МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

«Вятский государственный университет» Факультет автоматики и вычислительной техники Кафедра электронных вычислительных машин

Отчет по лабораторной работе №6 по дисциплине

«Организация ЭВМ и систем» Вариант 7

Выполнил студент группы ИВТ-32 /Кудяшев Я.Ю./ Проверил преподаватель /Клюкин В.Л./

Киров 2021

1. Задание

Определить архитектуру ЭВМ с системой прерываний; разработать структурную схему и алгоритм работы ЭВМ; составить и отладить микропрограмму командного цикла ЭВМ.

Разработать микропрограмму, вычисляющую количество единиц в числе.

1. Определение архитектуры и программирование
   1. Формат данных

X может принадлежать в пределах 0...65535, поэтому любое число можно представить 16-разрядным двоичным кодом без знака.

* 1. Программно-доступные регистры

ЭВМ имеет девять программно-доступных регистров: шесть регистров общего назначения (r0-r5), программный счетчик – IP (r6), регистр признаков

* FLAGS (r7), содержащий разряд признака нуля (Z), регистр указателя стека
* SP (r8), регистр адреса таблицы прерываний – ITR (r9), а также 8-разрядный регистр маски RM.
  1. Система команд

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Мнемоника | Описание | Изменение признака Z |
| Суммирование | ADD r r\* | r := r + r\*, IP := IP + 1 | + |
| Вычитание | SUB r r\* | r := r – r\*, IP := IP + 1 | + |
| Добавление C | AD r C | r := r + C, IP := IP + 1 | + |
| Вычитание C | SB r C | r := r – C, IP := IP + 1 | + |
| Чтение в регистр | LD r A | r := M[A], IP := IP + 1 | - |
| Запись регистра | MV r A | M[A] := r, IP := IP + 1 | - |
| Чтение в регистр с индексацией | LDI r, r\* | r := M[r\*], IP := IP + 1 | - |
| Запись в стек | PUSH r (SP) | M[SP] := r, SP := SP – 1, IP := IP + 1 | - |
| Чтение из стека | POP r (SP) | SP := SP + 1, r := M[SP], IP := IP + 1 | - |
| Переход | JMP A | IP := A | - |
| Переход, если нуль | JZ A | Если Z = 1, то IP := A, иначе IP := IP + 1 | - |
| Обращение к подпрограмме | CALL (SP) A | M[SP] := IP, SP := SP – 1, IP := IP + 1 | - |

|  |  |  |  |
| --- | --- | --- | --- |
| Загрузка маски | LM A | RM := M[A], IP := IP + 1 | - |
| Сдвиг вправо логический | SHR r r\* | r := r\* / 2, IP := IP + 1 | + |
| Останов | HLT A | IP := A, останов | - |
| Возврат из прерывающей программы | IRET | SP := SP + 1, RM := M[SP], SP := SP + 1, IP := M[SP] | - |

В описании системы команд приняты следующие обозначения:

* + - r, r\*∈ {r0, r1, … r8} – программно-доступные регистры: регистр r\* является источником данных, а регистр r – приемником результата, но может также служить источником второго операнда
    - М[А] – ячейка памяти с адресом А
    - Знак "+" в описании признаков означает, что устанавливается новое значение признака по результату выполнения команды, а знак "-" свидетельствует о сохранении старого значения признака

1. Кодирование программы и распределение памяти программ и данных
   1. Коды операций

|  |  |  |
| --- | --- | --- |
| Название | Мнемоника | Код операции |
| Суммирование | ADD | 0x01 |
| Вычитание | SUB | 0x02 |
| Добавление C | AD | 0x9 |
| Вычитание C | SB | 0xA |
| Чтение в регистр | LD | 0xB |
| Запись регистра | MV | 0xC |
| Чтение в регистр с индексацией | LDI | 0x0E |
| Запись в стек | PUSH | 0x07 |
| Чтение из стека | POP | 0x06 |
| Переход | JMP | 0x03 |
| Переход, если нуль | JZ | 0x04 |
| Обращение к подпрограмме | CALL | 0xD |
| Возврат из подпрограммы | IRET | 0x05 |
| Сдвиг вправо логический | SHR | 0x08 |
| Останов | HLT | 0x00 |
| Загрузка маски | LM | 0xF |

* 1. Распределение памяти программ и данных

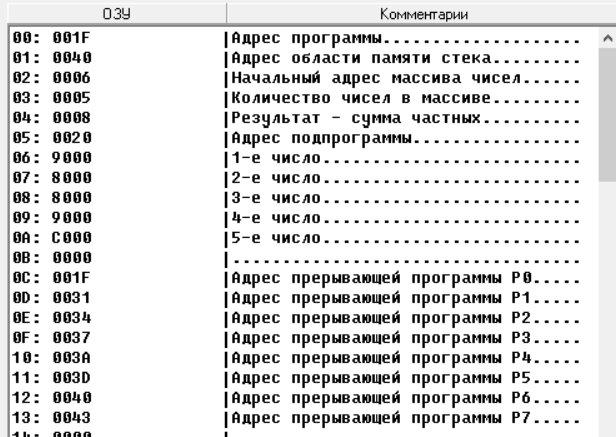


Рисунок 1 – Служебная программа и таблица прерываний

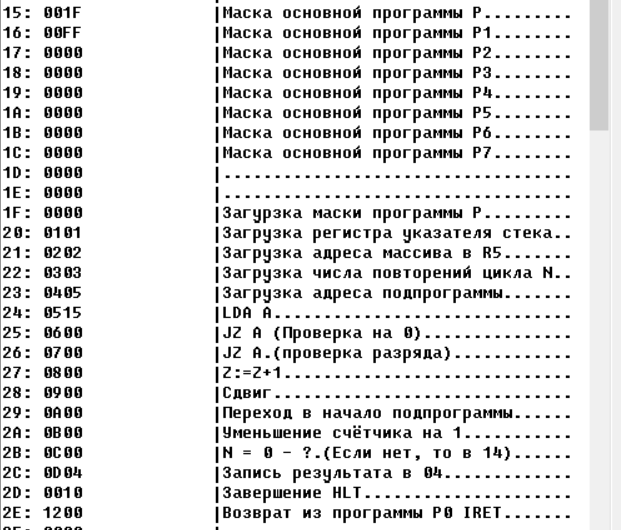


Рисунок 2 – Маски и основная программа

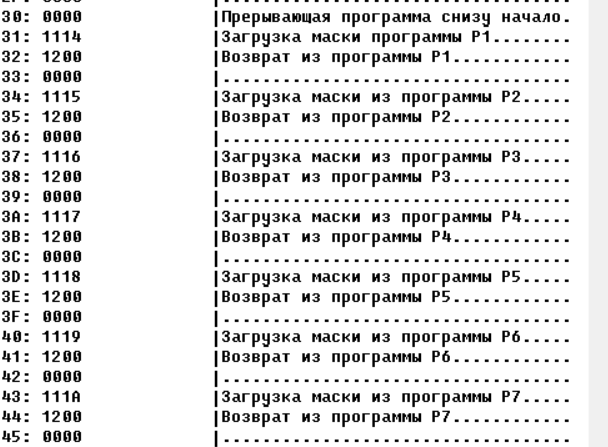


Рисунок 3 – Прерывающие программы P1…P7

* 1. Разработка структуры и алгоритма работы

Алгоритм работы ЭВМ представлен на рисунке 3 в виде укрупненной граф-схемы микропрограммы командного цикла.

Начало

Начало

INT

IN

RA := SP

SP := SP - 1

TST

1

1

RA := SP SP := SP - 1

M[SP] :=

RM

IP := M[RA]

RZ[RV] := 0

IP := SA

M[SP] := IP

INT

0

RF := ITA +

RV

RM := FF

1

INT

RA := RF

Конец

INT

Выбор

команды

F15

0

ДШ К1

INC INV PUSH POP CLR SHL

Преобразо

вание

ДШ К2, K3

MOV LOD CALL SET JMP JZ JF JNF HLT

INC POP CLR SHL

MOV

CALL

JMP

JF

INV

PUSH

LOD

SET

JZ

JNF

HLT

TST := 0

Конец

Рисунок 5 – Граф-схема микропрограммного командного цикла учебной

ЭВМ

* 1. Форматы команд

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Формат | | | | | Пример |
| 0 | КОП | | r\* | r | PUSH |
| 0 | КОП | | A | | JZ |
| 1 | КОП | r\* | A | | CALL |
| 1 | КОП | r\* | imm16 | | AD |

* 1. Распределение регистров

|  |  |  |  |
| --- | --- | --- | --- |
|  | РЗУ (R0 – R7) |  | РЗУ (R8 – R15) |
| 0 | r0 | 8 | r8 (SP) |
| 1 | r1 | 9 | r9 (ITA) |
| 2 | r2 | 10 |  |
| 3 | r3 | 11 |  |
| 4 | r4 | 12 |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | РЗУ (R0 – R7) | | |  | РЗУ (R8 – R15) |
| 5 | r5 | | | 13 | Буферный регистр команд |
| 6 | r6 (IP) | | | 14 | Регистр константы |
| 7 | r7 (FLAGS) | | Z | 15 | Счетчик адреса ЗУ RK[A] |
| RA | Адрес ЗУ |  | | RQ |  |

* 1. Коды операций

|  |  |  |
| --- | --- | --- |
| Название | Мнемоника | Код операции |
| Суммирование | ADD | 0x01 |
| Вычитание | SUB | 0x02 |
| Добавление C | AD | 0x9 |
| Вычитание C | SB | 0xA |
| Чтение в регистр | LD | 0xB |
| Запись регистра | MV | 0xC |
| Чтение в регистр с индексацией | LDI | 0x0E |
| Запись в стек | PUSH | 0x07 |
| Чтение из стека | POP | 0x06 |
| Переход | JMP | 0x03 |
| Переход, если нуль | JZ | 0x04 |
| Обращение к подпрограмме | CALL | 0xD |
| Возврат из подпрограммы | IRET | 0x05 |
| Сдвиг вправо логический | SHR | 0x08 |
| Останов | HLT | 0x00 |
| Загрузка маски | LM | 0xF |

* 1. Микропрограмма командного цикла

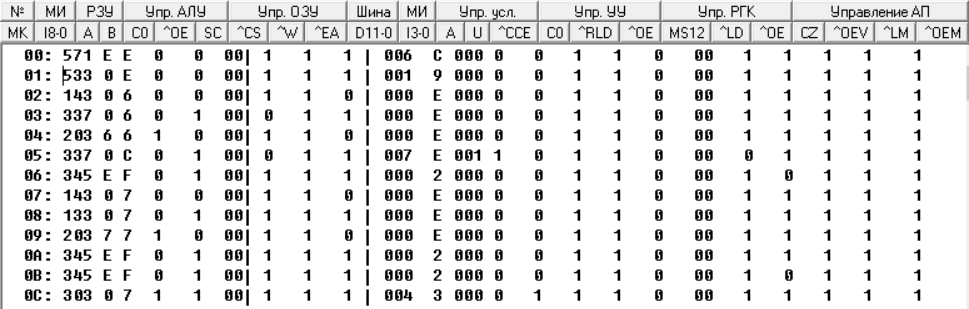


Рисунок 6 – Микропрограмма командного цикла

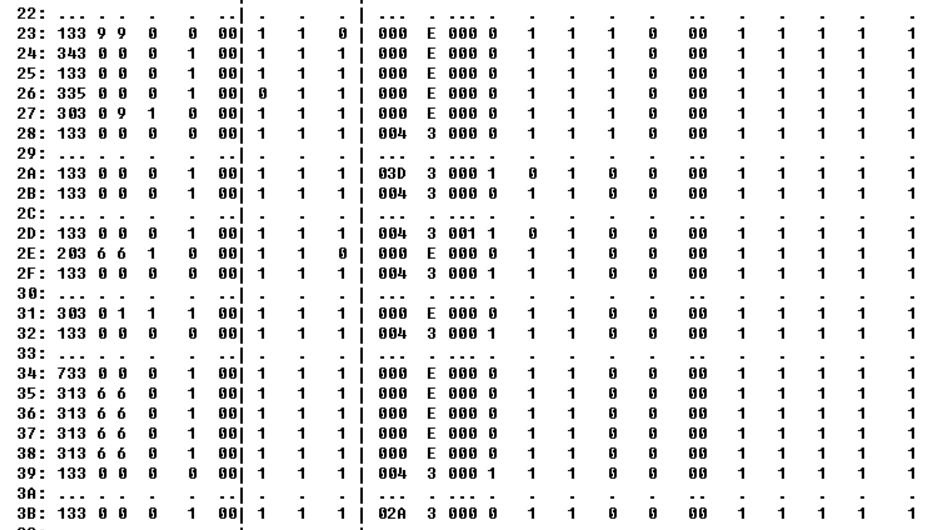


Рисунок 7 – Подмикропрограммы операций (IRET и INT)

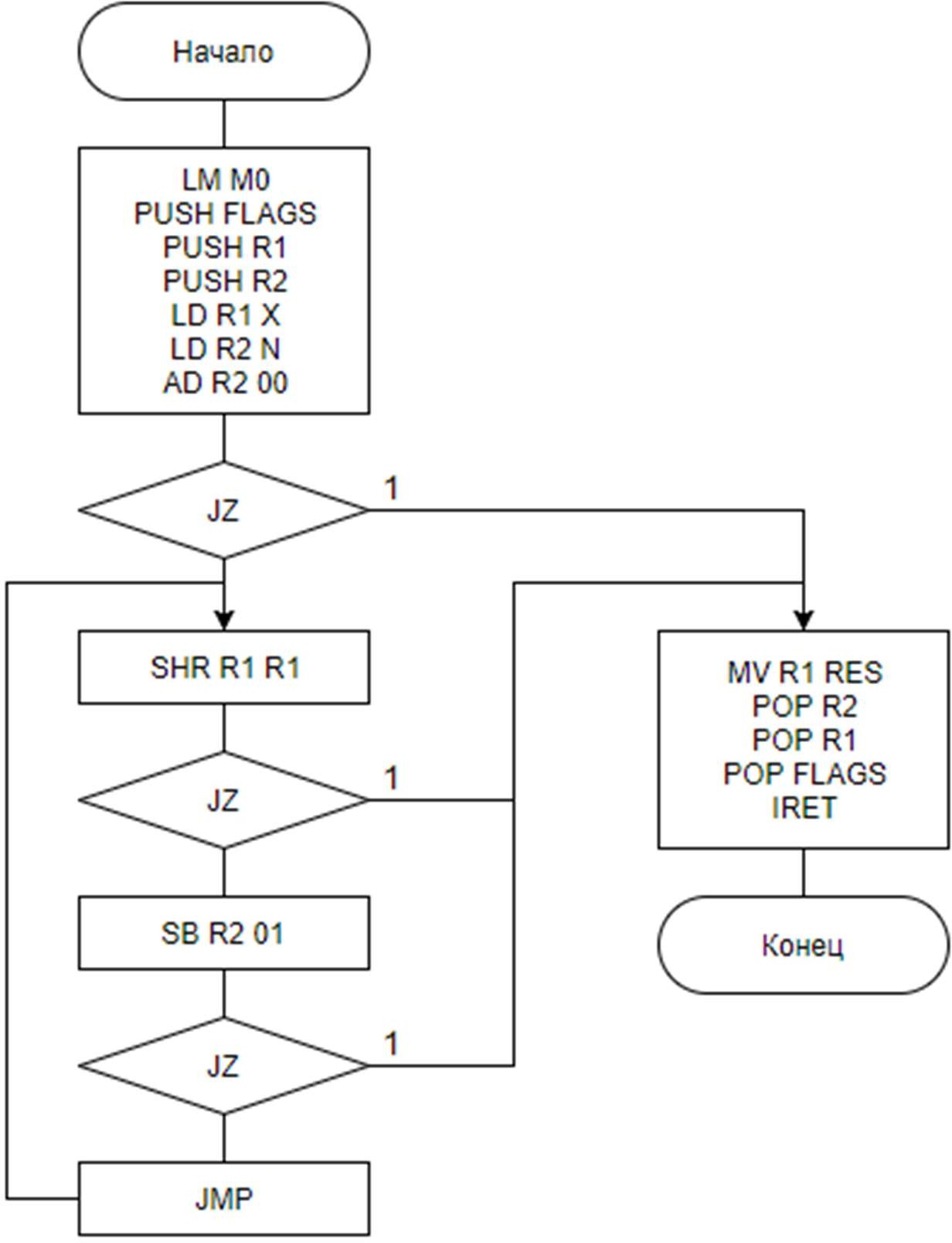


Рисунок 8 – Алгоритм решения задачи

1. Ввод и отладка микропрограммы командного цикла и программы решения задачи

Диаграмма переключения представлена на рисунке 9.

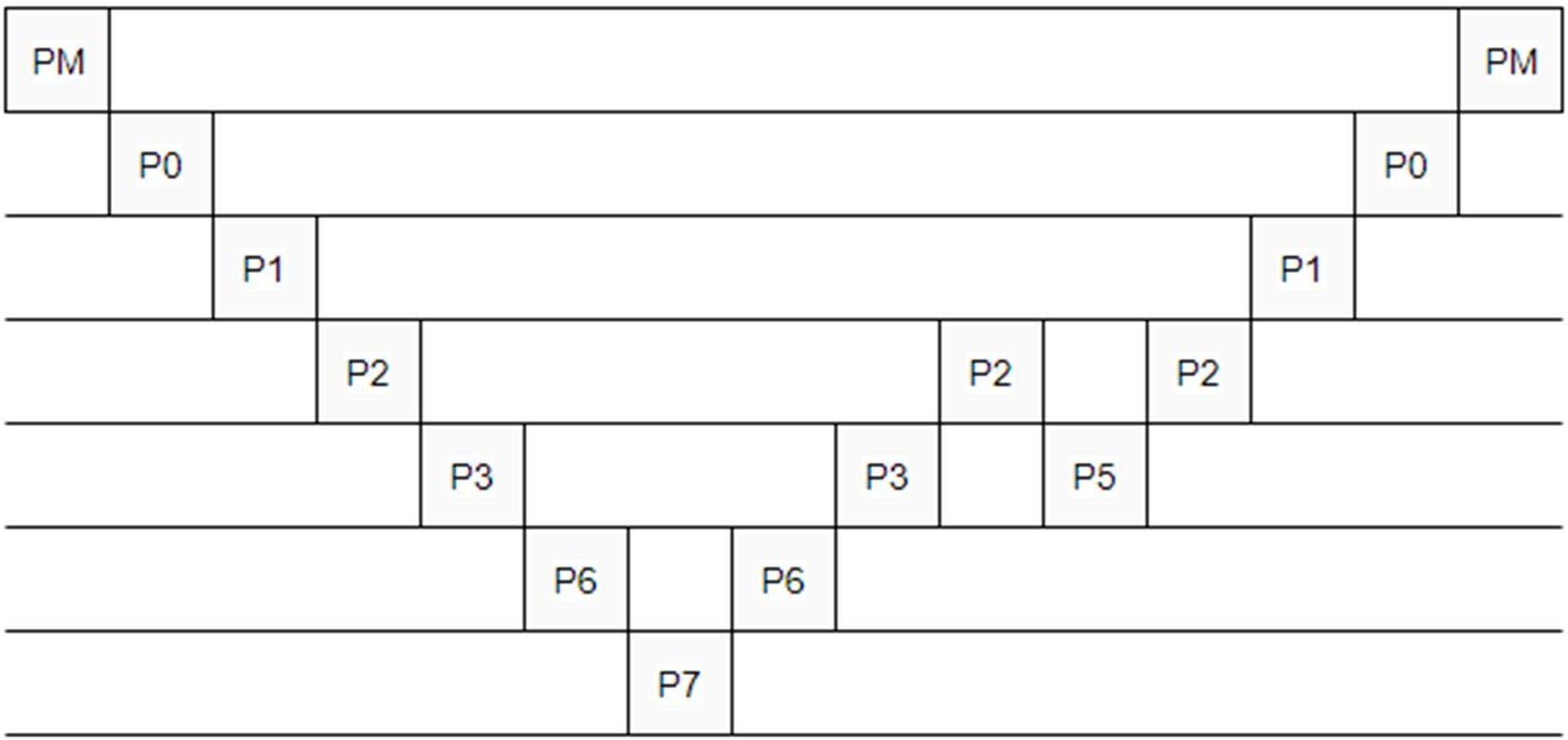


Рисунок 9 – Диаграмма переключений

1. Выводы

В ходе лабораторной работы была разработана и изучена учебная ЭВМ. В ходе выполнения лабораторной работы была изучена система и добавлена система прерываний. Добавленная система прерываний может получать 8 запросов на прерывание. Запросы имеют приоритет, выражаемый позицией запроса в регистре запросов от 7 до 0, чем ниже значение, тем выше приоритет. Для запрета или разрешения обработки запроса существует маска. Так, если некоторый бит маски равен 1, то соответствующий запрос (располагающийся в соответствующем бите регистра запросов) обработан не будет. Таким образом были запрещены любые прерывания в служебной программе. После обработки бит обработанного запроса устанавливается в 0 для устранения возможности повторной обработки запроса.