

ТЕМА 5: Система команд микропроцессора

Содержание:

1. **Набор команд микропроцессора**
2. Мнемоническая запись команды
3. Способы адресации

Команда микропроцессора

Команда микропроцессора – это такое двоичное слово, которым обозначается его элементарное действие. Другие, отличные от команд двоичные слова полезных действий в микропроцессоре вызывать не могут.

Большинство команд осуществляет пересылку или обработку данных, расположенных в памяти или в одном из регистров микропроцессора. Некоторые команды предназначены для управления вспомогательными функциями микропроцессора, поддерживающими необходимый режим его работы.

Всю совокупность команд микропроцессора называют набором команд.

Длина команды как двоичного слова может совпадать с длиной слова данных или быть кратной ей. Так, длина команды 8-разрядного микропроцессора может равняться 8, 16 или 24 битам.

Загрузка и декодирование команд

Для выполнения команда загружается в регистр команд. В дешифраторе команда идентифицируется, и схема управления, формирует сигналы, направляемые в другие модули микропроцессора. С помощью этих сигналов выполняются операции, предписываемые командой.

Микропроцессор загружает команду в регистр команд в течение **цикла выборки**. В течение следующего за ним **цикла выполнения** команда декодируется и формируются сигналы управления процессом выполнения операций этой команды.

Команда микропроцессора должна содержать **информацию двух видов**:

во-первых, команда должна сообщать микропроцессору, что делать (выполнять сложение, очистку, пересылку, сдвиг и т.п.);

во-вторых, указывать адрес, т. е. местоположение обрабатываемых данных.

Структура команды

Команда состоит из двух частей:

- ◆ *код операции (КОП),*
- ◆ *адрес операнда.*

Код операции сообщает микропроцессору, что делать.

Адрес указывает местоположение данных, участвующих в операции.

Число команд обычно превышает число различных видов операций, Это связано с тем, что при формировании команды одно и тоже действие может задаваться различными кодами в соответствии со способом указания места расположения операнда, участвующего в операции, т.е. от способа адресации.



Содержание:

1. Набор команд микропроцессора
2. **Мнемоническая запись команды**
3. Способы адресации

Мнемоническая запись команды

Команда микропроцессора – это двоичное число. Восемьмеричную и шестнадцатеричную системы счисления используют для краткой записи длинных двоичных чисел, а, следовательно, и команд микропроцессора. Однако при этом возникает основная проблема: трудно запомнить что означает каждая команда, выраженная в подобной форме.

Данная проблема решается путем применения **мнемонического обозначения** – сокращенной записи названия команды. Для этой цели обычно используются три буквы названия операции, выполняемой командой.

Например, мнемоническое обозначение команды очистки имеет следующий вид: **CLA**.

Команда очистки аккумулятора будет записываться как **CLA A**, где **CLA** – код операции, а **A** - адрес размещения обрабатываемых данных.

Мнемоническая запись команды

Если же команда оперирует числовыми данными или адресами областей памяти, то в адресной части команды целесообразно использование чисел.

Например, код операции с мнемоническим обозначением **JMP** требует указания адреса перехода. Подобная команда может иметь вид **JMP 177756₈**, где адрес выражен 6-разрядным восьмеричным числом, двоичный эквивалент которого **1 111 111 111 101 110₂** есть адрес области памяти. Мнемоническое обозначение кода операции **JMP** легче запомнить, чем его восьмеричный эквивалент **3038**.

Сочетание сокращенного буквенного обозначения кода операции с числовой формой записи адреса является одной из наиболее удобных форм записи команды.

Такой способ записи программ составляет основу языка **ассемблер**.

Написанная на ассемблере программа обрабатывается компилятором, который преобразует мнемоническое обозначение кодов операции в соответствующие двоичные эквиваленты.



Содержание:

1. Набор команд микропроцессора
2. Мнемоническая запись команды
3. **Способы адресации**

Адресация операнда

8-разрядные микропроцессорные системы, как правило, имеют 16-разрядную шину адреса (2 байта), и могут работать с объемом памяти 64 Кбайт (**65536** ячеек).

В зависимости от того, как в команде микропроцессора задается место расположения операнда, участвующего в операции, длина команды может составлять **один, два** или **три** байта. Способ доступа к операнду называют способом адресации.

Существует **четыре** базовые способы адресации :

1- **неявная** (численное значение адреса операнда в команде не указывается);

2- **непосредственная** (значение операнда указано в команде);

3- **прямая** (адрес указан в команде);

4- **косвенная** (численное значение адреса содержится в регистре микропроцессора).

В описании системы команд конкретного микропроцессора может быть указано гораздо больше способов адресации. Однако их многообразие получается за счет отдельных нюансов реализации перечисленных выше четырех способов.

Неявная адресация

Особенностью неявной адресации является то, что операнды располагаются не в системной памяти компьютера, а во внутренних регистрах микропроцессора.

В результате для записи команды достаточно одного байта, в котором одна часть битов обозначает код операции, а другая – номера (или адреса) регистров, задействованных в команде.

Например, 1-байтовая команда пересылки данных из регистра А в регистр В состоит из кода операции (01), адреса (111) источника данных – регистра А и адреса (000) приемника данных – регистра В.



Адреса источника и приемника указаны в команде неявно. Иногда говорят, что они «встроены» в команду. Вот почему такая адресация называется неявной.

Однобайтовая команда выполняется за два микроцикла: первый – выборка команды, второй – ее выполнение.

Непосредственная адресация

Команды с непосредственной адресацией имеют длину 2 байта:

- в первом байте содержится код операции,
- во втором байте – данные.



Примером команды с непосредственной адресацией является загрузка аккумулятора 8-разрядным двоичным числом.

Команда с непосредственной адресацией выполняется микропроцессором за два микроцикла:

- ◆ в течение первого - производится выборка команды,
- ◆ в течение второго - ее выполнение.

При использовании непосредственной адресации данные размещаются в памяти программ, т.е. там же, где находится сама команда.

Прямая адресация

Команды с прямой адресацией могут иметь длину, равную 2 или 3 байтам:

- ◆ **первый байт** предназначен для кода операции,
- ◆ **второй и третий** (если имеется) – для адреса.

Совместное использование второго и третьего байтов команды позволяет адресоваться к любой из 65536 ячеек памяти.

Прямая адресация, как правило, представляется наиболее естественной - программист имеет возможность, явным образом задавая адрес необходимых данных.

Примером использования такой адресации может служить команда записи содержимого аккумулятора в память по адресу 0x000E:

КОП	0x00	0x0E
1-й байт	2-й байт	3-й байт

Адрес, заданный в шестнадцатеричной форме, занимает второй и третий байты 3-х байтовой команды.

Прямая адресация *продолжение*

3-х байтные команды с прямой адресацией выполняются за 4 микроцикла:

- 1** - выборка кода операции команды и его декодирование;
- 2** - извлечение из памяти старшего байта адреса операнда обрабатываемых данных (при каждом дополнительном обращении к памяти требуется один микроцикл);
- 3** - извлечение из памяти младшего байта адреса операнда
- 4** - выполнение команды.

Таким образом, время выполнения команд с прямой адресацией в два раза больше, чем команд с непосредственной адресацией.

Прямую адресацию целесообразно использовать в тех случаях, когда необходимо иметь возможность размещать данные в любой области доступной микропроцессору памяти. Поэтому при написании программы рекомендуется использовать как можно меньше команд с прямой адресацией, так как для их выполнения требуется больше времени.

Прямая адресация *продолжение*

Прямая адресация может реализовываться за три микроцикла. В таких случаях команда имеет длину 2 байта:

- первый – для кода операции,
- второй – для адреса операнда.

Однобайтовый адрес в позволяет обращаться только к одной из 256 ячеек в начале массива памяти. Этот участок памяти называют **нулевой страницей памяти**.

Преимущество команд с 1-байтовыми адресами по сравнению с 2-байтовыми заключается в меньших затратах времени на их выполнение, поскольку при этом «экономится» один микроцикл.

Команды с 1-байтовыми адресами целесообразно применять в тех случаях, когда требуется многократно обращаться к сравнительно небольшому количеству байтов данных, размещающихся в нулевой странице памяти.

2-х байтовые команды с прямой адресацией называют иногда командами **прямой адресации** (или командами **адресации нулевой страницы**), а **3-байтовые** – командами **расширенной прямой адресации**.

Косвенная адресация

Большинство микропроцессоров располагает еще одним способом адресации к памяти, реализуемой командами длиной **один байт**. Такая адресация называется **косвенной или иногда косвенной регистровой**.

Помимо **кода операции** в такой команде указывается **номер регистра**, содержимое которого является адресом местоположения данных в памяти.

Косвенная адресация дает наибольший эффект при обработке данных, расположенных в следующих друг за другом ячейках памяти.

Примером может служить задача организации файла данных в памяти микропроцессорной системы. Из порта ввода-вывода данные загружаются в аккумулятор, а затем записываются в следующие друг за другом ячейки памяти. В основу организации этого файла положен принцип косвенной регистровой адресации. Каждому пронумерованному блоку алгоритма соответствует одна команда.

Косвенная адресация *продолжение*

