

Лекция6. Устройство управления: структура, способы формирования управляющих сигналов и адресации микрокоманд

Устройство управления в том или ином виде присутствует в любом современном микропроцессоре, микроконтроллере и даже в графических процессорах видеокарт. Устройство управления может быть распределенным, но, тем не менее, четко логически различимым, с определенным набором функций. Из обобщенной структуры МП, представленной на рис. 1.2.1, видно, что УУ имеет связи со всеми основными блоками, т.е. занимает значимое место в этой структуре.

Впервые УУ было выделено в отдельный функциональный блок в 1946 году при описании архитектуры фон Неймана. С тех пор его роль и важность в процессорных системах не изменились.

Устройство управления — это законченный цифровой автомат, организующий выборку команд программы и обеспечивающий выполнение последовательности микроопераций в соответствии с кодом выполняемой команды. Напомним, что *микрооперацией* называют элементарное преобразование данных, не разложимое на более простые и выполняющееся в течение одного такта сигналов синхронизации.

Иногда говорят, что устройство управления является «мозгом мозга», потому что оно управляет процессором, который в свою очередь управляет всей вычислительной системой в целом.

В современных микропроцессорах принято выделять следующие основные функции УУ:

- синхронизация и координирование всех исполнительных устройств микропроцессора;
- выборка и декодирование команд из памяти;
- управление исполнительными блоками;
- управление ресурсами вычислительной системы;
- распределение потоков данных между процессором и внешними устройствами.

УУ устанавливает однозначное соответствие между множеством управляющих кодов и внешних условий $\{Q, U\}$ и множеством управляющих сигналов Y . Взаимодействие УУ с остальными блоками микропроцессорной системы представлено на рис. 2.1.1.

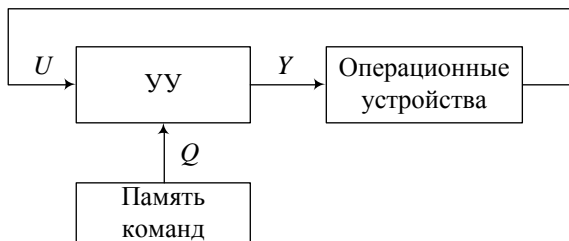


Рис. 2.1.1. Взаимодействие УУ с блоками микропроцессорной системы

Сформулируем главные требования, предъявляемые к УУ:

- УУ должно формировать полную совокупность управляющих сигналов Y ;
- число состояний автомата, моделирующего работу УУ, должно быть не меньше числа команд, подлежащих реализации;
- должна быть обеспечена возможность произвольной выборки команд в зависимости от управляющих кодов Q ;
- должна быть обеспечена возможность модификации последовательности команд в зависимости от внешних условий U .

Существуют два способа реализации устройств управления:

- на жесткой логике (hardwired control unit);
- с микропрограммным управлением (microprogrammed control unit).

Устройство управления на жесткой логике — это цифровой автомат, закон функционирования которого определяется коммутацией внутренних цепей, условия перехода из одного состояния в другое

реализуются с помощью КС, а состояния автомата хранятся в регистрах. Термин «жесткая логика» означает, что поведение УУ не может быть изменено без физической модификации схемы устройства, будь то печатная плата или микросхема. Это делает невозможным быстрое устранение найденных ошибок, а также добавление новых или изменение имеющихся функций процессора.

Рассмотрим пример реализации устройства управления на жесткой логике (рис. 2.1.2). Результатом работы УУ являются сигналы Y , каждый из которых есть функция от дешифрованного кода команды Q , флагов состояний процессора U и метки времени T :

$$Y = f(Q, U, T).$$

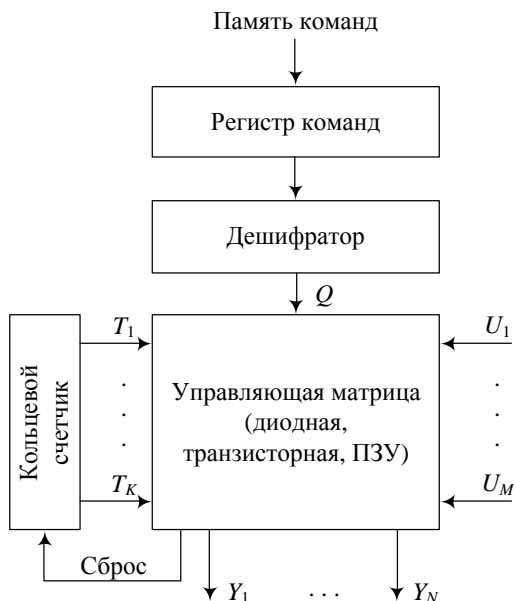


Рис. 2.1.2. Пример схемы УУ на жесткой логике

Текущая выполняемая команда поступает из памяти команд в регистр команд, откуда выбирается код операции и подается на дешифратор. Управляющая матрица объединяет все входящие сигналы и

выдает управляющие сигналы Y . Применение диодной, транзисторной матрицы или ПЗУ позволяет сделать УУ более регулярным, упростить топологию и уменьшить время разработки процессора в целом.

В общем случае управляющая память в устройстве управления может быть реализована с помощью программируемой логической матрицы (ПЛМ), ПЗУ или даже ОЗУ. В современных процессорах общего назначения фирм Intel и AMD память микропрограмм можно изменять с помощью специального программного обеспечения. Это позволяет исправлять технические ошибки в процессорах без замены всего устройства.

До конца 1970-х годов подавляющее большинство программ писались на ассемблере. При этом вся тяжесть управления процессором ложилась на программиста, что, в свою очередь, повышало требования к его квалификации и увеличивало время разработки программы. Для облегчения процесса написания программ производители МП вводили в них все более сложные команды, такие как, например, обработка текстовой строки одной инструкцией. Очевидно, что эти команды увеличивали сложность УУ, которое могло занимать до 70% площади всего кристалла процессора. В качестве примера можно привести компьютеры IBM System/360 и Digital Equipment Corporation VAX, обладающие весьма развитым набором сложных машинных команд. Вскоре поддерживать все эти команды стало трудно, и разработчикам пришла мысль заменить жесткую логику микропрограммированием.

Впервые конструкция УУ с микропрограммным управлением была предложена американским ученым Моррисом Вилкесом в 1951 году. Идея состояла в том, что каждой инструкции машинного кода соответствует некоторый набор микрокоманд, которые хранятся в специальной памяти, называемой *управляющей памятью*. Таким образом, *устройство управления с микропрограммированием* — это цифровой автомат, закон функционирования которого определяется управляющей памятью.

Микропрограммирование имеет следующие преимущества перед жесткой логикой:

- простота разработки УУ;
- возможность построения сколь угодно сложных команд;
- меньшее количество логических вентилях, а следовательно, уменьшение площади кристалла и тепловыделения;

- возможность быстрого исправления ошибок проектирования процессора: нет необходимости в физической модификации микросхемы, достаточно запрограммировать в управляющую память новый набор микрокоманд;

- возможность специализации МП под конкретную задачу.

Микропрограмма состоит из последовательности микроинструкций. Каждая инструкция формирует набор низкоуровневых управляющих сигналов. Например, типовая операция сложения двух операндов может быть представлена следующим набором микроопераций:

- 1) перенести в регистр *A* РАЛУ значение из РОН 1;
- 2) перенести в регистр *B* РАЛУ значение из РОН 4;
- 3) выполнить операцию сложения в РАЛУ;
- 4) сохранить результат РАЛУ в РОН 2;
- 5) сохранить флаги состояния РАЛУ (нулевой результат, переполнение и т. д.) в регистре состояния;
- 6) перейти к следующей машинной команде.

Упрощенная схема устройства микропрограммного управления (УМУ) представлена на рис. 2.1.3.

Устройство формирования адреса (УФА) вырабатывает адрес следующей микрокоманды в зависимости от внешних сигналов, поступающих от ОУ и от текущей микрокоманды. Код микрокоманды из управляющей памяти подается в регистр микрокоманд. Регистр состоит из двух частей: операционной, в которой содержатся управляющие сигналы, и адресной, которая поступает в УФА для управления загрузкой следующей микрокоманды (рис. 2.1.4).

УМУ можно классифицировать по нескольким признакам. Один из них — по способу представления операционной части регистра микрокоманд. По этому признаку различают УМУ с горизонтальным, вертикальным и смешанным микропрограммированием.

При *горизонтальном микропрограммировании* каждому разряду операционной части регистра микрокоманд соответствует один определенный управляющий сигнал из набора Y (рис. 2.1.5). Если N — количество управляющих сигналов, которое необходимо реализовать, то разрядность операционной части регистра микрокоманд будет также равна N . Для этого типа микропрограммирования характерна очень большая разрядность управляющей памяти (более сотни битов) и, соответственно, большой объем микрокода. В то же время такая реализация проще со схемотехнической точки зрения.



Рис. 2.1.3. Упрощенная схема устройства микропрограммного управления

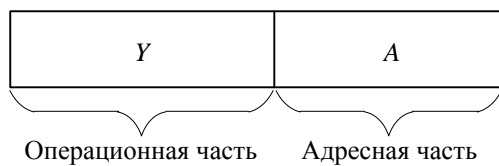


Рис. 2.1.4. Регистр микрокоманд

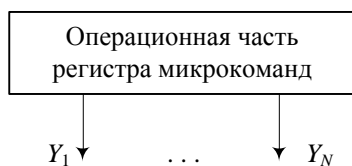


Рис. 2.1.5. Горизонтальное микропрограммирование

При *вертикальном микропрограммировании* операционная часть регистра микрокоманд хранится в памяти в закодированном виде и подвергается дешифрации перед подачей в операционные устройства МП (рис. 2.1.6). Достоинством этого типа микропрограммирования является уменьшение требований к объему памяти для микрокода. При этом сложность микропроцессора увеличивается, а быстродействие уменьшается в связи с появлением дополнительных комбинационных схем — дешифраторов.

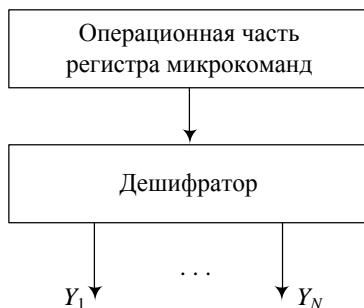


Рис. 2.1.6. Вертикальное микропрограммирование

Смешанное микропрограммирование сочетает достоинства горизонтального и вертикального. При этом набор управляющих сигналов Y разбивается на несколько поднаборов. Каждый поднабор управляет отдельным ОУ либо устройствами, которые совместно используются редко. Далее на каждый поднабор устанавливается свой дешифратор (рис. 2.1.7). Как результат, вследствие некоторого допустимого усложнения схемы УУ уменьшается объем микрокода.

Развитие технологий производства микросхем, удешевление изготовления внутренней памяти, а также необходимость увеличения частот МП привели к тому, что в настоящее время вертикальное микропрограммирование практически вытеснено горизонтальным и смешанным микропрограммированием.

Другой распространенный признак классификации УМУ — по способу формирования адреса следующей микрокоманды. Различают УМУ с принудительной адресацией и с естественной.

В УМУ с *принудительной адресацией* (рис. 2.1.8) каждая микрокоманда содержит адресную часть, которая управляет порядком

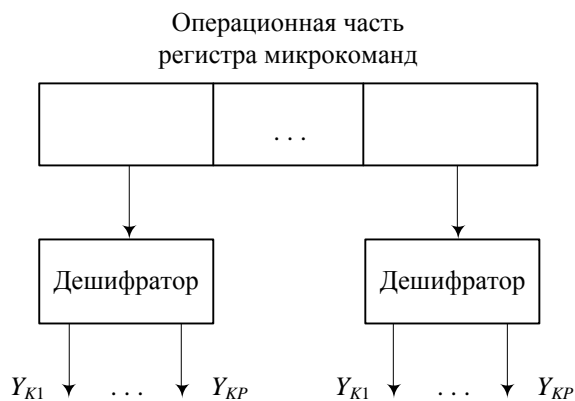


Рис. 2.1.7. Смешанное микропрограммирование

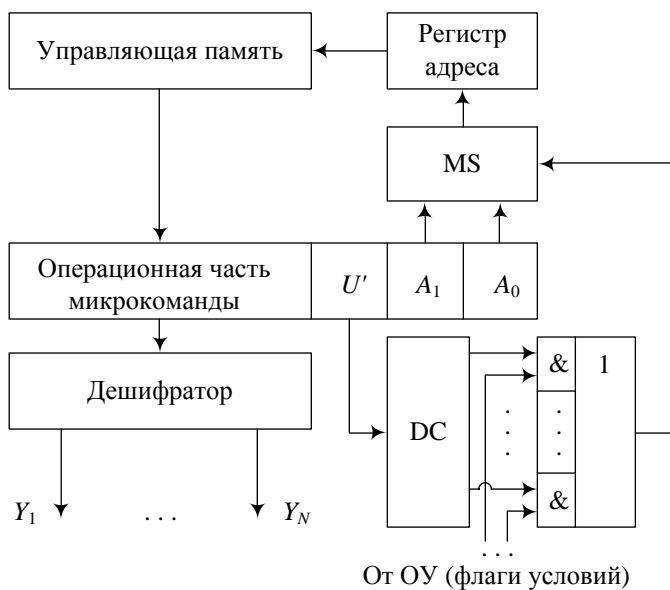


Рис. 2.1.8. Схема УМУ с принудительной адресацией

следования микрокоманд. Адресная часть в свою очередь содержит поле логических условий U' , служащее маской для набора внешних логических условий U , и два поля адреса A_0 и A_1 следующей микрокоманды. Поле U' совместно с флагами от операционных устройств управляет мультиплексором выбора адреса следующей микрокоманды. На мультиплексор будет подана единица, только если в позиции соответствующих коду условий будет единица, и при этом условия были выполнены. В таком случае выполняется ветвление, и в регистр адреса будет загружен A_1 . Если условия не выполнены, то ветвления нет, и используется поле A_0 .

В УМУ с естественной адресацией (рис. 2.1.9) применяется инкрементирующий счетчик адреса следующей микрокоманды. Если содержащийся в микрокоманде признак P равен нулю, то линейный порядок следования микрокоманд не изменяется и адрес просто инкрементируется в счетчике. Если P равен единице, то операционная часть микрокоманды выступает в роли нового адреса, который загружается в счетчик. Таким образом реализуется механизм ветвлений.

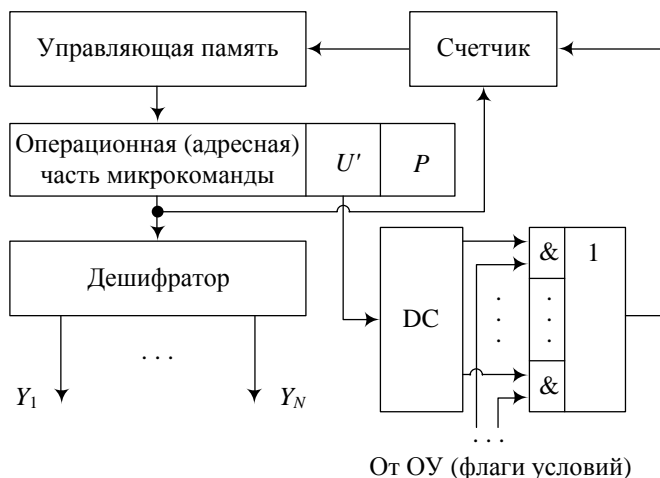


Рис. 2.1.9. Схема УМУ с естественной адресацией

С появлением языков программирования высокого уровня, таких как С, С++ и др., проблема сложности системы команд процессо-

ра утратила свою остроту, поскольку теперь компилятор занимается непосредственной генерацией машинного кода. В связи с этим команды процессоров стали упрощать и УУ на жесткой логике снова оказались востребованными. На сегодняшний день большинство специализированных микропроцессоров, сигнальных процессоров, микроконтроллеров и синтезируемых процессоров используют УУ на жесткой логике. В то же время высокопроизводительные процессоры, в том числе процессоры общего назначения, с большим количеством сложных команд применяют УУ с микропрограммным управлением.

Контрольные вопросы

1. Дайте определение устройства управления.
2. Какие бывают устройства управления?
3. Перечислите достоинства и недостатки УУ с микропрограммным управлением.
4. Расскажите о вертикальном, горизонтальном и смешанном микропрограммировании.
5. Какие функции выполняет устройство формирования адреса?

Литература

1. Микропроцессоры. В 3-х кн. Кн. 1. Архитектура и проектирование микроЭВМ: учебник для вузов / **Под ред. Л.Н. Преснухина**. — М.: Высшая школа, 1986. — 495 с.
2. Микропроцессорные системы: учеб. пособие для вузов / **Е.К. Александров, Р.И. Грушвицкий, М.С. Куприянов и др.; под ред. Д.В. Пузанкова**. — СПб.: Политехника, 2002. — 935 с.