

Надежность – это способность системы выполнять полностью и правильно все заданные ей функции. Надежность ПК измеряется обычно средним временем наработки на отказ.

16. Стоимость.

17. Габариты и масса.

3.6. Система команд микропроцессора

Решение задач на ЭВМ реализуется программным способом, т. е. путем выполнения последовательно во времени отдельных операций над информацией, предусмотренных алгоритмом решения задачи.

Алгоритм – это точно определенная последовательность действий, которые необходимо выполнить над исходной информацией, чтобы получить решение задачи.

Алгоритм решения задачи, заданный в виде последовательности команд на языке вычислительной машины (в кодах машины), называется *машинной программой* [2, 6].

Проектирование системы команд оказывает влияние на структуру ЭВМ. Оптимальную систему команд иногда определяют как совокупность команд, которая удовлетворяет требованиям проблемно-ориентированных применений таким образом, что избыточность аппаратных и аппаратно-программных средств на реализацию редко используемых команд оказывается минимальной. В различных программах ЭВМ частота появления команд различна; например, по данным фирмы DEC в программах для ЭВМ семейства PDP-11 наиболее часто встречается команда передачи MOV(B), на ее долю приходится приблизительно 32 % всех команд в типичных программах. Систему команд следует выбирать таким образом, чтобы затраты на редко используемые команды были минимальными.

При наличии статистических данных можно разработать (выбрать) ЭВМ с эффективной системой команд. Одним из подходов к достижению данной цели является разработка команд длиной в одно слово и кодирование их таким образом, чтобы разряды таких коротких команд использовать оптимально, что позволит сократить время реализации программы и ее длину.

Другим подходом к оптимизации системы команд является использование микроинструкций. В этом случае отдельные биты или группы бит команды используются для кодирования нескольких элементарных операций, которые выполняются в одном командном цикле.

Эти элементарные операции не требуют обращения к памяти, а последовательность их реализации определяется аппаратной логикой.

Сокращение времени выполнения программ и емкости памяти достигается за счет увеличения сложности логики управления.

Важной характеристикой команды является ее формат, определяющий структурные элементы команды, каждый из которых интерпретируется определенным образом при ее выполнении. Среди таких элементов (полей) команды выделяют следующие: *код операции*, определяющий выполняемое действие; *адрес* ячейки памяти, регистра процессора, внешнего устройства; режим адресации; операнд при использовании непосредственной адресации; код анализируемых признаков для команд условного перехода [2, 6].

Классификация команд по основным признакам представлена на рис. 6.



Рис. 6. Классификация команд

Важнейшим структурным элементом формата любой команды является *код операции* (КОП), определяющей действие, которое должно быть выполнено. Большое число КОП в процессоре очень важно, так как аппаратная реализация команд экономит память и время. Но при

выборе ЭВМ необходимо концентрировать внимание на полноте операций с конкретными типами данных, а не только на числе команд, на доступных режимах адресации. Число бит, отводимое под КОП, является функцией полного набора реализуемых команд [3, 6].

При использовании фиксированного числа бит под КОП для кодирования всех команд необходимо в поле КОП выделить m двоичных разрядов. Однако, учитывая ограниченную длину слова мини- и микроЭВМ, различное функциональное назначение команд, источники и приемники результатов операций, а также то, что не все команды содержат адресную часть для обращения к памяти и периферийным устройствам, в малых ЭВМ для кодирования команд широко используется принцип кодирования с переменным числом бит под поле КОП для различных групп команд.

В некоторых командах необходим только один операнд и они называются *однооперандными* (или одноадресными) командами в отличие от *двухоперандных* (или двухадресных), в которых требуются два операнда. При наличии двух операндов командой обычно изменяется только один из них. Так как информация берется только из одной ячейки, эту ячейку называются *источником*; ячейка, содержимое которой изменяется, называется *приемником*.

3.6.1. Структура и виды команд

Команда машинной программы (иначе, машинная команда) – это элементарная инструкция машине, выполняемая ею автоматически без каких-либо дополнительных указаний и пояснений.

Машинная команда состоит из двух частей: операционной и адресной.

Операционная часть команды – это группа разрядов в команде, предназначенная для представления кода операции машины.

Адресная часть команды – это группа разрядов в команде, в которой записываются коды адреса (адресов) ячеек памяти машины, предназначенных для оперативного хранения информации или иных объектов, задействованных при выполнении команды. Часто эти адреса называются адресами операндов, т. е. чисел, участвующих в операции.

По количеству адресов, записываемых в команде, команды делятся на безадресные, одно-, двух- и трехадресные.

Типовая структура трехадресной команды [2, 3, 6]:

КОП a_1, a_2, a_3 ,

где a_1 и a_2 – адреса ячеек (регистров), где расположены, соответственно, первое и второе числа, участвующие в операции;

a3 – адрес ячейки (регистра), куда следует поместить число, полученное в результате выполнения операции.

Типовая структура двухадресной команды:

КОП a1, a2,

где a1 – это обычно адрес ячейки (регистра), где хранится первое из чисел, участвующих в операции, и куда после завершения операции должен быть записан результат операции;

a2 – обычно адрес ячейки (регистра), где хранится второе участвующее в операции число.

Типовая структура одноадресной команды:

КОП a1,

где a1 в зависимости от модификации команды может обозначать либо адрес ячейки (регистра), где хранится одно из чисел, участвующих в операции, либо адрес ячейки (регистра), куда следует поместить число результат операции.

Безадресная команда содержит только код операции, а информация для нее должна быть заранее помещена в определенные регистры машины (безадресные команды могут использоваться только совместно с командами другой адресности).

Пример. Поступила представленная на языке символического кодирования команда:

СЛ 0103, 5102.

Такую команду следует расшифровать так: "сложить число, записанное в ячейке 0103 памяти, с числом, записанным в ячейке 5102, а затем результат (т. е. сумму) поместить в ячейку 0103".

3.6.2. Особенности программного и микропрограммного управления

В микропроцессорах используют два метода выработки совокупности функциональных управляющих сигналов: программный и микропрограммный [2, 3, 6].

Выполнение операций в машине сводится к элементарным преобразованиям информации (передача информации между узлами в блоках, сдвиг информации в узлах, логические поразрядные операции, проверка условий и т. д.) в логических элементах, узлах и блоках под воздействием функциональных управляющих сигналов блоков (устройств) управления. Элементарные преобразования, неразложимые на более простые, выполняются в течение одного такта сигналов синхронизации и называются микрооперациями.

В аппаратных (схемных) устройствах управления каждой операции соответствует свой набор логических схем, вырабатывающих определенные функциональные сигналы для выполнения микроопераций в определенные моменты времени. При этом способе построения устройства управления реализация микроопераций достигается за счет однажды соединенных между собой логических схем, поэтому ЭВМ с аппаратным устройством управления называют ЭВМ с жесткой логикой управления. Это понятие относится к фиксации системы команд в структуре связей ЭВМ и означает практическую невозможность каких-либо изменений в системе команд ЭВМ после ее изготовления.

При микропрограммной реализации устройства управления в состав последнего вводится ЗУ, каждый разряд выходного кода которого определяет появление определенного функционального сигнала управления. Поэтому каждой микрооперации ставится в соответствие свой информационный код – микрокоманда. Набор микрокоманд и последовательность их реализации обеспечивают выполнение любой сложной операции. Набор микроопераций называют микропрограммами. Способ управления операциями путем последовательного считывания и интерпретации микрокоманд из ЗУ (наиболее часто в виде микропрограммного ЗУ используют быстродействующие программируемые логические матрицы), а также использования кодов микрокоманд для генерации функциональных управляющих сигналов называют микропрограммным, а микроЭВМ с таким способом управления – микропрограммными или с хранимой (гибкой) логикой управления.

К микропрограммам предъявляют требования функциональной полноты и минимальности. Первое требование необходимо для обеспечения возможности разработки микропрограмм любых машинных операций, а второе связано с желанием уменьшить объем используемого оборудования. Учет фактора быстродействия ведет к расширению микропрограмм, поскольку усложнение последних позволяет сократить время выполнения команд программы.

Преобразование информации выполняется в универсальном арифметико-логическом блоке микропроцессора. Он обычно строится на основе комбинационных логических схем.

Для ускорения выполнения определенных операций вводятся дополнительно специальные операционные узлы (например, узел циклического сдвига). Кроме того, в состав микропроцессорного комплекта (МПК) БИС вводятся специализированные оперативные блоки арифметических расширителей.

Операционные возможности микропроцессора можно расширить за счет увеличения числа регистров. Если в регистровом буфере закре-

пление функций регистров отсутствует, то их можно использовать как для хранения данных, так и для хранения адресов. Подобные регистры микропроцессора называются регистрами общего назначения (РОН). По мере развития технологии реально осуществлено изготовление в микропроцессоре 16, 32 и более регистров.

В целом же, принцип микропрограммного управления (ПМУ) включает следующие позиции:

1) любая операция, реализуемая устройством, является последовательностью элементарных действий – микроопераций;

2) для управления порядком следования микроопераций используются логические условия;

3) процесс выполнения операций в устройстве описывается в форме алгоритма, представляемого в терминах микроопераций и логических условий, называемого микропрограммой;

4) микропрограмма используется как форма представления функции устройства, на основе которой определяются структура и порядок функционирования устройства во времени.

ПМУ обеспечивает гибкость микропроцессорной системы и позволяет осуществлять проблемную ориентацию микро- и миниЭВМ.