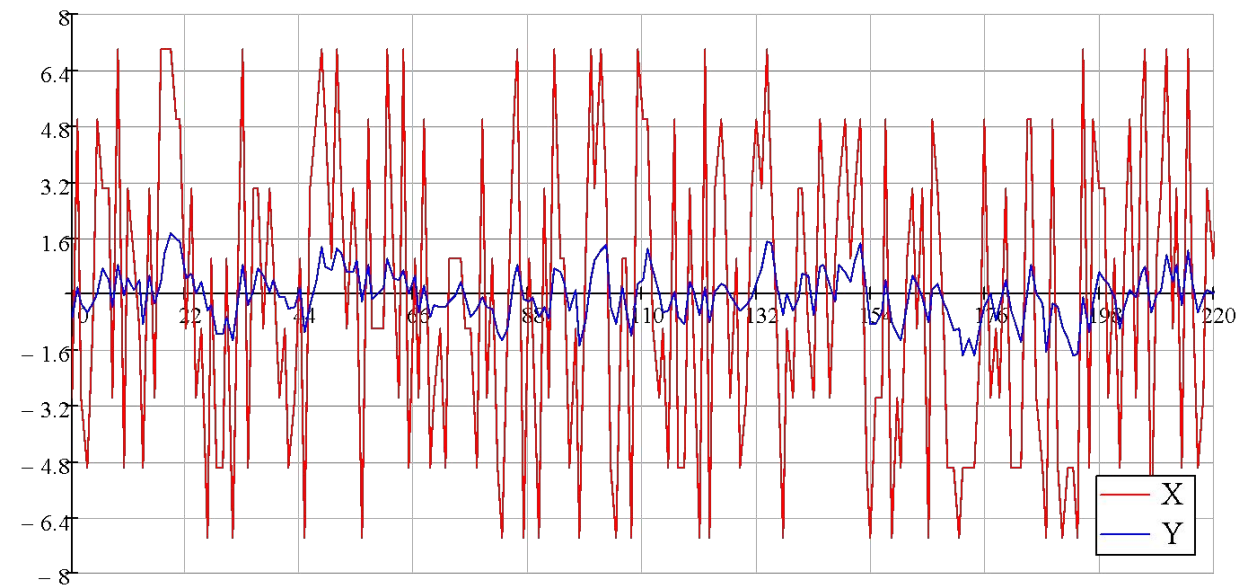
# Запоминающие устройства



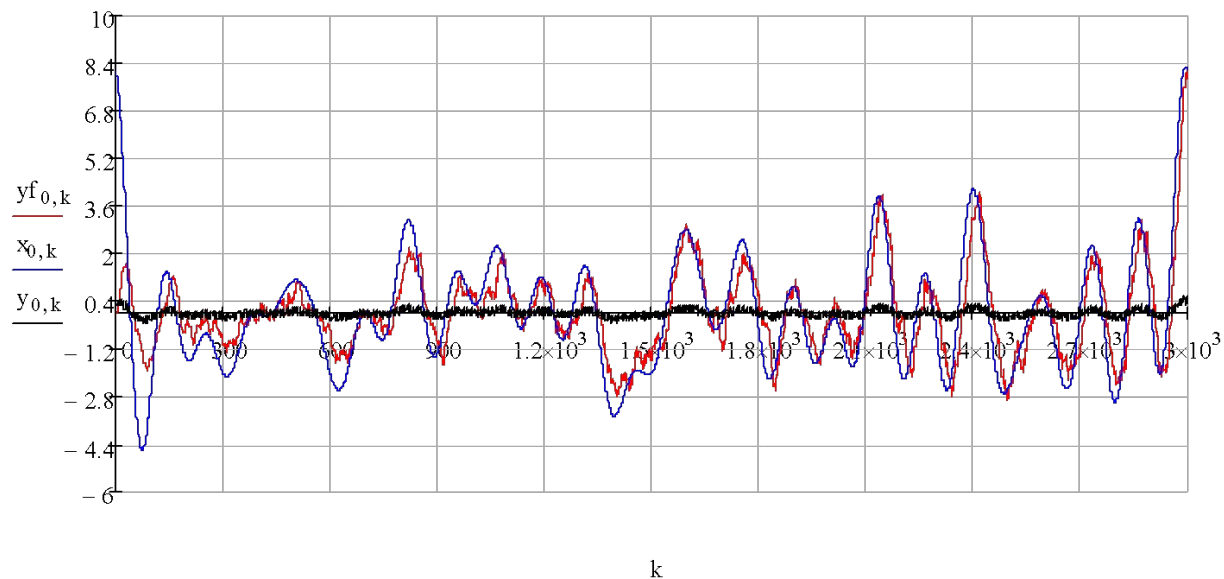
# АДАПТИВНАЯ СИСТЕМА

$$y^i = h^a \otimes x^i + n^i = \sum_{a=0}^{A-1} h^a x^{i-a} + n^i \quad \hat{x}^i = k_{\text{а.ф.}}^m \otimes y^i = \sum_{m=0}^{M-1} k^m y^{i-m}$$



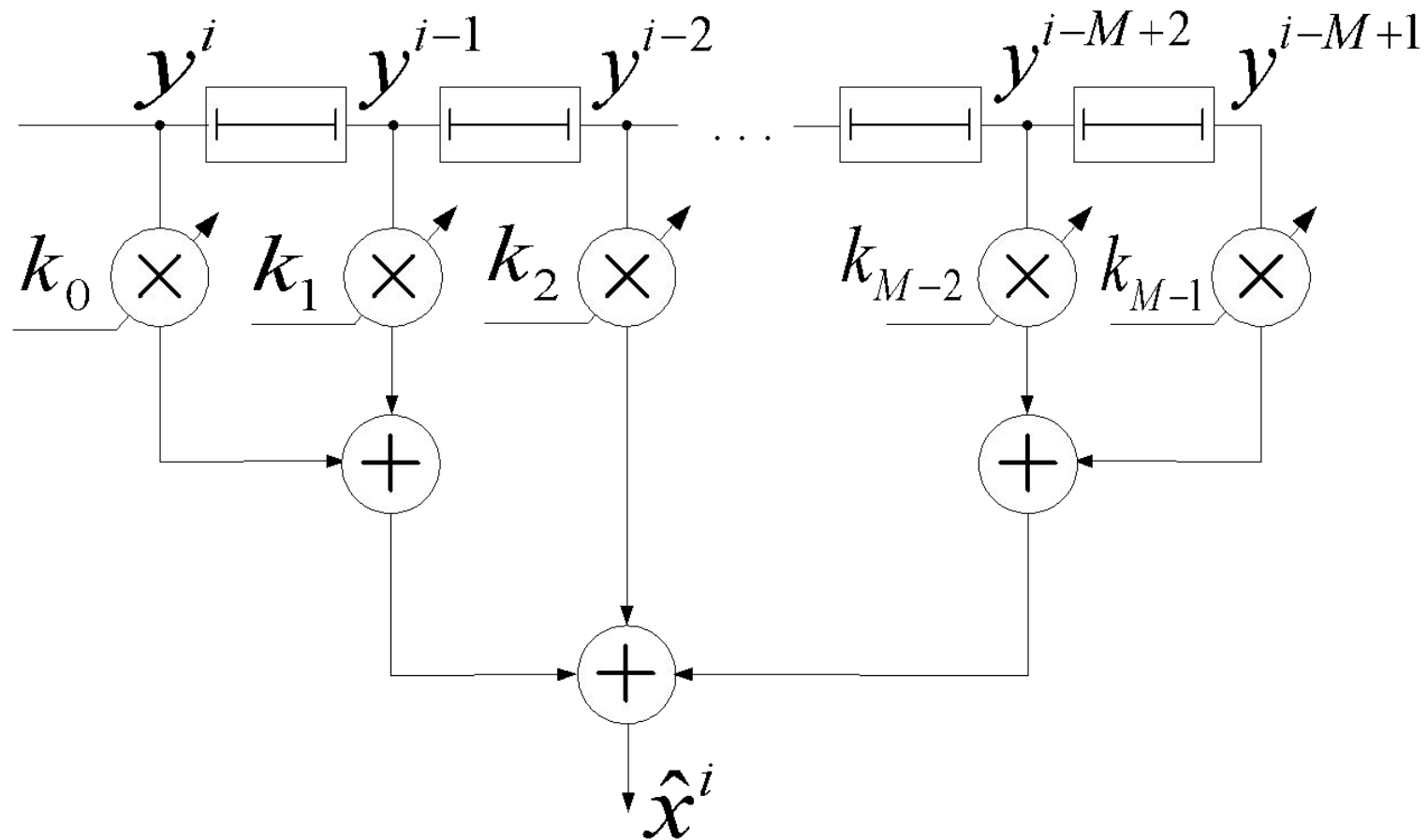


Сигнал на входе и выходе неизвестной системы



Сигнал на входе и выходе неизвестной системы, отфильтрованный сигнал

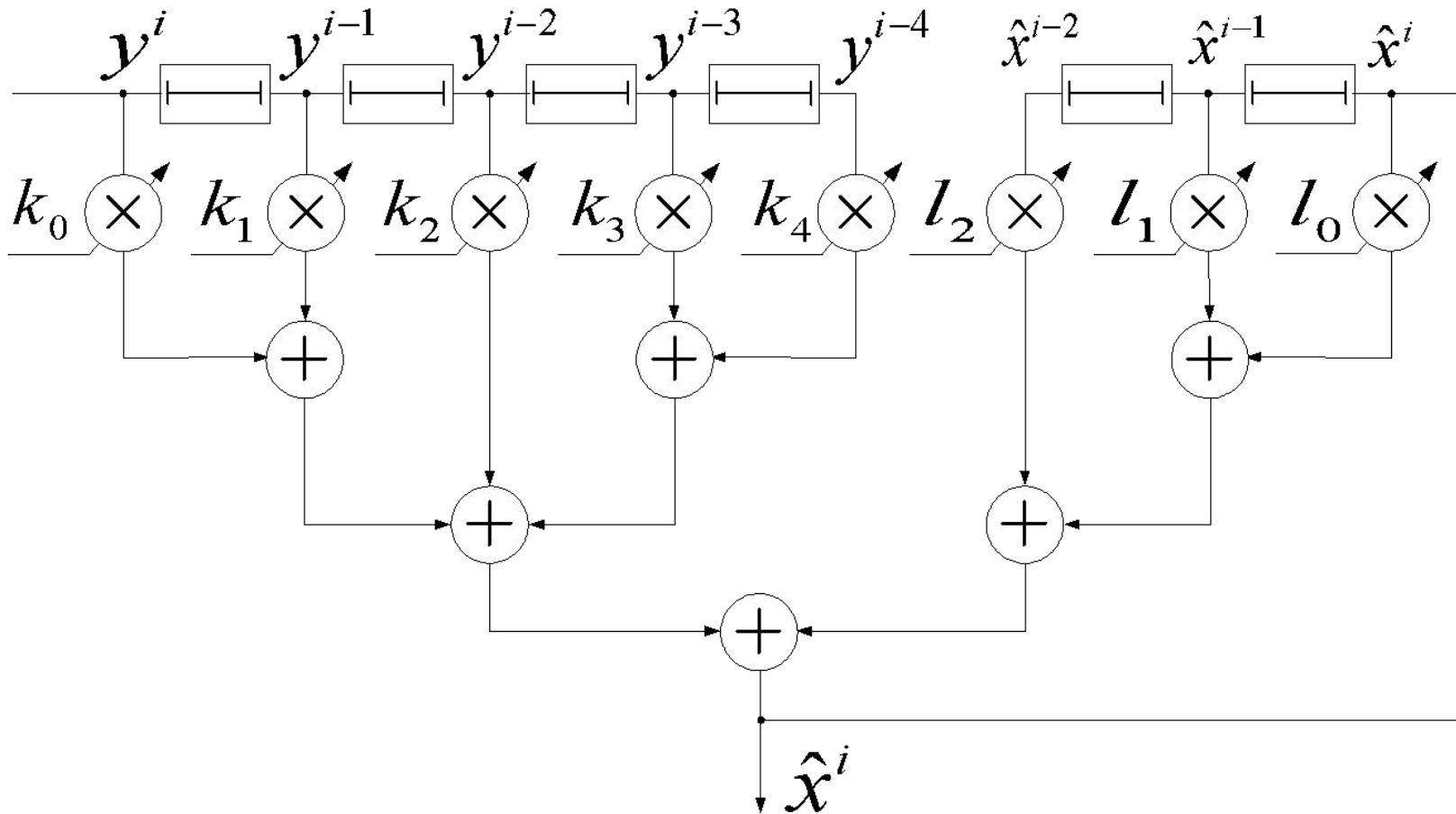
# Адаптивный нерекурсивный фильтр



$$\hat{x}^i = \sum_{m=0}^{M-1} k^m y^{i-m} = \mathbf{ky}^i$$



# Адаптивный рекурсивный фильтр



$$\hat{x}^i = \sum_{m=0}^{M-1} k^m y^{i-m} - \sum_{n=0}^{N-1} l^n \hat{x}^{i-n} = \mathbf{k} \mathbf{y}^i - \mathbf{l} \hat{\mathbf{x}}^i$$

$$\boldsymbol{\varepsilon}^i = x^i - \hat{x}^i = x^i - \sum_{m=0}^{M-1} k^m x^{i-m} = x^i - (\mathbf{k}^i)^T \mathbf{y}^i$$

$$(\boldsymbol{\varepsilon}^i)^2 = (x^i)^2 + (\mathbf{k}^i)^T \mathbf{y}^i (\mathbf{y}^i)^T \mathbf{k}^i - 2x^i (\mathbf{y}^i)^T \mathbf{k}^i$$

$$E[(\boldsymbol{\varepsilon}^i)^2] = E[(x^i)^2] + (\mathbf{k}^i)^T E[\mathbf{y}^i (\mathbf{y}^i)^T] \mathbf{k}^i - 2E[x^i (\mathbf{y}^i)^T] \mathbf{k}^i$$

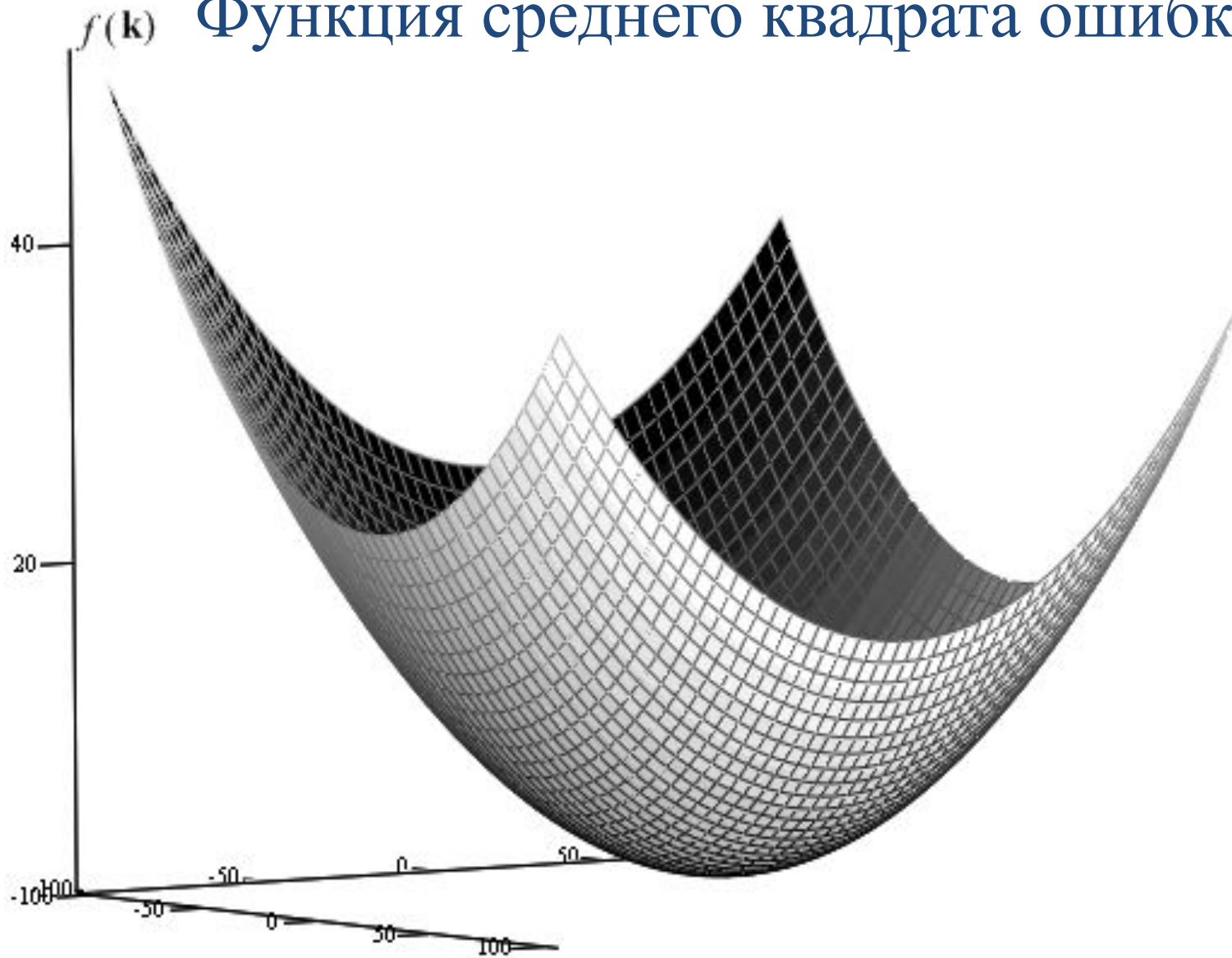
$$\xi = E[(\boldsymbol{\varepsilon}^i)^2] = E[(x^i)^2] + (\mathbf{k}^i)^T \mathbf{R} \mathbf{k}^i - 2\mathbf{P}^T \mathbf{k}^i$$

$$\nabla = \frac{\partial \xi}{\partial \mathbf{k}^i} = 2\mathbf{R}(\mathbf{k}^i) - 2\mathbf{P} \quad \nabla = 0 = 2\mathbf{R}\mathbf{k}^* - 2\mathbf{P}$$

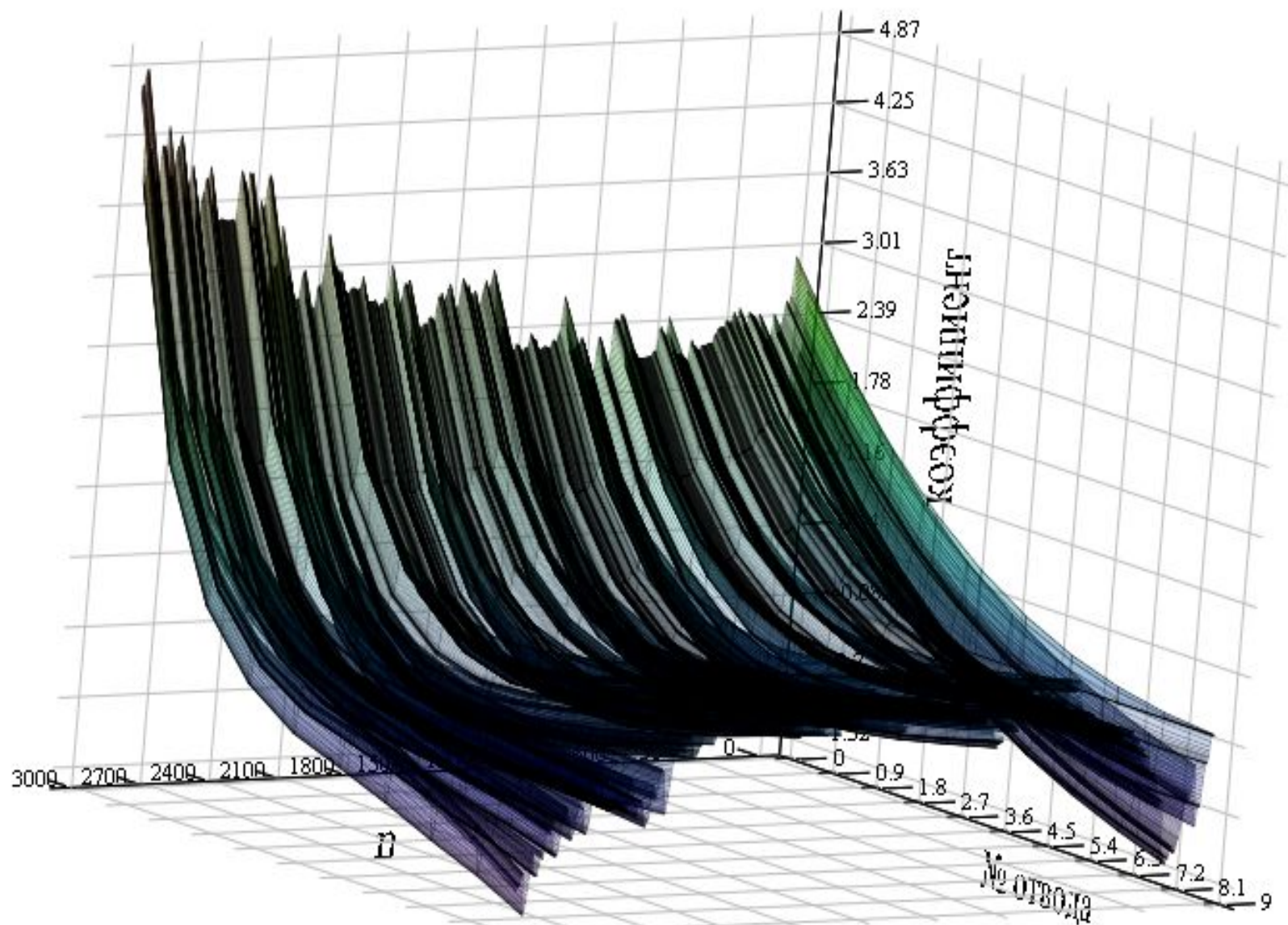
$$\mathbf{k}^* = \mathbf{R}^{-1} \mathbf{P}$$

$$\xi_{\min} = E[(x^i)^2] - \mathbf{P}^T \mathbf{k}^*$$

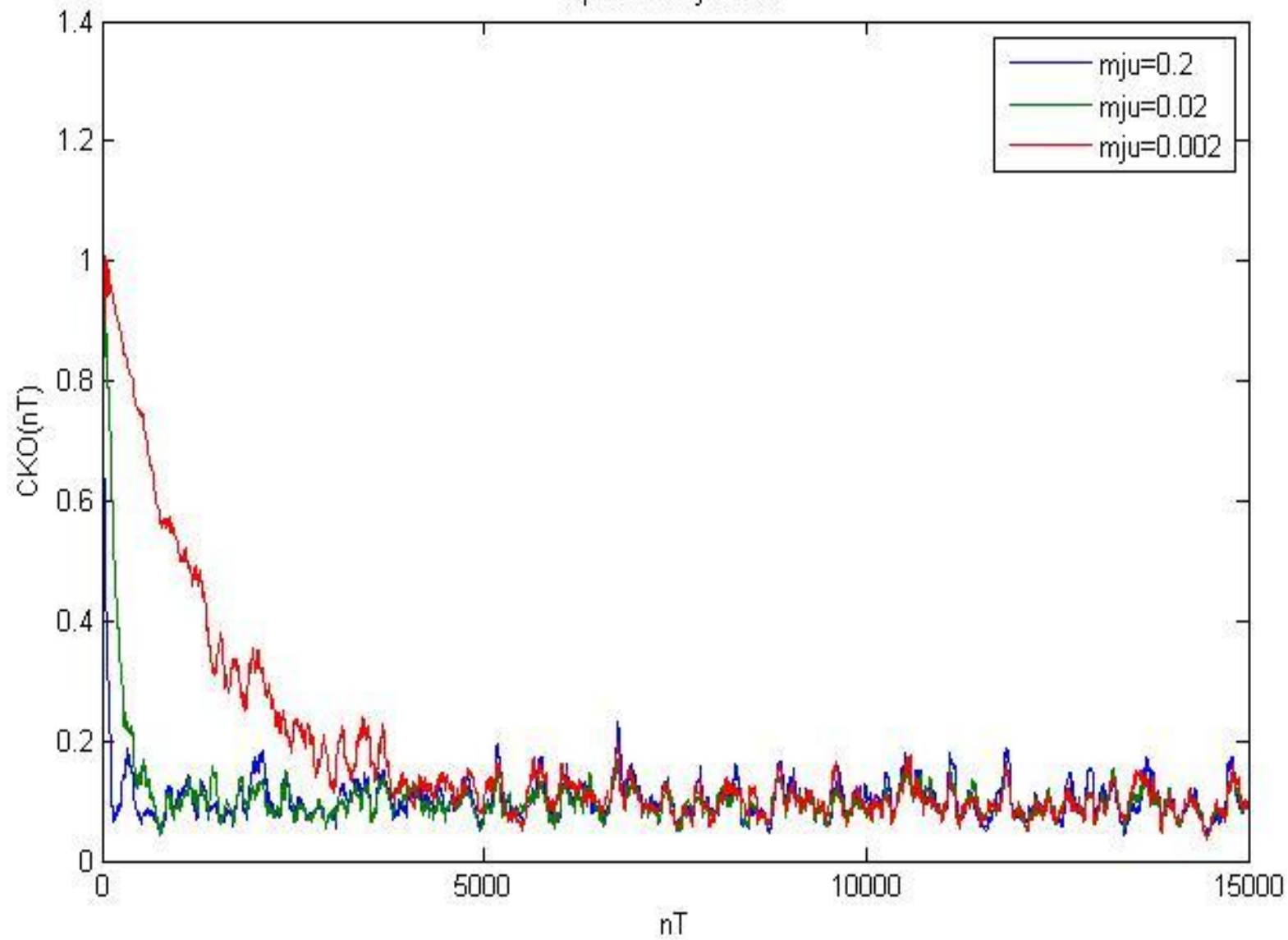
# Функция среднего квадрата ошибки



# динамика подстройки вектора весовых коэффициентов



Кривые обучения



## Другое представление градиента

$$\xi = \xi_{\min} + (\mathbf{k}^i - \mathbf{k}^*)^T \mathbf{R}(\mathbf{k}^i - \mathbf{k}^*)$$

$$\xi = \xi_{\min} + (\mathbf{v}^i)^T \mathbf{R} \mathbf{v}^i$$

$$\nabla = \frac{\partial \xi}{\partial \mathbf{v}^i} = 2\mathbf{R} \mathbf{v}^i$$

Декорреляция сигнала ошибки и входного сигнала

$$\mathbf{k}^i = \mathbf{k}^*$$

$$\varepsilon^i = x^i - (\mathbf{k}^i)^T \mathbf{y}^i$$

$$\varepsilon^i \mathbf{y}^i = x^i \mathbf{y}^i - \mathbf{y}^i (\mathbf{y}^i)^T \mathbf{k}^i$$

$$E[\varepsilon^i \mathbf{y}^i] = E[x^i \mathbf{y}^i] - E[\mathbf{y}^i (\mathbf{y}^i)^T] \mathbf{k}^i$$

# Поиск СКО

$$\xi = \xi_{\min} + \lambda(k - k^*)^2 \quad \frac{\partial \xi}{\partial k} = 2\lambda(k - k^*) \quad \frac{\partial^2 \xi}{\partial k^2} = 2\lambda$$

Простой алгоритм градиентного поиска

$$k^{i+1} = k^i + \mu(-\nabla^i) \quad \nabla^i = \left. \frac{\partial \xi}{\partial k} \right|_{k=k^i} = 2\lambda(k^i - k^*)$$

$$k^{i+1} = k^i - 2\mu\lambda(k^i - k^*) = (1 - 2\mu\lambda)k^i + 2\mu\lambda k^*$$

$$k^i = k^i - 2\mu\lambda(k^i - k^*) = (1 - 2\mu\lambda)k^i + 2\mu\lambda k^*$$

$$k^i = (1 - 2\mu\lambda)^i k^0 + 2\mu\lambda k^* \frac{1 - (1 - 2\mu\lambda)^i}{1 - (1 - 2\mu\lambda)}$$

$$k^i = k^* + (1 - 2\mu\lambda)^i (k^0 - k^*)$$

# Поиск СКО

$$\xi = \xi_{\min} + \lambda(k - k^*)^2 \quad \frac{\partial \xi}{\partial k} = 2\lambda(k - k^*) \quad \frac{\partial^2 \xi}{\partial k^2} = 2\lambda$$

Простой алгоритм градиентного поиска

$$k^{i+1} = k^i + \mu(-\nabla^i) \quad \nabla^i = \left. \frac{\partial \xi}{\partial k} \right|_{k=k^i} = 2\lambda(k^i - k^*)$$

$$k^{i+1} = k^i - 2\mu\lambda(k^i - k^*) = (1 - 2\mu\lambda)k^i + 2\mu\lambda k^*$$

$$k^i = k^i - 2\mu\lambda(k^i - k^*) = (1 - 2\mu\lambda)k^i + 2\mu\lambda k^*$$

$$k^i = (1 - 2\mu\lambda)^i k^0 + 2\mu\lambda k^* \frac{1 - (1 - 2\mu\lambda)^i}{1 - (1 - 2\mu\lambda)}$$

$$k^i = k^* + (1 - 2\mu\lambda)^i (k^0 - k^*)$$