МИНИСТЕРСТВО НАУКИ И ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное учреждение

Высшего профессионального образования

**«Вятский государственный университет»**

**(ФГБОУ ВПО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

Микропрограммная реализация ЭВМ

Отчет по лабораторной работе №6 дисциплины

«Организация ЭВМ и систем»

Выполнил студент группы ИВТб-31\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Седов М.Д./

Проверил преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Клюкин В.Л./

Киров 2019

1. Задание на лабораторную работу

Подсчитать число одинаковых цифр, расположенных в одноименных разрядах xi и yi 8-разрядных двоичных чисел X и Y.

1. Определение архитектуры и программирование

*Форматы данных.* Данные в примере являются целыми числами, изменяющимися в пределах от 0 до 32767. В этом случае любое число можно представить 16-разрядным двоичным кодом, старший разряд которого определяет знак числа.

*Программно – доступные регистры.* Программно – доступными регистрами МЭ, система команд которой состоит из одноадресных команд, можно считать: шесть регистров общего назначения (r0-r5), программный счетчик – PC (r6), регистр признаков – RP (r7), содержащий разряды двух признаков: нуля (PZ) и знака (PS), а также регистр указателя стека – rSP (r8), регистра адреса таблицы прерываний – ITR(r9), 8-разрядный регистр маски RM.

*Система команд.* Разработка системы команд предполагает определение и кодирование операций, способов адресаций, модификаций и форматов данных.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Наименование | Мнемоника | Описание | Признаки | |
| PZ | PS |
| СУММИРОВАНИЕ | ADD r r\* | r:=r+r\*, PC:=PC+1 | - | - |
| ВЫЧИТАНИЕ | SUB r r\* | r:=r-r\*, PC:=PC+1 | - | - |
| ДОБАВЛЕНИЕ С | AD r C | r:=r+C, PC:=PC+1 | - | - |
| ВЫЧИТАНИЕ С | SB r C | r:=r-C, PC:=PC+1 | + | - |
| ЧТЕНИЕ В РЕГИСТР | LD r A | r:=M[A], PC:=PC+1 | - | - |
| ЗАПИСЬ РЕГИСТРА | MV r A | M[A]:=r, PC:=PC+1 | - | - |
| ЧТЕНИЕ В РЕГИСТР  с индексацией | LDI r (r\*)+ | r:=M[r\*], r\*:=r\*+1;  PC:=PC+1 | - | - |
| ЗАПИСЬ В СТЕК | PUSH r (rSP)- | M[rSP]:=r;  rSP:=rSP-1, PC:=PC+1 | - | - |
| ЧТЕНИЕ ИЗ СТЕКА | POP r (rSP)+ | rSP:=rSP+1;  r:=M[rSP], PC:=PC+1 | - | - |
| ПЕРЕХОД | BR A | PC:=A | - | - |
| ПЕРЕХОД,  ЕСЛИ НУЛЬ | BEQ A | Если PZ=1, то PC:=A,  Иначе PC:=PC+1 | - | - |
| ОБРАЩЕНИЕ К ПОДПРОГРАММЕ | CALL (rSP)- A | M[rSP]:=PC;  rSP:=rSP-1, PC:=A | - | - |
| ОСТАНОВ | HLT A | PC:=A, останов | - | - |
| НЕ ИСКЛЮЧАЮЩЕЕ ИЛИ | NXOR r r\* | r:=^(r xor r\*),  PC:=PC+1 | - | - |
| СДВИГ ВПРАВО | SHR r r\* | r:=r >> 1, PC:=PC+1 | + | - |
| ВОЗВРАТ ИЗ ПРЕРЫВАЮЩЕЙ ПРОГРАММЫ | IRET | rSP:=rSP+1, RM:=M[SP];  rSP:=rSP+1 PC:=M[rSP] | - | - |

Таблица 1 – Система команд

Программа суммирования равных чисел в разрядах двух чисел c использованием символики ассемблера приведена на рисунке 1.

В программе приняты следующие обозначения:

AASP – адрес ячейки памяти, в которой находится адрес начала накопителя стека;

AAM – адрес ячейки памяти, в которой находится начальный адрес массива исходных данных (Y1, X1, Y2, X2, …, YN, XN);

AD – начальный адрес подпрограммы подсчета одинаковых чисел в разрядах;

AS – адрес ячейки, в которую записывается сумма S;

SA – начальный адрес программы суммирования.

|  |  |  |
| --- | --- | --- |
|  | LM AS | Загрузка маски служебной программы |
|  | LD rSP ASP | Загрузка указателя стека SP |
|  | LD rATI ATI | Загрузка адреса таблицы прерывания ATI |
|  | LD PC AP | Загрузка начального адреса программы P |

Рис. 1 – Служебная программа начальных установок

|  |  |  |
| --- | --- | --- |
|  | LM AMP | Загрузка маски основной программы P |
|  | LD r5 AAM | Загрузка адреса массива AM в регистр r5 |
|  | LD r4 AN | Загрузка числа повторений цикла |
|  | SUB r3 r3 | Очистка регистра r3 для суммы S |
| m1 | LDI r1 (r5)+ | Чтение Y в регистр r1 |
|  | ADD r3 r1 | Суммирование |
|  | SB r4 “1” | Вычитание единицы из числа повторений цикла |
|  | BEQ m2 | Если N=0, то переход на метку m2 |
|  | BR m1 | Переход на метку m1 |
| m2 | MV r3 AS | Запись суммы S адресу AS |
|  | HLT SA | Загрузка PC и останов |

Рис. 2 – Основная программа

Распределение программно – доступных регистров ЭВМ показано на рис. 3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Регистры ЭВМ | | |  |
| r0 | X | | | Первое число |
| r1 | Y | | | Второй число |
| r2 |  | | |  |
| r3 | S | | | Сумма S |
| r4 | N | | | Число повторений цикла N |
| r5 | AM | | | Адрес массива AM |
| r6 | PC | | | Программный счетчик |
| r7 | PS | RP | PZ | Регистр признаков |
| r8 | rSP | | | Регистр указателя стека |

Рис. 3 – Распределение регистров ЭВМ

Прерывающая программа P0 приведена на рис. 4.

|  |  |  |
| --- | --- | --- |
|  | LM AM0 | Загрузка маски основной программы P |
|  | PUSH RP | Сохранение в стеке содержимого регистра RP |
|  | PUSH r0 | Сохранение в стеке содержимого регистра r0 |
|  | PUSH r1 | Сохранение в стеке содержимого регистра r1 |
|  | PUSH r2 | Сохранение в стеке содержимого регистра r2 |
|  | SUB r2 r2 | Очистка регистра r2 |
|  | LD r0 AX | Загрузка X в регистр r0 |
|  | LD r1 AY | Загрузка Y в регистр r1 |
|  | NXOR r0 r1 | Не исключающее или регистров r0 и r1 |
| m1 | BEQ m2 | Если PZ=1, то переход на метку m2 |
|  | SHR r0 r0 | Сдвиг регистра r0 на один разряд вправо |
|  | AD r2 “1” | Увеличение на единицу частного Z |
|  | BR m1 | Переход на метку m1 |
| m2 | MV r2 AP | Запись содержимого регистра r2 в РЗУ |
|  | POP r2 | Чтение из стека содержимого регистра r2 |
|  | POP r1 | Чтение из стека содержимого регистра r1 |
|  | POP r0 | Чтение из стека содержимого регистра r0 |
|  | POP RP | Чтение из стека содержимого регистра RP |
|  | IRET | Возврат из программы P0 |

Рис. 4 – Прерывающая программа P0

Команды ЭВМ имеют четыре формата и в зависимости от признака формата (Ф) и кода операции (К) делятся на четыре группы (рис. 5).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ф |  | |  | |  |
| 15 | 14…12 | 11…8 | 7…4 | 3…0 |  |
| 0 | K1 | | r | r\* | ADD, SUB, LDI, PUSH, POP, NXOR, SHR |
|  |  | |  | |  |
| 0 | К2 | | А | | BR, BEQ, HLT |
|  |  | |  | |  |
| 1 | K3 | R | C | | AD, SB |
|  |  | |  | |  |
| 1 | K4 | r | A | | LD, MV, CALL |

Рис. 5 – Форматы команд

Наименования, мнемонические обозначения и коды операция приведены в табл. 2.

Табл. 2 – Коды операций

|  |  |  |
| --- | --- | --- |
| Наименование | Мнемоника | Код операции |
| СУММИРОВАНИЕ | ADD | 15 |
| ВЫЧИТАНИЕ | SUB | 14 |
| ДОБАВЛЕНИЕ КОНСТАНТЫ | AD | 16 |
| ВЫЧИТАНИЕ КОНСТАНТЫ | SB | 17 |
| ЧТЕНИ В РЕГИСТР | LD | 10 |
| ЗАПИСЬ РЕГИСТРА | MV | 12 |
| ЧТЕНИЕ В РЕГИСТР  с индексацией | LDI | 18 |
| ЧТЕНИЕ ИЗ СТЕКА | POP | 1F |
| ПЕРЕХОД | BR | 22 |
| ПЕРЕХОД ЕСЛИ НУЛЬ | BEQ | 21 |
| ОБРАЩЕНИЕ К ПОДПРОГРАММЕ | CALL | 24 |
| ОСТАНОВ | HLT | 00 |
| НЕ ИСКЛЮЧАЮЩЕЕ ИЛИ | NXOR | 27 |
| СДВИГ ВПРАВО | SHR | 28 |
| ЗАГРУЗКА МАСКИ | LM | 30 |
| ВОЗВРАТ ИЗ ПРЕРЫВАЮЩЕЙ ПРОГРАММЫ | IRET | 32 |

Распределение памяти программ и данных приведено на рис. 6.

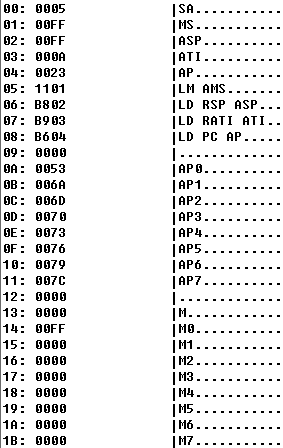


Рис. 6 – Служебная программ и начальные данные

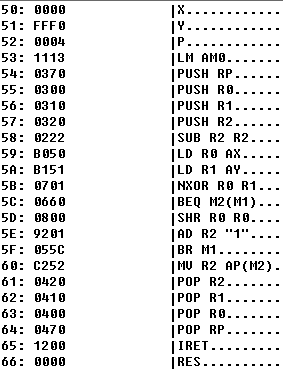


Рис. 7 – Основная программа

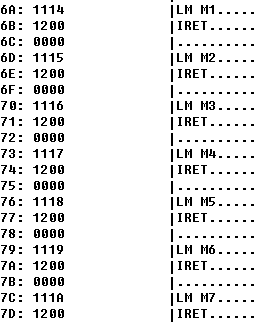


Рис. 8 – Прерывающие программы P1-P7

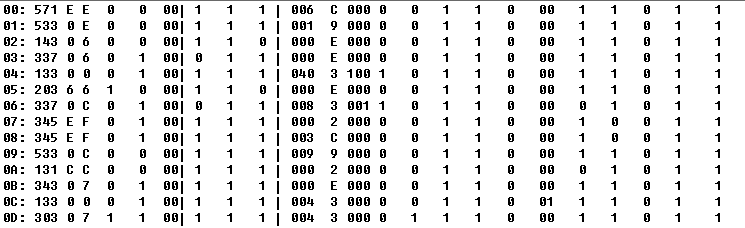


Рис. 9 – Микропрограмма командного цикла (выборка команды и установка признаков)

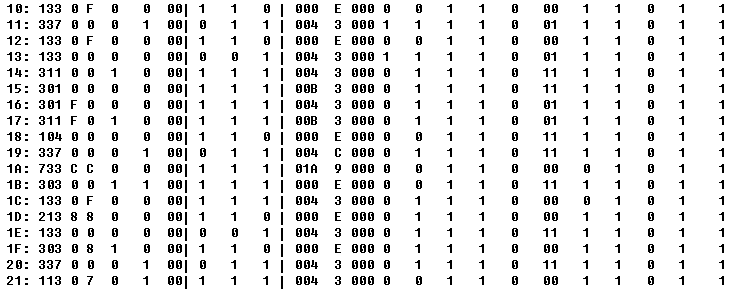


Рис. 10 – Микропрограмма командного цикла (выполнение операций)