1. Программная модель процессоров семейства Intel

Программная модель – это минимальный набор сведений о базовом процессоре, необходимый для понимания особенностей языка ассемблера. Это набор сведений включает в себя:

- организацию оперативной памяти
- доступные программисту регистры процессора
- базовые типы данных
- форматы машинных команд

Рассмотрим эти вопросы более подробно.

1.1. Организация оперативной памяти

Оперативная память организована как набор байтов, каждый из которых имеет уникальный адрес в виде целого числа в диапазоне от 0 до некоторого N. Каждый бит внутри байта имеет свой номер от 0 до 7, причем младшие биты имеют меньшие номера. Нумерация битов в байте – справа налево.

биты	0	1	1	0	1	0	1	1
номер бита	7	6	5	4	3	2	1	0

Два соседних байта образуют слово с номерами битов от 0 до 15. Четыре соседних байта могут рассматриваться как единое целое - двойное слово, с номерами битов от 0 до 63.

Максимальный размер адресного пространства определяется шиной адреса, которая для наиболее распространенных типов процессоров в настоящее время имеет 36 разрядов, что позволяет адресовать до 64 Гб памяти.

1.2. Доступные регистры процессора

Прежде всего необходимо выделить 8 регистров общего назначения (РОН), которые очень часто используются в ассемблерных программах. Все регистры четырехбайтовые и для удобства задаются в программах своими именами: EAX, EBX, ECX, EDX, EBP, ESI, EDI, ESP. Довольно часто

специализация РОНов никак не проявляется и они могут использоваться достаточно произвольно, но в некоторых случаях возникает вполне конкретная специализация регистров. С подобной специализацией нам придется столкнуться в дальнейшем при изучении некоторых команд.

При необходимости вместо четырехбайтовых регистров можно использовать двухбайтовые с именами AX, BX, CX, DX, BP, SI, DI, SP. Более того, каждый из первых четырех регистров можно разбить еще на две однобайтовые составляющие: AH и AL, BH и BL, CH и CL, DH и DL. В этих именах символ L (т.е. Low) используется для обозначения самого младшего байта регистра, а символ H (т.е. High) – для обозначения старшего байта. Тем самым, первые четыре РОНа имеют следующую структуру:

			AX	
Регистр EAX			AH	AL
	31 16		15 8	7 0

			BX	
Регистр EBX			BH	BL
	31 16		15 8	7 0

			CX	
Регистр ECX			CH	CL
	31 16		15 8	7 0

			DX	
Регистр EDX			DH	DL
	31 16		15 8	7 0

Остальные четыре регистра имеют следующую структуру:

Регистр EBP		BP

	31 16	15 0

Регистр ESI

	SI
31 16	15 0

Регистр EDI

	DI
31 16	15 0

Регистр ESP

	SP
31 16	15 0

Кроме регистров общего назначения обязательно надо отметить еще два очень важных регистра.

Регистр EIP и его младшая половина с именем IP (т.е. Instruction Pointer, указатель команд) используются для хранения адреса следующей подлежащей выполнению машинной команды. Этот регистр активно используется при организации переходов между командами и при вызове подпрограмм.

Регистр EFlags (младшая половина имеет имя Flags) называется флаговым и содержит набор отдельных битов-флагов, в каждый момент времени определяющих текущее состояние выполняемой программы. Для удобства использования большинство флагов имеют свои имена, которые можно использовать в ассемблерных программах. Имена некоторых флагов будут приведены в последующем материале пособия.

1.3. Базовые типы данных

Базовые типы реализованы на уровне самого процессора и включают следующие основные типы.

а) Целые беззнаковые числа длиной в 1 байт, 2 байта (слово), 4 байта (двойное слово) и 8 байтов (учетверенное слово).

Особенностью внутреннего представления чисел длиной более 1 байта является их «перевернутое» хранение: по более младшим адресам памяти записываются более младшие байты, затем старшие. Например, двухбайтовое целое 16-ричное число 12AB вместо ожидаемого естественного представления 12 AB будет записано в память как AB 12, а четырехбайтовое число 12 34 AB CD будет сохранено как CD AB 34 12.

б) Целые знаковые числа представляются с использованием дополнительного кода и тоже имеют «перевернутое» внутреннее представление.

в) Символьные данные как обычно представляются в виде внутренних кодов символов по1 или 2 байта (для кодировки Unicode)

г) Вещественные числа представляются в виде мантиссы и порядка

д) Двоично-десятичные числа (BCD, т.е. binary coded decimal) – это прямая замена десятичных цифр двоичными эквивалентами, например, число 1234_{10} = 0001 0010 0011 0100.

1.4. Форматы машинных команд

Формат команды определяет ее внутреннюю структуру, т.е. закодированную информацию, необходимую процессору для выполнения конкретного элементарного действия. Основные составляющие машинной команды – это код операции и один-два операнда (иногда встречаются и безоперандные команды).

Код операции – это набор битов, определяющих действие, выполняемое данной командой. Число используемых битов определяет общее количество возможных команд. Например, если для кода команды используется 1 байт, то система команд может содержать до 256 различных команд.

Операнды команды каким-то образом определяют объекты, с которыми должно быть выполнено данное действие. Важнейшими типами операндов являются:

- **непосредственный**: объект (например - константа) задается **внутри** самой команды и **не требует** обращения к памяти; этот способ

наиболее быстрый, но в то же время наименее гибкий, т.к. для изменения операнда придется изменять саму команду

- **регистровый**: в команде задается имя (номер) регистра, содержащего необходимый операнд; этот способ также очень быстрый, но число регистров очень небольшое и поэтому их использование требует многократного обращения к памяти
- **адресный**: в команде задается адрес области памяти, где находится операнд; при этом могут использоваться разные способы формирования этого адреса (прямой, индексный, косвенный).

В зависимости от комбинации типов операндов можно выделить следующие разновидности двухоперандных команд:

- регистр - непосредственный операнд
- регистр - регистр
- регистр - память
- память - непосредственный операнд.

Важным параметром машинных команд является их длина. Длина может изменяться от 1 байта до 10 байт, хотя большинство чаще всего используемых команд имеют длину от трех до пяти байтов