Способы адресации процессора Pentium 4 нерегулярны и зависят от того, в каком формате находятся команды: 16- или 32-битном.

Режим адресации определяют один или два байта адресации, которые находятся после байта адресации. Первым является байт *mod r/m*, вторым байт *sib* (масштаб, индекс, база). Байт *sib* может присутствовать только в командах с 32-разрядной адресацией, когда байт *mod r/m* содержит *r/m* = 100 и значение в поле *mod*≠11.

Для адресации теперь можно использовать любой регистр общего назначения, а индекс разрешается масштабировать (умножать) на 1, 2, 4 или 8. Ниже приводятся все способы 32–разрядной адресации.

Таблица 4.3

|  |  |  |  |
| --- | --- | --- | --- |
| Непосредственная | mov | eax, | 12345678h |
| Регистровая | mov | eax, | ecx |
| Прямая (абсолютная) | mov | eax, | [3456789h] |
| Регистровая косвенная | mov | eax, | [ecx] |
| Базовая (индексная) со смещением | mov | eax, | [ecx]+1200h |
| Базовая индексная со смещением | mov | eax, | [ecx][edx]+40h |
| Индексная с масштабированием и смещением | mov | eax, | [eci\*4]+400h |
| Базовая индексация с масштабированием | mov | eax, | [edx][ecx\*8] |
| Базовая индексация с масштабированием и смещением | mov | eax, | [ebx][edi\*2]+20h |

Проблема заключается в том, что не все способы адресации применимы ко всем командам и не все регистры могут использоваться при всех способах адресации. Это существенно усложняет работу компилятора.

Схема формирования адреса в 32-разрядной адресации приведена на рис 4.24

32–разрядные режимы адресации кодируются с помощью двух байт mod r/m и sib.

Байт *mod* управляет способами адресации. Один из операндов определяется по комбинации полей mod и r/m. Второй операнд всегда является регистром и определяется по значению поля reg. (таблицы 4.4. и 4.5).

Формирование адреса памяти в 32-битной адресации (r/m ≠100).

Таблица 4.4 M[x] – слово в памяти с адресом х. D -- смещение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| r/m | Адрес памяти второго операнда | | | |
| mod = 00 | mod = 01 | mod = 10 | Mod = 11 |
| 000 | М[EAX] | M[EAX+d8] | M[EAX+d32] | EAX или AL |
| 001 | M[ECX] | M[ECX+d8] | M[ECX+d32] | ECX или CL |
| 010 | M[EDX] | M[EDX+d8] | M[EDX+d32] | EDX или DL |
| 011 | M[EBX] | M[EBX+d8] | M[EBX+d32] | EBX или BL |
| 100 | имеется sib | имеется sib | имеется sib | ESP или AH |
| 101 | Прямая адр. | M[EBP+d8] | M[EBP+d32] | EBP или CH |
| 110 | M[ESI] | M[ESI+d8] | M[ESI+d32] | ESI или DH |
| 111 | M[EDI] | M[EDI+d8] | M[EDI+d32] | EDI или BH |

Колонки 01 и 10 включают способы адресации, при которых значение регистра прибавляется к 8-битному или 32-битному смещению, которое следует за командой.

Иногда вслед за байтом mod следует дополнительный байт sib. Байт sib определяет масштабный коэффициент и два регистра.

Поле SS указывает масштабный коэффициент индекса, поле index определяет любой регистр, кроме ESP, который служит индексным регистром, а поле base – определяет базовый регистр. Формирование адреса памяти в 32-битной адресации (r/m =100, имеется sib) приведено в табл. 4.6.

Таблица 4.6

|  |  |  |  |
| --- | --- | --- | --- |
| поле base | Адрес памяти второго операнда | | |
| mod = 00 | mod = 01 | mod = 10 |
| 000 | M[EAX+ss\*ind] | M[EAX+ss\*ind+d8] | M[EAX+ss\*ind+d32] |
| 001 | M[ECX+ss\*ind] | M[ECX+ss\*ind+d8] | M[ECX+ss\*ind+d32] |
| 010 | M[EDX+ss\*ind] | M[EDX+ss\*ind+d8] | M[EDX+ss\*ind+d32] |
| 011 | M[EBX+ss\*ind] | M[EBX+ss\*ind+d8] | M[EBX+ss\*ind+d32] |
| 100 | M[ESP+ss\*ind] | M[ESP+ss\*ind+d8] | M[ESP+ss\*ind+d32] |
| 101 | M[d32+ss\*ind] | M[D32+ss\*ind+d8] | M[d32+ss\*ind+d32] |
| 110 | M[ESI+ss\*ind] | M[ESI+ss\*ind+d8] | M[ESI+ss\*ind+d32] |
| 111 | M[EDI+ss\*ind] | M[EDI+ss\*ind+d8] | M[EDI+ss\*ind+d32] |

Таблица кодирования полей index и ss.

|  |  |  |  |
| --- | --- | --- | --- |
| index | индексный регистр | ss | Множитель |
| 000 | EAX | 00 | \*1 |
| 001 | ECX | 01 | \*2 |
| 010 | EDX | 10 | \*4 |
| 011 | EBX | 11 | \*8 |
| 100 | нет |  |  |
| 101 | EBP |  |  |
| 110 | ESI |  |  |
| 111 | EDI |  |  |