Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерных технологий

Основы профессиональной деятельности

Лабораторная работа №5

Вариант 6791

Выполнила:

Павличенко Софья Алексеевна, Р3115

Проверила:

Ершова Анна Ильинична

Санкт-Петербург 2024г.

Оглавление

[Цель 3](#_Toc163771399)

[Задание 4](#_Toc163771400)

[Текст исходной программы 5](#_Toc163771401)

[Описание программы 6](#_Toc163771402)

[Область определения 6](#_Toc163771403)

[ОДЗ 6](#_Toc163771404)

[Таблица трассировки 7](#_Toc163771405)

[Вывод 9](#_Toc163771406)

# Цель

Изучение организации системы ввода-вывода базовой ЭВМ, команд ввода-вывода и исследование процесса функционирования ЭМВ при обмене данными по сигналам готовности внешних устройств (ВУ).

# Задание

По выданному преподавателем варианту разработать программу асинхронного обмена данными с внешним устройством. При помощи программы осуществить ввод или вывод информации, используя в качестве подтверждения данных сигнал (кнопку) готовности ВУ.

1. Программа осуществляет асинхронный вывод данных на ВУ-1
2. Программа начинается с адреса 1BE16. Размещаемая строка находится по адресу 56116.
3. Строка должна быть представлена в кодировке Windows-1251.
4. Формат представления строки в памяти: АДР1: СИМВ1 СИМВ2 АДР2: СИМВ3 СИМВ4 ... СТОП\_СИМВ.
5. Ввод или вывод строки должен быть завершен по символу c кодом 0D (CR). Стоп символ является обычным символом строки и подчиняется тем же правилам расположения в памяти что и другие символы строки.

# Текст исходной программы

|  |  |
| --- | --- |
| ORG 0x1BE | ; Программа начинается с адреса 1BE |
| STRING: WORD 0x561 | ; Адрес начала строки |
| POINTER: WORD ? | ; Указатель текущих двух символов строки |
| STOP: WORD 0x0D | ; Стоп-символ |
|  |  |
| START: CLA | ; 0 -> AC |
| LD STRING | ; Установка указателя на первых двух символах строки |
| ST POINTER |
|  |  |
| S1: IN 0x03 | ; SR#3 -> AC |
| AND #0x40 | ; “Готов” нажата? |
| BEQ S1 | ; Нет – “Спин-луп” |
| LD (POINTER) | ; Загрузка текущих двух символов строки в аккумулятор |
| SWAB | ; Перемещение первого символа в младший байт AC |
| OUT 0x02 | ; Вывод на ВУ1 первый символ |
|  |  |
| SXTB | ; Проверка на стоп-символ |
| CMP STOP |
| BEQ FINISH |
|  |
| S2: IN 0x03 | ; SR#3 -> AC |
| AND #0x40 | ; “Готов” нажата? |
| BEQ S2 | ; Нет – “Спин-луп” |
| LD (POINTER)+ | ; Загрузка текущих символов строки в аккумулятор (+ перемещение указателя) |
|  |  |
| OUT 0x02 | ; Вывод на ВУ1 второй символ |
|  | ; Проверка на стоп-символ |
| SXTB |
| CMP STOP |
| BEQ FINISH |
|  |
| JUMP S1 | ; Повтор |
| FINISH: HLT | ; Конец программы |
|  |  |
| ORG 0x561 | ; Строка размещается по адресу 561 |
| WORD 0xD1EF | ; Сп |
| WORD 0xE0F1 | ; ас |
| WORD 0xC8F2 | ; ит |
| WORD 0xE50D | ; eCR |

# Описание программы

Программа осуществляет асинхронный вывод данных на ВУ1.

## Область определения

STRING (адрес начала строки) – 11-разрядное беззнаковое число

POINTER (указатель текущих двух символов строки) – 11-разрядное беззнаковое число

STOP (стоп-символ) – 8-разрядное беззнаковое число = 0D

S (символ строки) – 8-разрядное беззнаковое число

N – длина строки

## ОДЗ

­­­­



# Таблица трассировки

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Выполняемая команда** | | **Содержимое регистра процессора после выполнения команды** | | | | | | | | **Ячейка, содержимое которой изменилось после выполнения команды** | |
| **Адрес** | **Код команды** | **IP** | **CR** | **AR** | **DR** | **SP** | **BR** | **AC** | **NZVC** | **Адрес** | **Новый код** |
| 81 | 200 | 82 | 200 | 81 | 200 | 0 | 81 | 0 | 100 |  |  |
| 82 | EE1A | 83 | EE1A | 09D | 0 | 0 | 001A | 0 | 100 | 09D | 0 |
| 83 | AE17 | 84 | AE17 | 09B | 04B0 | 0 | 17 | 04B0 | 0 |  |  |
| 84 | 0C00 | 85 | 0C00 | 7FF | 04B0 | 7FF | 84 | 04B0 | 0 | 7FF | 04B0 |
| 85 | D687 | 687 | D687 | 7FE | 86 | 7FE | D687 | 04B0 | 0 | 7FE | 86 |
| 687 | AC01 | 688 | AC01 | 7FF | 04B0 | 7FE | 1 | 04B0 | 0 |  |  |
| 688 | F207 | 689 | F207 | 688 | F207 | 7FE | 688 | 04B0 | 0 |  |  |
| 689 | 7E09 | 68A | 7E09 | 693 | 07AE | 7FE | 9 | 04B0 | 1000 |  |  |
| 68A | F905 | 68B | F905 | 68A | F905 | 7FE | 068A | 04B0 | 1000 |  |  |
| 68B | 4C01 | 68C | 4C01 | 7FF | 04B0 | 7FE | 1 | 960 | 0 |  |  |
| 68C | 4C01 | 68D | 4C01 | 7FF | 04B0 | 7FE | 1 | 0E10 | 0 |  |  |
| 68D | 4C01 | 68E | 4C01 | 7FF | 04B0 | 7FE | 1 | 12C0 | 0 |  |  |
| 68E | 6E05 | 68F | 6E05 | 694 | 12 | 7FE | 5 | 12AE | 1 |  |  |
| 68F | CE01 | 691 | CE01 | 68F | 691 | 7FE | 1 | 12AE | 1 |  |  |
| 691 | EC01 | 692 | EC01 | 7FF | 12AE | 7FE | 1 | 12AE | 1 | 7FF | 12AE |
| 692 | 0A00 | 86 | 0A00 | 7FE | 86 | 7FF | 692 | 12AE | 1 |  |  |
| 86 | 800 | 87 | 800 | 7FF | 12AE | 0 | 86 | 12AE | 1 |  |  |
| 87 | 6E15 | 88 | 6E15 | 09D | 0 | 0 | 15 | 12AE | 1 |  |  |
| 88 | EE14 | 89 | EE14 | 09D | 12AE | 0 | 14 | 12AE | 1 | 09D | 12AE |
| 89 | AE12 | 08A | AE12 | 09C | 05DC | 0 | 12 | 05DC | 1 |  |  |
| 08A | 700 | 08B | 700 | 08A | 700 | 0 | 008A | 05DD | 0 |  |  |
| 08B | 0C00 | 08C | 0C00 | 7FF | 05DD | 7FF | 008B | 05DD | 0 | 7FF | 05DD |
| 08C | D687 | 687 | D687 | 7FE | 008D | 7FE | D687 | 05DD | 0 | 7FE | 008D |
| 687 | AC01 | 688 | AC01 | 7FF | 05DD | 7FE | 1 | 05DD | 0 |  |  |
| 688 | F207 | 689 | F207 | 688 | F207 | 7FE | 688 | 05DD | 0 |  |  |
| 689 | 7E09 | 68A | 7E09 | 693 | 07AE | 7FE | 9 | 05DD | 1000 |  |  |
| 68A | F905 | 68B | F905 | 68A | F905 | 7FE | 068A | 05DD | 1000 |  |  |
| 68B | 4C01 | 68C | 4C01 | 7FF | 05DD | 7FE | 1 | 0BBA | 0 |  |  |
| 68C | 4C01 | 68D | 4C01 | 7FF | 05DD | 7FE | 1 | 1197 | 0 |  |  |
| 68D | 4C01 | 68E | 4C01 | 7FF | 05DD | 7FE | 1 | 1774 | 0 |  |  |
| 68E | 6E05 | 68F | 6E05 | 694 | 12 | 7FE | 5 | 1762 | 1 |  |  |
| 68F | CE01 | 691 | CE01 | 68F | 691 | 7FE | 1 | 1762 | 1 |  |  |
| 691 | EC01 | 692 | EC01 | 7FF | 1762 | 7FE | 1 | 1762 | 1 | 7FF | 1762 |
| 692 | 0A00 | 08D | 0A00 | 7FE | 008D | 7FF | 692 | 1762 | 1 |  |  |
| 08D | 800 | 08E | 800 | 7FF | 1762 | 0 | 008D | 1762 | 1 |  |  |
| 08E | 740 | 08F | 740 | 08E | 740 | 0 | 008E | 1761 | 1 |  |  |
| 08F | 6E0D | 90 | 6E0D | 09D | 12AE | 0 | 000D | 04B3 | 1 |  |  |
| 90 | EE0C | 91 | EE0C | 09D | 04B3 | 0 | 000C | 04B3 | 1 | 09D | 04B3 |
| 91 | AE08 | 92 | AE08 | 09A | 07AE | 0 | 8 | 07AE | 1 |  |  |
| 92 | 700 | 93 | 700 | 92 | 700 | 0 | 92 | 07AF | 0 |  |  |
| 93 | 0C00 | 94 | 0C00 | 7FF | 07AF | 7FF | 93 | 07AF | 0 | 7FF | 07AF |
| 94 | D687 | 687 | D687 | 7FE | 95 | 7FE | D687 | 07AF | 0 | 7FE | 95 |
| 687 | AC01 | 688 | AC01 | 7FF | 07AF | 7FE | 1 | 07AF | 0 |  |  |
| 688 | F207 | 689 | F207 | 688 | F207 | 7FE | 688 | 07AF | 0 |  |  |
| 689 | 7E09 | 68A | 7E09 | 693 | 07AE | 7FE | 9 | 07AF | 1 |  |  |
| 68A | F905 | 690 | F905 | 68A | F905 | 7FE | 5 | 07AF | 1 |  |  |
| 690 | AE02 | 691 | AE02 | 693 | 07AE | 7FE | 2 | 07AE | 1 |  |  |
| 691 | EC01 | 692 | EC01 | 7FF | 07AE | 7FE | 1 | 07AE | 1 | 7FF | 07AE |
| 692 | 0A00 | 95 | 0A00 | 7FE | 95 | 7FF | 692 | 07AE | 1 |  |  |
| 95 | 800 | 96 | 800 | 7FF | 07AE | 0 | 95 | 07AE | 1 |  |  |
| 96 | 700 | 97 | 700 | 96 | 700 | 0 | 96 | 07AF | 0 |  |  |
| 97 | 6E05 | 98 | 6E05 | 09D | 04B3 | 0 | 5 | 02FC | 1 |  |  |
| 98 | EE04 | 99 | EE04 | 09D | 02FC | 0 | 4 | 02FC | 1 | 09D | 02FC |
| 99 | 100 | 09A | 100 | 99 | 100 | 0 | 99 | 02FC | 1 |  |  |
| 09A | 07AE |  |  |  |  |  |  |  |  |  |  |
| 09B | 04B0 |  |  |  |  |  |  |  |  |  |  |
| 09C | 05DC |  |  |  |  |  |  |  |  |  |  |
| 09D | FFF0 |  |  |  |  |  |  |  |  |  |  |

# Вывод

В ходе выполнения данной лабораторной работы я познакомилась с организацией подпрограмм в БЭВМ, изучила работу стека и команды CALL и RET.

ORG 0x00

IN 0X1C

ST 0x81

IN 0x1C

ST 0x82

ASL

ASL

ASL

ASL

IN 0x1C

ST 0x83

LD 0x81

SWAB

OR 0x82

OR 0x83

ST 0x90

CMP #0x01B1

BEQ SONG

0x81 = 0000

0x90 = 0000

AC = 0000

ORG 0x00

CLA

START: IN 0x1C

ST 0x81

LD 0x90

ASL

ASL

ASL

ASL

OR 0x81

ST 0x90

CMP 0X92

BEQ SONG

SONG:LD 0X91

|  |
| --- |
| OUT 0x02  JUMP START |

ORG 0x91

WORD 0xD0F3

WORD 0x1B1