



**ВОЕННАЯ КАФЕДРА
при НАО «КазНИТУ имени К.И. САТПАЕВА»**

**ЦИКЛ
ИНФОРМАЦИОННОЙ ЗАЩИТЫ**



Дисциплина

«Структура компьютерных средств»

Тема №2

«Устройства управления»

Занятие №1/2

**«Микропрограммный автомат с
программируемой логикой»**



Учебные вопросы:

- 1. Кодирование микропрограмм.**
- 2. Обеспечение порядка следования микрокоманд.**
- 3. Организация памяти микропрограмм.**

Цели занятия:

- ✓ Изучить принципы организации микропрограммного автомата с программируемой логикой;**
- ✓ Обучить кодированию микропрограмм и организации памяти микропрограмм.**

Учебный вопрос №1.

«Кодирование микропрограмм»

Принципиально иной подход, позволяющий преодолеть сложность УУ с аппаратной (жесткой) логикой, был предложен британским ученым М.Уилксом в начале 50-х годов. В основе идеи лежит тот факт, что для инициирования любой микрооперации достаточно сформировать соответствующий СУ на соответствующей линии управления, то есть перевести такую линию в активное состояние. Это может быть представлено с помощью двоичных цифр 1 (активное состояние - есть СУ) и 0 (пассивное состояние - нет СУ).

Вопрос №1. «Кодирование микропрограмм»

Сигналы управления в МПА с программируемой логикой представляются с помощью управляющих слов - микрокоманд (МК). Микрокоманда соответствует одному такту работы ВМ и определяет, какие СУ должны быть сформированы в данном такте.

Последовательность МК, по тактам описывающая выполнение определенного этапа цикла команды, образует микропрограмму (МП).

Вопрос №1. «Кодирование микропрограмм»

Основное содержимое УПМ составляют микропрограммы этапа исполнения для каждой из команд, входящих в систему команд ВМ. Некоторые из хранящихся в УПМ микропрограмм являются общими для всех команд, например, микропрограмма этапа выборки команды или этапа формирования адреса следующей команды.

Вопрос №1. «Кодирование микропрограмм»



Рис. 2.6. Размещение микропрограмм в управляющей памяти

Вопрос №1. «Кодирование микропрограмм»

Работу формирователя сигналов управления МПА с программируемой логикой задает схема формирования адреса микрокоманды (СФАМ). Она обеспечивает определение *адреса очередной микрокоманды*, занесение его в *регистр адреса управляющей памяти*, *считывание микрокоманды* из *управляющей памяти* и *занесение в регистр микрокоманды*.

Вопрос №1. «Кодирование микропрограмм»



Рис. 2.7. Формирователь сигналов управления в микропрограммном автомате с программируемой логикой

Вопрос №1. «Кодирование микропрограмм»

В состав СФАМ входит *синхронизатор*, и *вспомогательные логические схемы*.

Цикл любой команды начинается с исходного состояния синхронизатора (такт T_0 - счетчик тактов обнулен). СФАМ заносит в регистр адреса управляющей памяти адрес первой МК микропрограммы выборки команды. В том же такте по сигналу чтения Чт происходит считывание адресуемой микрокоманды из управляющей памяти и ее запись в регистр микрокоманды (РМК).

Вопрос №1. «Кодирование микропрограмм»

Каждая микрокоманда, считанная из управляющей памяти содержит микрооперационную (МО) и адресную (А) части.

Микрооперационная часть микрокоманды поступает на дешифратор микрокоманды, на выходе которого образуются сигналы управления. Адресная часть микрокоманды подается в СФАМ, где формируется адрес следующей микрокоманды. Сформированный адрес микрокоманды снова записывается в регистр адреса управляющей памяти, и процесс повторяется вплоть до завершения микропрограммы.

Вопрос №1. «Кодирование микропрограмм»

Информация о том, какие сигналы управления должны быть сформированы в процессе выполнения текущей МК, в закодированном виде содержится в микрооперационной части (МО) микрокоманды.

Способ кодирования микрооперации во многом определяет сложность аппаратных средств формирователя сигналов управления (ФСУ) и его скоростные характеристики.

Вопрос №1. «Кодирование микропрограмм»

Известные варианты кодирования сигналов управления можно свести к трем группам: горизонтальному, вертикальному и смешанному кодированию.



Рис. 2.9. Способы кодирования микроопераций: а – горизонтальный; б – вертикальный; в – горизонтально-вертикальный; г – вертикально-горизонтальный.

Вопрос №1. «Кодирование микропрограмм»

При горизонтальном кодировании (рис. 2.9, а) каждый разряд микрооперационной части микрокоманды представляет один из возможных сигналов управления, то есть определенную микрооперацию. При общем количестве СУ равном t , длина микрооперационной части МК составляет t битов.

Независимость разрядов МО позволяет в рамках одного тактового периода одновременно выполнять вплоть до t микроопераций, причем в любом их сочетании, что является несомненным достоинством горизонтального кодирования.

Вопрос №1. «Кодирование микропрограмм»

При вертикальном кодировании (рис. 2.9, б) каждой микрооперации присваивается уникальный код, который и указывается в микрооперационной части МК. Разрядность этого кода (длина МО) равна наименьшему целому, большему или равному $\log_2 m$, что можно записать как $\lceil \log_2 m \rceil$. Таким образом, требования к емкости управляющей памяти по сравнению с горизонтальным кодированием существенно снижаются, но возникает потребность в дешифраторе для преобразования кода микрооперации в соответствующий сигнал управления.

Вопрос №1. «Кодирование микропрограмм»

Компромиссное сочетание достоинств и недостатков *горизонтального и вертикального кодирования* достигается путем смешанного кодирования. Известны несколько вариантов такого кодирования, среди которых наиболее распространены *горизонтально-вертикальное и вертикально-горизонтальное кодирование*. В обоих вариантах множество всех m возможных сигналов управления разбивается на k подмножеств. При разбиении на подмножества каждая микрооперация может присутствовать лишь в одном из подмножеств.

Вопрос №1. «Кодирование микропрограмм»

Иногда используется двухуровневое кодирование микроопераций. На первом уровне с вертикальным кодированием выбирается микрокоманда, поле МО содержит адрес горизонтальной микрокоманды второго уровня - нанокоманды. Данный способ сочетания вертикального и горизонтального кодирования часто называют нанокодированием. Метод предполагает двухуровневую систему кодирования микроопераций и, соответственно, двухуровневую организацию управляющей памяти (рис 2.10).

Вопрос №1. «Кодирование микропрограмм»

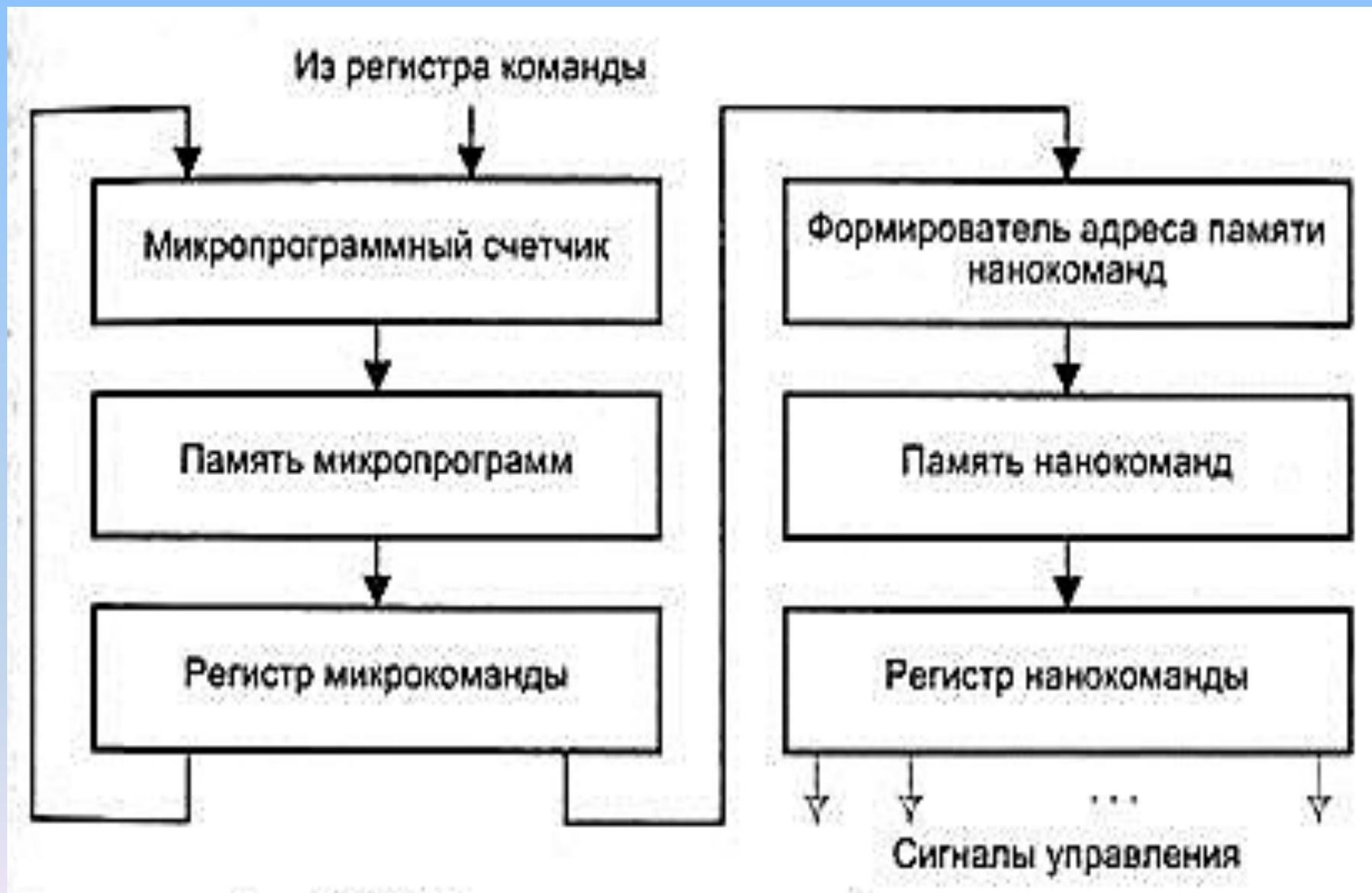


Рис. 2.10. Нанопрограммное устройство управления

Вопрос №1. «Кодирование микропрограмм»

Контрольные вопросы:

- 1. Обоснуйте название МПА с программируемой логикой. Сформулируйте достоинства и недостатки таких МПА.**
- 2. Дайте характеристику элементов структуры МПА с программируемой логикой.**
- 3. Объясните принцип управления на основе МПА с программируемой логикой.**
- 4. Какие способы кодирования микрокоманд вы знаете? Перечислите их. достоинства и недостатки.**

Учебный вопрос №2.

«Обеспечение порядка следования микрокоманд»

Порядок следования микрокоманд в микропрограмме в общем случае может зависеть от состояния какого-либо признака, обычно из тех, что хранятся в регистре признаков АЛУ. По этой причине в УУ необходимо предусмотреть эффективную систему реализации переходов.

При выполнении микропрограммы адрес очередной микрокоманды относится к одной из трех категорий:

- определяется кодом операции команды;**
- является следующим по порядку адресом;**
- является адресом перехода.**

Учебный вопрос №3.

«Организация памяти микропрограмм»

Основные способы организации памяти микропрограмм:

1. Каждое слово УПМ содержит одну микрокоманду. Это наиболее простая организация УПМ. Основной недостаток - в каждом такте работы МПА требуется обращение к памяти микропрограмм, что приводит к снижению быстродействия МПА.

2. Одно слово УПМ содержит несколько микрокоманд. В результате осуществляется одновременное считывание из УПМ нескольких МК, что позволяет повысить быстродействие УУ.

Вопрос №3. «Организация памяти микропрограмм»

3. Сегментация ПМП, при которой память разделяется на сегменты, состоящие из 2^q соседних слов, при этом адрес слова АМК разделяется на два поля: S и A. Поле S определяет адрес сегмента, а поле A - адрес слова в сегменте. Адрес S устанавливается специальной микрокомандой. В последующих микрокомандах указывается только адрес слова A в сегменте. Таким образом, разрядность адресной части МК уменьшается.

4. Двухуровневая память. Первый уровень - *микропамять*, хранящая микрокоманды. Вторым уровнем - *нанопамять*, содержащая наноккоманды.