

Архитектура ВМиС

Питолин Владимир Евгеньевич -

к.т.н., доцент, 8-(029)-813-18-81

pitolin594@gmail.com

1. Введение.

Информация- сведения о чем-то, совокупность данных об определенном объекте, явлении, выраженные в определенном виде.

Аналоговый вид- сравнение с аналогом (по степени отклонения стрелки, яркости, цвету, вкусу) является в большей степени субъективным.

Дискретный вид- или оцифрованный более точен и объективен, позволяет передавать информацию в отличие от аналогового вида двумя принципиально разными способами:

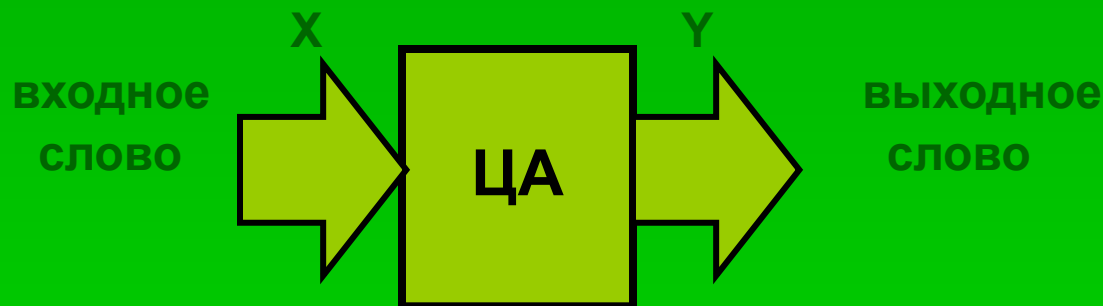
последовательно и параллельно

Цифровой автомат (ЦА) – всякая искусственная система обработки информации, представленной в дискретной форме

- в последовательном виде :



- в параллельном виде :



Работа ЦА заключается в преобразовании входного слова того или иного алфавита в выходное слово.

Алфавит – совокупность символов из которых состоят слова информации.

Преобразование слов в ЦА происходит только в строго определенные моменты времени, которые называются тактовыми.

Интервал времени между двумя тактовыми моментами называется тактом.

Длительность тактов может быть неодинаковой, но в общем случае они равновелики и определяются генератором синхронизации (ГС).

Принципы построения алгоритма (А)

1. А - это жесткая инструкция для исполнителя.
2. Отдельный А может составлять часть другого, более сложного.
3. Любой А является дискретным, т.е. распадающимся на ряд последовательных предписаний.
4. Исполнитель не должен ошибаться.
5. Алгоритм универсален, т.к. работает с цифровой обезличенной информацией.
6. Алгоритм результативен, т.е. имеет начало и свое завершение.

Принципы построения архитектуры ЭВМ (принципы Д. Неймана 1945 г.)

1. Информация кодируется в двоичной форме и распределяется на единицы (элементы) информации, называемые словами.
2. Разнотипные слова информации различаются по способу использования, своему размеру, но не по способу кодирования.
3. Слова информации размещаются в ячейках памяти ЭВМ и идентифицируются номерами ячеек, которые называются адресами.
4. Алгоритм решаемой задачи представляется в форме последовательности управляющих слов – команд, которые определяют наименование операции и слова информации, участвующие в операции.

Алгоритм, представленный в терминах машинных команд, называется программой.

Официально считается, что первую программу написал Алан Тьюринг (1936-й г.), представленную в виде таблицы с полем данных (символов), полем результата и набором основных и служебных (системных) команд назвали «Машиной Тьюринга».

5. Выполнение вычислений, предписанных алгоритмом, сводится к последовательному выполнению команд в порядке однозначно определяемом программой.

Первой выполняется команда, заданная пусковым адресом программы.

Адрес следующей команды однозначно определяется в процессе выполнения текущей команды и может быть:

- либо адресом следующей по порядку команды,
- либо адресом любой другой команды.

Процесс вычислений продолжается до тех пор, пока не будет выполнена команда, предписывающая прекращение вычислений.

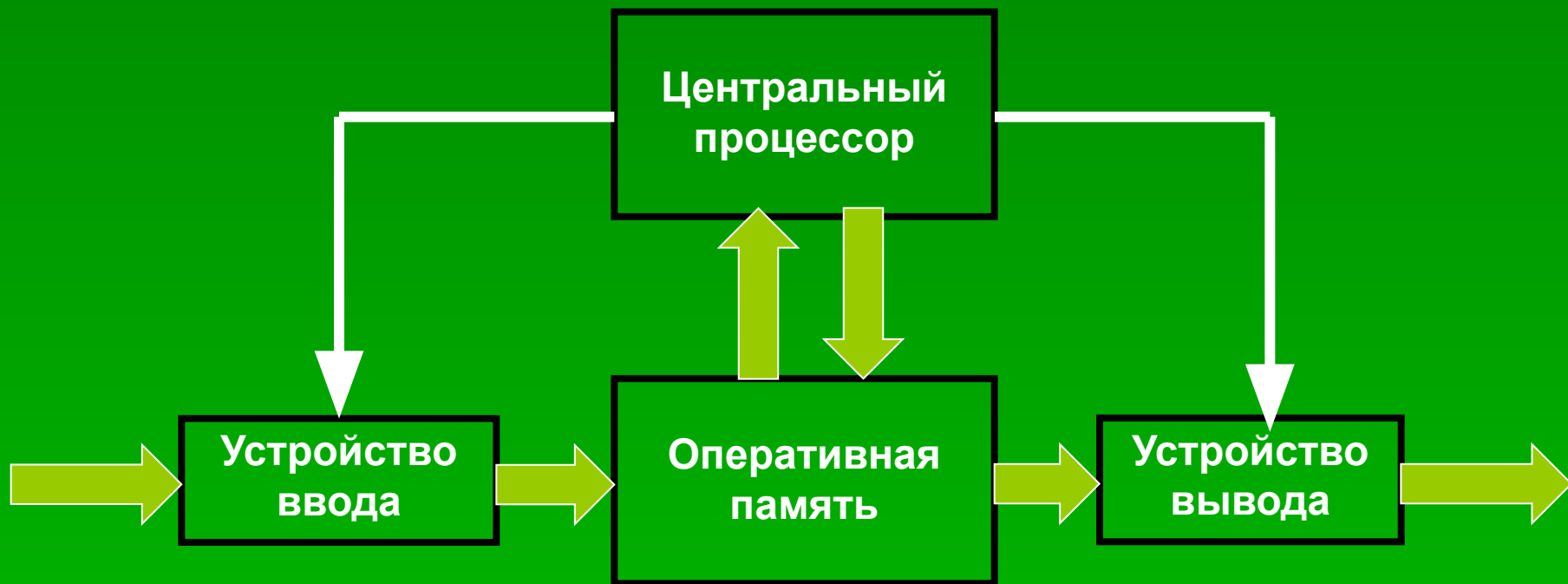
Общее определение ЭВМ:

ЭВМ – это программно управляемая инженерная система, предназначенная для восприятия, хранения, обработки и передачи информации.

Нэймановское определение ЭВМ:

ЭВМ – это универсальный цифровой автомат, работающий под управлением программы, хранящейся в его памяти.

Принципиальная базовая архитектура ЭВМ



Устройство ввода – ввод информации с различных носителей.

Устройство вывода – вывод информации в форме, доступной человеку.

Оперативная память – содержит программу обработки, сохраняет исходные данные и записывает промежуточные результаты.

Центральный процессор – универсальный цифровой автомат для обработки информации и управления вводом-выводом данных.

2. Кодирование информации в ПЭВМ

Физический носитель нуля и единицы (VT-диаграмма).



Исходное напряжение снимается с источника питания +5 вольт. Системные решения определяют рабочие уровни напряжения «1», при этом безразлично, что является причиной этого напряжения: заряд конденсатора, сопротивление р-п перехода транзистора или полупроводникового слоя ячейки памяти.

Длительность импульса задается генератором синхросигнала.

Такая схема представления «1» разработана и предложена английским физиком Клодом Шенноном в 1950-60 гг.

Двоичное кодирование простых чисел (пределы).

Двоичное число, носителем которого является напряжение, называют двоичным разрядом или битом.

Несколько разрядов, объединённые в регистр, насчитывающий 8 разрядов, называется байтом и изображаются -

2 байта и более составляют машинное слово.

1	0	0	0	1	1	0	0
---	---	---	---	---	---	---	---

Простые числа кодируются прямым двоичным кодом т.е. простым переводом десятичного числа в двоичную форму.

Простые числа без знака занимают все разряды байта - от 0 до 255.

Простые числа со знаком:

0 - «+»	1 - «-»
---------	---------

знак числа -

1	0	0	0	1	1	0	0
---	---	---	---	---	---	---	---

пределы представления - от -127 до + 127.

Но возникает проблема нуля:

для вычислительной машины $+0 \neq -0$ так как $00000000 \neq 10000000$

Существует решение этой проблемы: числа со смещённым кодом.

База смещения кода определяется старшим разрядом байта = 10000000 .

В смещённом коде все числа положительные.

Число в смещённом коде получается за счет дополнения базы смещения реальным числом с учетом его знака:

например: $10000000 + 11001 = 10011001$ – положительное число + 25,

$10000000 - 11001 = 01100111$ – отрицательное число - 25.

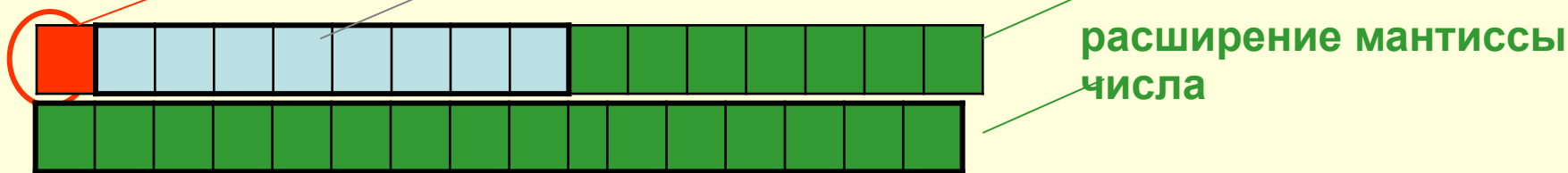
Пределы представления однобайтовых чисел : от - 128 до + 128.

Увеличение пределов достигается за счет количества байтов в слове.

Двоичное кодирование вещественных чисел (с плавающей запятой).

Нормализованная форма представления двоичного числа с плавающей запятой для коротких вещественных чисел стандарта IEEE

Знак числа степень числа в смещенном коде **мантисса числа**

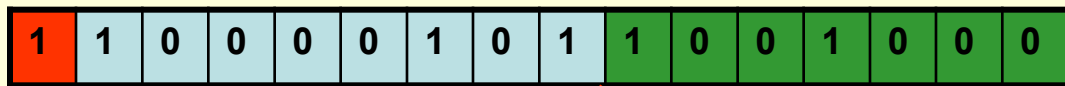


например: десятичное число 25
Приводим его к нормализованному виду:
Преобразуем в двоичную форму:
Введем в шаблон:

$$\begin{aligned} -25 &= -0,25 * 10^2 \\ &= -0,11001 * 2^{101} = -0,1001 * 2^{101} \end{aligned}$$

примечание:

в связи с тем, что мантисса двоичного числа всегда начинается с единицы, то ее исключают из состава шаблона, а в схемах преобразования вводят специальное дополнение.



Пределы представления числа по абсолютной величине $10^{-38} - 10^{+39}$ и

7-8 значащих цифр мантиссы. Для длинных вещественных чисел стандарта Intel (8-ми байтных) количество значащих цифр увеличено до 16-ти.

Недостатки использования чисел с плавающей запятой:

- 1) При умножении $2*2$ получается десятичное число 3,999... , а не 4.
- 2) Для небольших чисел резервируется слишком много памяти (8 байт для длинных вещественных чисел).
- 3) Для обработки чисел с плавающей запятой необходим сопроцессор.

Кодирование символов (принцип).

Принцип кодирования символов заключается в присвоении порядковых номеров стандартным изображениям символов, используемых в алфавите.

Сами изображения символов представляют собой т.н. «тексели», т.е. элементарные изображения в виде матрицы 8x8 пикселей, объединенные в массив 8-байтовых элементов и хранящийся в ПЗУ BIOS (адрес F000:FA6E), например, латинский символ «а», стоящий под номером 97 таблицы ASCII:

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	1	1	1	0	0	0
0	0	0	0	1	1	0	0
0	1	1	1	1	1	0	0
1	1	0	0	1	1	0	0
0	1	1	1	0	1	1	0

00 FFD75 В 8-байтном элементе (текселе) «1»

00 означает горящий пиксел, «0» - погашенный.

00 На первый байт (из 8) элемента тексела в

38 ПЗУ BIOS указывает вектор прерывания Int1F,

0C расположенный в таблице векторов по адресу

7C 0000:007C. Вектор используется, как

CC указатель на массив расширения таблицы

76 ASCII для считывания 8-ми байт элемента

изображения тексела (начиная с F000:FD78).

К сожалению стандартная таблица ASCII IBM не содержит изображений символов кириллицы. Для этой цели принято изображение национальных шрифтов помещать в таблицу расширения ASCII, т.е. с номера 128 и далее, на котором в стандартной таблице расположены символы греческого алфавита. В MS DOS страница ASCII с кириллицей имеет номер 866.

Существует несколько стандартизованных таблиц изображений символов. Фирма Intel в базовых конфигурациях BIOS своих изделий, да и другие разработчики BIOS используют таблицу ASCII. Фирма MicroSoft для OS Windows с целью облегчения применения различных шрифтов и знаков использует расширенную таблицу ANSI. Существуют советские аналоги для ЭВМ отечественного изготовления: КОИ-7, КОИ-8 и т.д. (сокращенно от слов «КОдирование Информации»).

Смещение номеров символов кириллицы для ASCII по таблице ANSI:

ANSI (WINDOWS)

$$192 \leq N_{(ANSI)} < 240$$

$$240 \leq N_{(ANSI)} < 256$$

$$N_{(ANSI)} = 184$$

$$N_{(ANSI)} = 168$$

ASCII 866 (MS DOS)

$$N_{(ASCII)} = N_{(ANSI)} - 64$$

$$N_{(ASCII)} = N_{(ANSI)} - 16$$

$$N_{(ASCII)} = 241$$

$$N_{(ASCII)} = 240$$

В настоящее время разработано очень большое количество различных шрифтов для различных областей приложений.

Обеспечена возможность и самостоятельной разработки, как шрифтов, так и символов различного назначения.

Основное условия их применения: обеспечение увязки с соответствующими СКЭН-кодами клавиатуры и алгоритмом обработки этих СКЭН-кодов.

3. Понятие о микропрограммном автомате (МПА).

Схема и принцип действия МПА с программой в памяти.

В любом устройстве обработки цифровой информации можно выделить два основных блока – операционный и управляющий.

Операционный блок состоит из регистров, сумматоров и других узлов, обеспечивающих прием из внешней среды, обработку и выдачу во внешнюю среду цифровой информации, а также выдачу в управляющий блок и внешнюю среду оповещающих сигналов о состоянии операционного блока.

Управляющий блок (или управляющий автомат) вырабатывает распределенную во времени последовательность управляющих сигналов, порождающих в операционном блоке нужную последовательность микроопераций.

Совокупность управляющих сигналов зависит от кода операции и от совокупности поступающих оповещающих сигналов.



Любая команда, операция или процедура, выполняемая в операционном блоке, описывается некоторой микропрограммой или комбинационной схемой и реализуется за несколько циклов, в каждом из которых выполняется одна или несколько микрокоманд.

По причине реализации управляющего автомата с микропрограммным управлением его иногда называют **микропрограммным автоматом** (МПА).

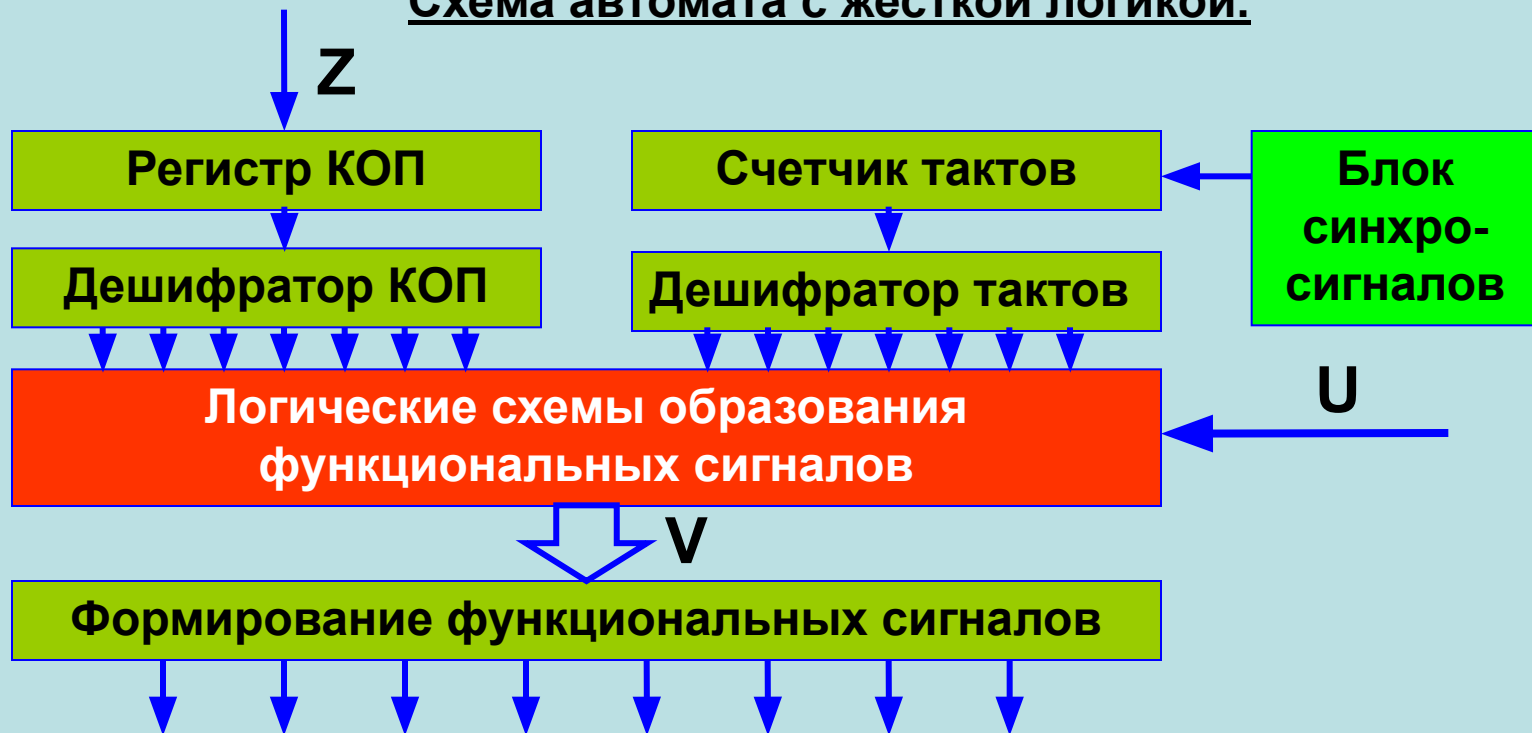
Управляющий Автомат с жёсткой логикой – каждой операции, задаваемой кодом операции, строится конкретный набор комбинационных схем, которые в соответствующих тактах вырабатывают соответствующие сигналы управления. Комбинационные схемы могут содержать триггерные ячейки памяти, дешифраторы, мультиплексоры, сумматоры и т.д.

Работа автомата с жесткой логикой интерпретируется или может быть интерпретирована микропрограммой МПА, но реализуется аппаратными средствами, схемотехническими конструкциями.

Для формирования счетчика-дешифратора тактов в автомате с жесткой логикой используется сдвиговый регистр.

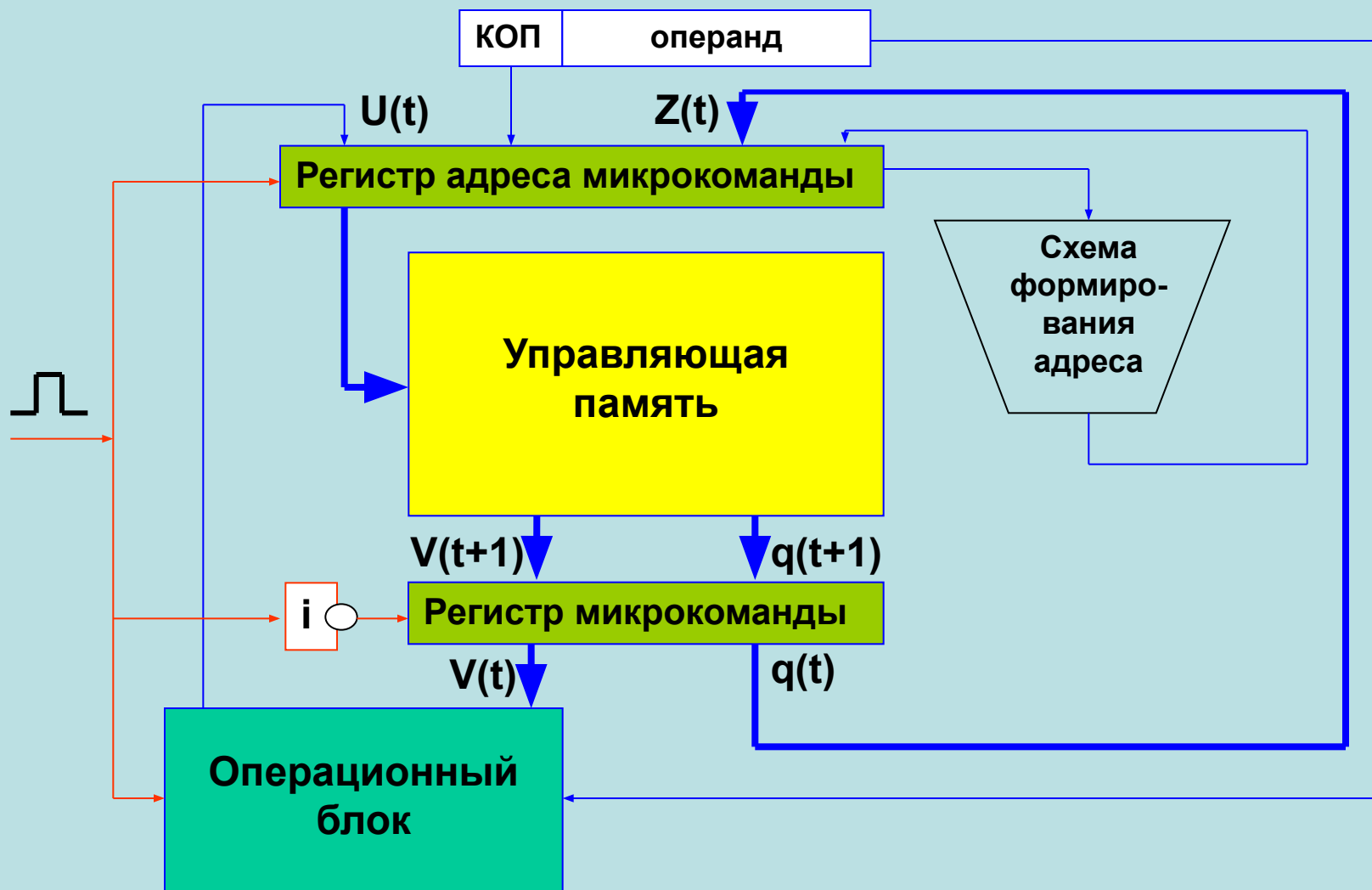
Логические схемы строятся на базе программируемых логических матриц (ПЛМ). Программирование ПЛМ осуществляется фотошаблонированием за счет выжигания ненужных логических связей.

Схема автомата с жесткой логикой.



В свое время МПА с жесткой логикой на базе комбинационных схем были очень распространены, т.к. являлись достаточно скоростными. Но изменение этих схем требовало перепроектирования аппаратных средств реализации логики и на определенном этапе эта сложность становилась непреодолимой. Кроме этого логические схемы статичны и рассчитаны на ряд простых функциональных операций. Изменение состава микрокоманд и функциональных сигналов также требует перепроектирования схем жесткой логики управляющих автоматов.

Схема микропрограммного автомата



МПА с хранимой в памяти программой – коду операции (КОП) ставится в соответствие совокупность хранимых в памяти слов – микрокоманд, содержащих информацию о микрооперациях, подлежащих выполнению в течении одного машинного такта и указание на то, какое должно быть выбрано из памяти следующее слово (следующая микрокоманда).

Совокупность микропрограмм хранится в специальном ППЗУ.

Микропрограммы используются в явной форме, программируются в кодах микрокоманд и в таком виде заносятся в память.

Набор значений аргументов удобно отождествляясь с адресом микрокоманды. Этот набор вносится в регистр адреса во время такта.

В управляющей памяти по этому адресу хранится код, задающий набор значений выходных сигналов управляющего автомата $V(t+1)$ – операционная часть микрокоманды и набор значений переменных $q(t+1)$.

Воздействие сигналов $V(t)$ на операционный блок синхронизируется сигналом генератора из регистра микрокоманды, находящемся в режиме хранения.

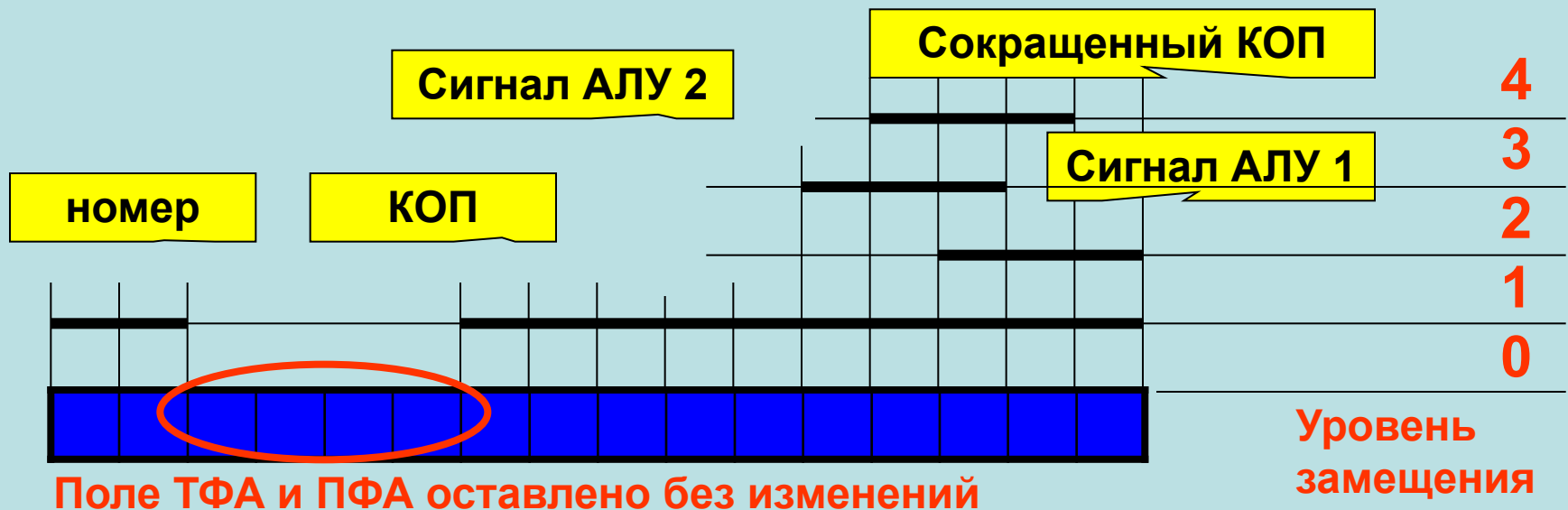
Для сокращения размера памяти на порядок адрес микрокоманды формируется специальной комбинационной схемой формирования адреса.

Принцип принудительной адресации МПА

Адрес очередной микрокоманды (а микропрограмма обычно состоит из нескольких микрокоманд) можно назначить без учета $Z(t)$ и $U(t)$, т.е. поле адреса разбивается по этому признаку на ряд групп, часть из которых заполняется принудительно (последовательно) схемой формирования адреса, а другие остаются без изменений, например:

- поле типа формирования адреса (ТФА),
- поле формирования разряда очередного адреса (ПФА).

Схема принудительного заполнения регистра адреса



Принцип естественной адресации заключается в замене адресной части счетчиком команд, используя естественный порядок следования команд.