**Федеральное государственное бюджетное образовательное учреждение высшего образования "Белгородский государственный технологический университет им. В.Г. Шухова"**

Кафедра программного обеспечения вычислительной техники и

автоматизированных систем.

**Лабораторная работа № 2**

Структура команд процессора.

Вариант 13

Выполнил:

Студент группы КБ-211

Коренев Д.Н.

Принял:

Осипов О.В.

*Цель работы:* изучить структуру команд процессора, научиться составлять машинный код простейших команд.

**Задание**

1. Ознакомиться с теоретическим материалом главы 2 учебника В.И. Юрова «Assembler» “Программно-аппаратная архитектура IA-32 процессоров Intel”.
2. В соответствии с вариантом задания определить по символьному описанию команд их машинный код (для 5 команд), а также по машинному коду команд определить их символьное описание (для 2 машинных кодов).

**Вариант 13**

1. MOV AX, 120h
2. CMP DI, AX
3. SUB [EBP\*8], AX
4. ADC CX, [ESI+101000101b]
5. OR BYTE PTR [EBP+EDI+8], 9
6. 8BC8
7. BB 1200

**Выполнение задания**

**Команда 1:** MOV AX, 120h == 66:B8 2001

Команда выполняет пересылку шестнадцатиричного числа 120 в регистр AX. Первый операнд имеет регисровую адресацию, второй является непосредственным операндом. Код операции данной команды MOV = 0xB8. Размер пересылаемых данных равен 2 байтам (машинное слово), значит добавляется префикс 0x66.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Префикс | Код операции (КОП) |  | | | | | SIB | | | Смещение | Непосредственный  операнд |
| d | w | mod | reg/КОП | r/m | scale | index | base |
| 16 бит | 1011100 |  | 0 |  | | | | | | | 120h |
| 1011 1000 | | |
| 0x66 | 0xB8 | | | 0x2001 |

**Команда 2**: CMP DI, AX == 66:3BF8

Команда выполняет пересылку значения из регистра AX в регистр DI. Оба операнда имеют регистровую адресацию. Код операции данной команды CMP = 0x0B. Регистр DI соответствует полю reg = 111, регистр AX соответствует полю r/m = 000. Команда выполняет перемещение 16 бит данных, префикс равен 0x66.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Префикс | Код операции (КОП) |  | | | | | SIB | | | Смещение | Непосредственный  операнд |
| d | w | mod | reg/КОП | r/m | scale | index | base |
| 16 бит | 001110 | 1 | 1 | 11 | 111 | 000 |  | | | | |
| 0011 1011 | | | 1111 1000 | | |
| 0x66 | 0x3B | | | 0xF8 | | |

**Команда 3:** SUB [EBP\*8], AX == 66:2904ED 00000000

Команда выполняет вычитание значения регистра AX из ячейки памяти с адресом [EBP\*8] и результат сохраняется в ячейку памяти с адресом [EBP\*8]. Код операции SUB = 0x29. Второй операнд имеет регистровую адресацию, первый – базово-индексную с масштабированием. Регистру AX соответствует reg = 000. У первого операнда индексная адресация, значит r/m = 100. SIB = 11, 101, 101 = 0xED. Команда выполняет перемещение 16 бит данных, префикс равен 0x66.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Префикс | Код операции (КОП) |  | | | | | SIB | | | Смещение | Непосредственный  операнд |
| d | w | mod | reg/КОП | r/m | scale | index | base |
| 16 бит | 001010 | 0 | 1 | 00 | 000 | 100 | 11 | 101 | 101 | 8 байт |  |
| 0010 1001 | | | 0000 0100 | | | 1110 1101 | | |
| 0x66 | 0x29 | | | 0x04 | | | 0xED | | | 0x00000000 |

**Команда 4:** ADC CX, [ESI+101000101b] == 66:138E 45010000

Команда выполняет сложение значения в ячейке памяти по адресу [ESI+101000101b] и значением регистра CX. Результат помещается в регистр CX. Команда выполняет перемещение 16 бит данных, префикс равен 0x66. Код данной операции ADC = 0x13. За регистр CX отвечает поле reg = 001. Поле r/m = 110 отвечает за базово-индексную адресацию со смещением 0x45010000.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Префикс | Код операции (КОП) |  | | | | | SIB | | | Смещение | Непосредственный  операнд |
| d | w | mod | reg/КОП | r/m | scale | index | base |
| 16 бит | 000100 | 1 | 1 | 10 | 001 | 110 |  | | | 145h |  |
| 0001 0011 | | | 1000 1110 | | | 1 0100 0101 |
| 0x66 | 0x13 | | | 0x8E | | | 0x45010000 |

**Команда 5:** OR BYTE PTR [EBP+EDI+8], 9 == 804C2F0809

Команда выполняет операцию логическое или над байтом в ячейке [EBP+EDI+8] и числом 9h = 0x09. Результат помещается в ячейку памяти по адресу [EBP+EDI+8]. Передается 32 бита, префикса нет. Код данной операции OR = 0x80. За непосредственный операнд отвечает поле reg = 001. За ячейку памяти отвечает поле SIB (r/m = 100) = 00, 101, 111 = 0x2F со смещением 0x08.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Префикс | Код операции (КОП) |  | | | | | SIB | | | Смещение | Непосредственный  операнд |
| d | w | mod | reg/КОП | r/m | scale | index | base |
| 32 бит | 1000000 |  | 0 | 01 | 001 | 100 | 00 | 101 | 111 | 8h | 9h |
| 1000 0000 | | | 0100 1100 | | | 0010 1111 | | |
| нет | 0x80 | | | 0x4C | | | 0x2F | | | 0x08 | 0x09 |

**Машинный код 1:** 8BC8 == MOV ECX,EAX

Префикса нет, значит передается 32 бита данных. КОП = 0x8B соответствует операции MOV r32,r/m32. Поле reg = 001, что соответствует регистру ECX. Поле r/m = 000, что соответствует регистру EAX.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Префикс | Код операции (КОП) |  | | | | | SIB | | | Смещение | Непосредственный  операнд |
| d | w | mod | reg/КОП | r/m | scale | index | base |
| нет | 1000101 |  | 1 | 11 | 001 | 000 |  | | | | |
| 1000 1011 | | | 1100 1000 | | |
| 32 бит | 0x8B | | | 0xC8 | | |
| MOV | | |  | ECX | EAX |

**Машинный код 1:** BB 1200 == MOV EBX,12h

Префикса нет, значит передается 32 бита данных. КОП = 0xBB соответствует операции MOV EBX,r/m32. 12h – непосредственный операнд.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Префикс | Код операции (КОП) |  | | | | | SIB | | | Смещение | Непосредственный  операнд |
| d | w | mod | reg/КОП | r/m | scale | index | base |
| нет | 1011101 |  | 1 |  | | | | | | | 12h |
| 1011 1011 | | |
| 32 бит | 0xBB | | | 0x1200000 |
| MOV EBX, | | | 1200 |

**Вывод:** В ходе лабораторной работы мы изучили структуру команд процессора и научились составлять машинный код простейших команд.