**Методические указания**

**по оформлению контрольной работы**

По дисциплине «ЭВМ и микропроцессоры» учащиеся безотрывной формы обучения выполняют две контрольные работы. Срок сдачи контрольных работ – по учебному графику.

К выполнению контрольной работы рекомендуется приступить после изучения теоретического материала в соответствии с программой и после проработки вопросов для самоконтроля. Литература для изучения указана к каждой теме в подразделе 1.2 данного пособия.

Вариант контрольной работы выбирается в соответствии с последней цифрой индивидуального шифра учащегося.

Контрольная работа выполняется либо рукописным спосо­бом в ученической тетради, либо компьютерным набором (шрифт 12–14, тип Times New Roman, интервал полуторный).

Объем контрольной работы должен соответствовать пол­ному изложению вопросов варианта. Ответ на вопрос должен быть в сжатой, конспективной форме и содержать информацию по данной теме. При необходимости можно привести структур­ные схемы и поясняющие рисунки, графики.

Контрольная работа представлена в десяти вариантах,

.

Контрольная работа содержит задания по теоретическому материалу разделов 4-8 учебной дисциплины и практическое задание.

Контрольная работа состоит из трех заданий. В третьем задании необходимо разработать программу вычисления значения функции:

***y* =**

Задание выполняется в соответствии с исходными данными, представленными в таблице 4. Для написания программы следует использовать систему команд учебной микроЭВМ (таблица 5).

Рассмотрим пример выполнения третьего задания контрольной работы.

В качестве примера рассмотрим программу вычисления значения функции

***y* =**



Т а б л и ц а 3 - Пример программы

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Команда | | Примечание |
| Мнемокод | Код |
| 000 | IN | 01 0 000 | Ввод х |
| 001 | WR 30 | 22 0 030 | Размещение х в ОЗУ(ОЗО) |
| 002 | SUВ #16 | 24 1 016 | Сравнение с границей — (х -16) |
| 003 | JS 010 | 13 0 010 | Переход по отрицательной разности |
| 004 | RD 30 | 21 0 030 | Вычисления по первой формуле |
| 005 | SUB #11 | 24 1 011 |  |
| 006 | WR 31 | 22 0 031 |  |
| 007 | MUL 31 | 25 0 031 |  |
| 008 | SUВ #125 | 24 1 125 |  |
| 009 | JMP 020 | 10 0 020 | Переход на вывод результата |
| 010 | RD 30 | 21 0 030 | Вычисления по второй формуле |
| 011 | MUL 30 | 25 0 030 |  |
| 012 | WR 31 | 22 0 031 |  |
| 013 | RD 30 | 21 0 030 |  |
| 014 | MUL #72 | 25 1 072 |  |
| 015 | ADD 31 | 23 0 031 |  |
| 016 | ADI 106400 | 43 0 000 |  |
| 017 |  | 106400 |  |
| 018 | DIVI 100163 | 46 0 000 |  |
| 019 |  | 100168 |  |
| 020 | OUT | 02 0 000 | Вывод результата |
| 021 | HLT | 09 0 000 | Стоп |

**Вариант 9**

1. Программная модель микропроцессора. Состав и назначение программнодоступных регистров
2. Режимы работы микропроцессоров: реальный, защищенный, виртуальный
3. Разработать программу вычисления и вывода значения функции

|  |  |
| --- | --- |
|  |  |

Т а б л и ц а 5

|  |  |  |  |
| --- | --- | --- | --- |
| КОП | Мнемокод | Название | Действие |
| 00 | NOP | Пустая операция | Нет |
| 01 | IN | Ввод | Асc ← IR |
| 02 | OUT | Вывод | OR ← Асc |
| 03 | IRET | Возврат из прерывания | FLAGS.PC ← M(SP): INC(SP) |
| 04 | WRRB | Загрузка RB | RB ← CR[ADR] |
| 05 | WRSP | Загрузка SP | SP ← CR[ADR] |
| 06 | PUSH | Поместить в стек | DEC(SP); M(SP) ← R |
| 07 | POP | Извлечь из стека | R → M(SP): INC(SP) |
| 08 | RET | Возврат | PC → M(SP); INC(SP) |
| 09 | HLT | Стоп | Конец командных циклов |
| 10 | JMP | Безусловный переход | PC ← CR[ADR] |
| 11 | JZ | Переход, если 0 | if Acc = 0 then PC←CR[ADR] |
| 12 | JNZ | Переход, если не 0 | if Асс ≠ 0 then PC ← CR[ADR] |
| 13 | JS | Переход, если отрицательно | if Асс < 0 then PC ← CR[ADR] |
| 14 | JNS | Переход, если положительно | if Acс > 0 then PC ←CR[ADR] |
| 15 | JO | Переход, если переполнение | if |Acc| > 99999 then PC ← CR[ADR] |
| 16 | JNO | Переход, если нет переполнения | if |Acc| < 99999 then PC ← CR[ADR] |
| 17 | JRNZ | Цикл | DEC(R); if R > 0 then PC ← CR[ADR] |
| 18 | INT | Программное прерывание | DEC(SP); M(SP) ← FLAGS.PC; PC ← M(V) |
| 19 | CALL | Вызов подпрограммы | DEC(SP); M(SP) ← PC; PC ← CR(ADR) |
| 21 | RD | Чтение | Acс ← DD |
| 22 | WR | Запись | M(\*) ← Acс |
| 23 | ADD | Сложение | Acс ← Acс + DD |
| 24 | SUB | Вычитание | Acс ← Acс - DD |
| 25 | MUL | Умножение | Acс ← Acс × DD |
| 26 | DIV | Деление | Acс ← Acc/DD |
| 28 | EI | Разрешить прерывание | IF← 1 |
| 29 | DI | Запретить прерывание | IF←0 |
| 30 | MOV | Пересылка | Rl ←R2 |
| 31 | RD | Чтение | Acс ← R\* |
| 32 | WR | Запись | R\* ← Acс |
| 33 | ADD | Сложение | Acс ← Acс + R\* |
| 34 | SUB | Вычитание | Acс ← Acс - R\* |
| 35 | MUL | Умножение | Acс ← Acс × R\* |
| 36 | DIV | Деление | Acс ← Acc/R\* |
| 37 | IN | Ввод | Acс ← BУ(CR[ADR\*]) |
| 38 | OUT | Вывод | BУ(CR[АDR\*]) ←Асс |
| 41 | RDI | Чтение | Асс ← I |
| 43 | ADI | Сложение | Асс ←Асс + I |
| 44 | SBI | Вычитание | Асс ← Асс - I |
| 45 | MULI | Умножение | Асс ← Асс × I |
| 46 | DIVI | Деление | Асс ← Асс/I |

В табл. 5 приняты следующие обозначения:

DD— данные, формируемые командой в качестве (второго) операнда: прямо или косвенно адресуемая ячейка памяти или трехразрядный непосредственный операнд;

R\* — содержимое регистра или косвенно адресуемая через регистр ячейка памяти;

ADR\* — два младших разряда ADR поля регистра CR;

V — адрес памяти, соответствующий вектору прерывания;

М(\*) — ячейка памяти, прямо или косвенно адресуемая в команде;

I — пятиразрядный непосредственный операнд со знаком.