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

Факультет Программной инженерии и компьютерной техники

Отчёт по лабораторной работе №5

Основы профессиональной деятельности

Вариант № 3485

Выполнил: студент группы P3131

Серов А. А.

Проверил: Остапенко О.Д.

Санкт-Петербург 2025г.

1. **Изображение выглядит как текст, Шрифт, снимок экрана, информация

   Контент, сгенерированный ИИ, может содержать ошибки.Текст задания**

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 ввод данных с ВУ-3
2. Программа начинается с адреса 5DA16. Размещаемая строка находится по адресу 58716.
3. Строка должна быть представлена в кодировке Windows-1251.
4. Формат представления строки в памяти: АДР0: ДЛИНА АДР1: СИМВ1 СИМВ2 АДР2: СИМВ3 СИМВ4 ..., где ДЛИНА - 16 разрядное слово, где значащими являются 8 младших бит.
5. Ввод строки начинается со ввода количества символов (1 байт), и должен быть завершен по вводу их необходимого количества.
6. **Описание программы**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.**

1. **Текст исходной программы**

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Комментарии** |
| 0500 | - | RES\_ADDRESS | Адрес текущей ячейки |
| 0501 | - | EOF | Символ конца ввода |
| 0502 | - | TMP | переменная для хранения временных данных |
| 5DA | 0200 | CLA | Очистить аккумулятор |
| 5DC | 1207 | IN #7 | Ожидание ввода |
| 5DD | 2F40 | AND #0x40 | Проверка статуса |
| 5DE | F0FD | BEQ IP-3 | Спин-луп в ожидании готовности |
| 5DF | 1206 | IN #6 | Считывание символа |
| 5E0 | 0680 | SWAB | Меняем старший и младший байты |
| 5E1 | E8A1 | ST (RES\_ADDRESS) | Сохраняем в текущей ячейке |
| 5E2 | 0680 | SWAB | Меняем обратно для проверки EOF |
| 5E3 | 7EA0 | CMP EOF | Проверяем на наличие EOF |
| 5E4 | F00B | BEQ IP+11 | Если EOF, то завершаем |
| 5E5 | 1207 | IN #7 | Ожидание ввода |
| 5E6 | 2F40 | AND #0x40 | Проверка статуса |
| 5E7 | F0FD | BEQ IP-3 | Спин-луп в ожидании готовности |
| 5E8 | 1206 | IN #6 | Считывание символа |
| 5E9 | EE9B | ST TMP | Сохраним во временной переменной |
| 5EA | 4898 | ADD (RES\_ADDRESS) | Добавим значение предыдущего символа |
| 5EB | EA97 | ST (RES\_ADDRESS)+ | Сохраним 2 символа в текущей ячейке |
| 5EC | AE98 | LD TMP | Загружаем в аккумулятор временную перменную |
| 5ED | 7E96 | CMP EOF | Проверяем на наличие EOF |
| 5EE | F001 | BEQ IP+1 | Если EOF, то завершаем |
| 5EF | CEEC | JUMP IP-20 | Прыжок на ячейку 069 |
| 5F0 | 0100 | HLT | Завершение программы |
| 587 | - | RES | Строка результата |

1. **Описание программы**

Программа осуществляет посимвольный асинхронный ввод данных с ВУ-3. Программа будет получать символы до тех пор, пока на ВУ-3 не будет введен стоп-символ (EOF) с кодировкой 0x0D, который она запишет в память и прекратит свое выполнение.

**Область представления:**

RES -? - 16-разрядные ячейки, хранящие в себе по два символа в кодировке Windows-1251

RES\_ADDRESS - 11 разрядная ячейка, хранящая адрес текущей ячейкт

EOF - 16-разрядная константа.

TMP - 16-разрядная ячейка для временного хранения введенных символов.

**Область определения:**

RES\_ADDRESS (указатель на ячейки массива, хранящий результат ввода) ∈ [5B9;2047]

EOF: [00; FF]

1. **Таблица трассировки**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Адр** | **Знчн** | **IP** | **CR** | **AR** | **DR** | **SP** | **BR** | | **AC** | **PS** | **NZVC** | **Адр** | **Знчн** |
| 5DA | 0200 | 5DA | 0000 | 000 | 0000 | 000 | 0000 | 0000 | | 004 | 0100 |  |  |
| 5DA | 0200 | 5DB | 0200 | 5DA | 0200 | 000 | 05DA | 0000 | | 004 | 0100 |  |  |
| 5DB | F000 | 5DC | F000 | 5DB | F000 | 000 | 0000 | 0000 | | 004 | 0100 |  |  |
| 5DC | 1206 | 5DD | 1206 | 5DC | 1206 | 000 | 05DC | 00C9 | | 004 | 0100 |  |  |
| 5DD | E800 | 5DE | E800 | 680 | 00C9 | 000 | 0000 | 00C9 | | 004 | 0100 | 680 | 00C9 |
| 5DE | 0680 | 5DF | 0680 | 5DE | 0680 | 000 | 05DE | C900 | | 008 | 1000 |  |  |
| 5DF | 7E00 | 5E0 | 7E00 | 5E0 | F009 | 000 | 0000 | C900 | | 008 | 1000 |  |  |
| 5E0 | F009 | 5E1 | F009 | 5E0 | F009 | 000 | 05E0 | C900 | | 008 | 1000 |  |  |
| 5E1 | F000 | 5E2 | F000 | 5E1 | F000 | 000 | 05E1 | C900 | | 008 | 1000 |  |  |
| 5E2 | 1206 | 5E3 | 1206 | 5E2 | 1206 | 000 | 05E2 | C9C9 | | 008 | 1000 |  |  |
| 5E3 | EE00 | 5E4 | EE00 | 5E4 | C9C9 | 000 | 0000 | C9C9 | | 008 | 1000 | 5E4 | C9C9 |
| 5E4 | C9C9 | 5E5 | C9C9 | 5AE | 0000 | 000 | FFC9 | C9C9 | | 008 | 1000 |  |  |
| 5E5 | EA00 | 5E6 | EA00 | 600 | C9C9 | 000 | 0000 | C9C9 | | 008 | 1000 | 5E6, 600 | AE01, C9C9 |
| 5E6 | AE01 | 5E7 | AE01 | 5E8 | F001 | 000 | 0001 | F001 | | 008 | 1000 |  |  |
| 5E7 | 7E00 | 5E8 | 7E00 | 5E8 | F001 | 000 | 0000 | F001 | | 005 | 0101 |  |  |
| 5E8 | F001 | 5EA | F001 | 5E8 | F001 | 000 | 0001 | F001 | | 005 | 0101 |  |  |
| 5EA | 0100 | 5EB | 0100 | 5EA | 0100 | 000 | 05EA | F001 | | 005 | 0101 | 5EA | 0100 |

1. **Вывод**

В ходе проделанной лабораторной работы, я познакомился с асинхронным вводом-выводом данных в БЭВМ, узнал о внешних устройствах и их регистрах. Также познакомился с представлением данных в различных кодировках и попрактиковался в написания кода на Ассемблере БЭВМ.