Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

**Отчет по лабораторной работе №5**

**«Асинхронный обмен данными с ВУ»**

**по дисциплине**

**«Основы профессиональной деятельности»**

**Вариант 1571**

Выполнил: студент гр. P3115

Бусыгин Д. А.

Проверила:

Блохина Е. Н.

Санкт-Петербург 2022

Оглавление

[Задание лабораторной работы 2](#_Toc98234913)

[Текст основной программы 3](#_Toc98234914)

[Текст подпрограммы 4](#_Toc98234915)

[Описание программы 4](#_Toc98234916)

[Таблица трассировки программы с выданными числами 6](#_Toc98234917)

[Вывод 7](#_Toc98234918)

Задание лабораторной работы

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 17F16. Размещаемая строка находится по адресу 61016.
3. Строка должна быть представлена в кодировке Windows-1251.
4. Формат представления строки в памяти: АДР0: ДЛИНА АДР1: СИМВ2 СИМВ1 АДР2: СИМВ4 СИМВ3 ..., 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.

Заданное слово и коды его символов

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Буква | М | а | г | а | з | и | н | = |
| Win-1251 | CC | E0 | E3 | E0 | C7 | E8 | ED | 3D |
| UTF-8 | D09C | D0B0 | D0B3 | D0B0 | D0B7 | D0B8 | D0BD | 003D |
| UTF-16 | 041C | 0430 | 0433 | 0430 | 0437 | 0438 | 043D | 003D |

Кол-во символов: 8

Текст исходной программы (БЭВМ)

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Описание команды** |
| **BEGIN** | | | |
| **17F** | **0200** | **CLA** | **Очистка аккумулятора** |
| **LEN\_INPUT** | | | |
| **180** | **1207** | **IN 7** | **Ожидание готовности ВУ** |
| **181** | **2F40** | **AND #0x40** |
| **182** | **F0FD** | **BEQ LEN\_INPUT** |
| **183** | **1206** | **IN 6** | **Получение данных с ВУ** |
| **184** | **E819** | **ST (LEN)** | **Сохранение длины вводимой строки** |
| **185** | **EA17** | **ST (CUR)+** | **Сохранение длины вводимой строки в массив** |
| **S1** | | | |
| **186** | **0200** | **CLA** | **Очистка аккумулятора** |
| **187** | **1207** | **IN 7** | **Ожидание готовности ВУ** |
| **188** | **2F40** | **AND #0x40** |
| **189** | **F0FC** | **BEQ S1** |
| **18A** | **1206** | **IN 6** | **Получение данных с ВУ** |
| **18B** | **E811** | **ST (CUR)** | **Сохранение первого символа в ячейку** |
| **18C** | **DE0B** | **CALL CHECK** | **Проверка количества введенных символов** |
| **S2** | | | |
| **18D** | **0200** | **CLA** | **Очистка аккумулятора** |
| **18E** | **1207** | **IN 7** | **Ожидание готовности ВУ** |
| **18F** | **2F40** | **AND #0x40** |
| **190** | **F0FC** | **BEQ S2** |
| **191** | **1206** | **IN 6** | **Получение данных с ВУ** |
| **192** | **0680** | **SWAB** | **Обмен старшего и младшего байтов** |
| **193** | **4809** | **ADD (CUR)** | **Объединение символа с предыдущим** |
| **194** | **EA08** | **ST (CUR)+** | **Сохранение пары символов** |
| **195** | **DE02** | **CALL CHECK** | **Проверка количества введенных символов** |
| **196** | **CEEF** | **JUMP S1** | **Возврат к считыванию символов** |
| **FINISH** | | | |
| **197** | **0100** | **HLT** | **Останов** |
| **CHECK** | | | |
| **198** | **A805** | **LD (LEN)** | **Загрузка значения длины строки в аккумулятор** |
| **199** | **0740** | **DEC** | **Декремент (ввели новый символ – осталось ввести на 1 меньше)** |
| **19A** | **F0FC** | **BEQ FINISH** | **Если 0, то заканчиваем программу** |
| **19B** | **E802** | **ST (LEN)** | **Иначе сохраняем длину строки обратно** |
| **19C** | **0A00** | **RET** | **Возврат из подпрограммы** |
| **ДОПОЛНИТЕЛЬНЫЕ ДАННЫЕ** | | | |
| **19D** | **0610** | **WORD 0x0610** | **Адрес ячейки, в которую нужно записать элемент массива** |
| **19E** | **0000** | **WORD 0x0000** | **Длина строки** |

# Текст исходной программы (Ассемблер)

|  |  |  |
| --- | --- | --- |
| **Адрес** | **Команда** | **Адресация** |
| **BEGIN:** | **CLA** |  |
| **LEN\_INPUT:** | **IN** | **7** |
| **AND** | **#0x40** |
| **BEQ** | **LEN\_INPUT** |
| **IN** | **6** |
| **ST** | **(LEN)** |
| **ST** | **(CUR)+** |
| **S1:** | **CLA** |  |
| **IN** | **7** |
| **AND** | **#0x40** |
| **BEQ** | **S1** |
| **IN** | **6** |
| **ST** | **(CUR)** |
| **CALL** | **CHECK** |
| **S2:** | **CLA** |  |
| **IN** | **7** |
| **AND** | **#0x40** |
| **BEQ** | **S2** |
| **IN** | **6** |
| **SWAB** |  |
| **ADD** | **(CUR)** |
| **ST** | **(CUR)+** |
| **CALL** | **CHECK** |
| **JUMP** | **S1** |
| **FINISH:** | **HLT** |  |
| **CHECK:** | **LD** | **(LEN)** |
| **DEC** |  |
| **BEQ** | **FINISH** |
| **ST** | **(LEN)** |
| **RET** |  |
| **CUR:** | **WORD** | **0x0610** |
| **LEN:** | **WORD** | **0x0000** |

# Описание программы

*Назначение программы*: считывание закодированных символов с ВУ-3 и сохранение их в память БЭВМ.

*Описание и назначение исходных данных и переменных*:

LEN – количество символов, которое введет пользователь (вводится перед вводом символов)

CUR – адрес ячейки, куда будет производиться запись (изначально 0610)

S1, S2 – вводимые пользователем символы в кодировке Windows-1251

*Область представления и область допустимых значений*:

S1, S2 – 8-разрядные двоичные числа, кодирующие символы [0, 255]

LEN – 8-разрядное двоичное число, характеризующее количество вводимых символов [1, 255]

CUR – 3-разрядное 16-ричное число, соответствующее адресу ячейки, с которой начнется запись массива [19F, 7FF][000, 17E]

*Расположение в памяти ЭВМ программы, подпрограммы, исходных данных и результата*:

Основная программа 17F – 196

Подпрограмма 197 – 19С

Исходные данные 19D, 19E

Результат 610 - ?

Максимальное кол-во вводимых символов: 7FF-610=1EF=495

# Таблица трассировки программы при слове длиной 2 символа

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Адрес** | **Значение** | **IP** | **CR** | **AR** | **DR** | **SP** | **BR** | **AC** | **NZVC** | **Адрес** | **Значение** |
| 180 | 1207 | 181 | 1207 | 180 | 1207 | 000 | 0180 | 0040 | 0100 |  |  |
| 181 | 2F40 | 182 | 2F40 | 181 | 0040 | 000 | 0040 | 0040 | 0000 |  |  |
| 182 | F0FD | 183 | F0FD | 182 | F0FD | 000 | 0182 | 0040 | 0000 |  |  |
| 183 | 1206 | 184 | 1206 | 183 | 1206 | 000 | 0183 | 0002 | 0000 |  |  |
| 184 | E818 | 185 | E818 | 000 | 0002 | 000 | 0018 | 0002 | 0000 |  |  |
| 185 | 0200 | 186 | 0200 | 185 | 0200 | 000 | 0185 | 0000 | 0100 |  |  |
| 186 | 1207 | 187 | 1207 | 186 | 1207 | 000 | 0186 | 0040 | 0100 |  |  |
| 187 | 2F40 | 188 | 2F40 | 187 | 0040 | 000 | 0040 | 0040 | 0000 |  |  |
| 188 | F0FC | 189 | F0FC | 188 | F0FC | 000 | 0188 | 0040 | 0000 |  |  |
| 189 | 1206 | 18A | 1206 | 189 | 1206 | 000 | 0189 | 0006 | 0000 |  |  |
| 18A | E811 | 18B | E811 | 610 | 0006 | 000 | 0011 | 0006 | 0000 | 610 | 00СС |
| 18B | 0680 | 18C | 0680 | 18B | 0680 | 000 | 018B | 0600 | 0000 |  |  |
| 18C | DE0A | 197 | DE0A | 7FF | 018D | 7FF | 0197 | 0600 | 0000 | 7FF | 018D |
| 197 | A805 | 198 | A805 | 000 | 0002 | 7FF | 0005 | 0002 | 0000 |  |  |
| 198 | 0740 | 199 | 0740 | 198 | 0740 | 7FF | 0198 | 0001 | 0001 |  |  |
| 199 | F0FC | 19A | F0FC | 199 | F0FC | 7FF | 0199 | 0001 | 0001 |  |  |
| 19A | E802 | 19B | E802 | 000 | 0001 | 7FF | 0002 | 0001 | 0001 |  |  |
| 19B | 0A00 | 18D | 0A00 | 7FF | 018D | 000 | 019B | 0001 | 0001 |  |  |
| 18D | 1207 | 18E | 1207 | 18D | 1207 | 000 | 018D | 0040 | 0101 |  |  |
| 18E | 2F40 | 18F | 2F40 | 18E | 0040 | 000 | 0040 | 0040 | 0001 |  |  |
| 18F | F0FD | 190 | F0FD | 18F | F0FD | 000 | 018F | 0040 | 0001 |  |  |
| 190 | 1206 | 191 | 1206 | 190 | 1206 | 000 | 0190 | 0006 | 0001 |  |  |
| 191 | 0680 | 192 | 0680 | 191 | 0680 | 000 | 0191 | 0600 | 0001 |  |  |
| 192 | 4809 | 193 | 4809 | 610 | 0006 | 000 | 0009 | 0606 | 0000 |  |  |
| 193 | EA08 | 194 | EA08 | 610 | 0606 | 000 | 0008 | 0606 | 0000 | 19C  610 | 0611  E0CC |
|  |  |  |  |  |  |  |  |  |  |
| 194 | DE02 | 197 | DE02 | 7FF | 0195 | 7FF | 0197 | 0606 | 0000 | 7FF | 0195 |
| 197 | A805 | 198 | A805 | 000 | 0001 | 7FF | 0005 | 0001 | 0000 |  |  |
| 198 | 0740 | 199 | 0740 | 198 | 0740 | 7FF | 0198 | 0000 | 0101 |  |  |
| 199 | F0FC | 196 | F0FC | 199 | F0FC | 7FF | FFFC | 0000 | 0101 |  |  |
| 196 | 0100 | 197 | 0100 | 196 | 0100 | 7FF | 0196 | 0000 | 0101 |  |  |
| 197 | A805 | 611 | 0100 | 196 | 0100 | 7FF | 0196 | 0000 | 0101 |  |  |
| 611 | 0000 | 612 | 0100 | 611 | 0000 | 7FF | 0196 | 0000 | 0101 |  |  |

# 

# Дополнительное задание

Организовать ввод символов с ВУ-8 (клавиатура) в кодировке Windows-1251, сохранение массива символов в памяти БЭВМ и вывод их на ВУ-7 (семисегментный индикатор).

# Выполнение дополнительного задания

Для ввода количества символов мною используется ВУ-3.

Т. к. ВУ-7 не поддерживает вывод символов, отличных от цифр и «-», мною было решено организовать вывод так:

1. При вводе с клавиатуры цифры на ВУ-7 выводится эта цифра.

2. При вводе любого другого символа на индикатор выводится «-», что сигнализирует о невозможности корректного вывода.

# Текст программы для дополнительного задания (Ассемблер)

|  |  |  |
| --- | --- | --- |
| **Маркер** | **Команда** | **Адресация** |
|  | **ORG** |  |
| **BEGIN:** | **CLA** |  |
| **NUM\_INP:** | **IN** | **7** |
| **ST** | **(CURSYMB)+** |
| **LD** | **(FIRSTSYMB)** |
| **INC** |  |
| **ST** | **COUNTER** |
| **ITERATOR1:** | **LOOP** | **COUNTER** |
| **JUMP** | **INPUT** |
| **JUMP** | **PRE\_OUTPUT** |
| **INPUT:** | **IN** | **0x19** |
| **AND** | **#0x40** |
| **BEQ** | **INPUT** |
| **IN** | **0x18** |
| **CALL** | **INPUT\_CHECKER** |
| **JUMP** | **ITERATOR1** |
| **PRE\_OUTPUT:** | **LD** | **(FIRSTSYMB)** |
| **INC** |  |
| **ST** | **COUNTER** |
| **LD** | **FIRSTSYMB** |
| **INC** |  |
| **ST** | **CURSYMB** |
| **ITERATOR2:** | **LOOP** | **COUNTER** |
| **JUMP** | **OUTPUT** |
| **HLT** |  |
| **OUTPUT:** | **CLA** |  |
| **CALL** | **OUTPUT\_CHECKER** |
| **AND** | **#0x0F** |
| **ADD** | **OFFSET** |
| **OUT** | **0x14** |
| **LD** | **OFFSET** |
| **ADD** | **#0x10** |
| **ST** | **OFFSET** |
| **JUMP** | **ITERATOR2** |
| **INPUT\_CHECKER:** | **PUSH** |  |
| **LD** | **COUNTER** |

|  |  |  |
| --- | --- | --- |
| **Маркер** | **Команда** | **Адресация** |
|  | **AND** | **#0x1** |
| **BEQ** | **INPUT\_SWAB** |
| **POP** |  |
| **ST** | **(CURSYMB)** |
| **RET** |  |
| **INPUT\_SWAB:** | **POP** |  |
| **SWAB** |  |
| **ADD** | **(CURSYMB)** |
| **ST** | **(CURSYMB)+** |
| **RET** |  |
| **OUTPUT\_CHECKER:** | **LD** | **COUNTER** |
| **AND** | **#0x1** |
| **BEQ** | **OUTPUT\_SWAB** |
| **LD** | **(CURSYMB)** |
| **AND** | **#0x7F** |
| **CMP** | **COND\_MAX** |
| **BPL** | **CHANGE** |
| **CMP** | **COND\_MIN** |
| **BMI** | **CHANGE** |
| **RET** |  |
| **OUTPUT\_SWAB:** | **LD** | **(CURSYMB)+** |
| **SWAB** |  |
| **AND** | **#0x7F** |
| **CMP** | **COND\_MAX** |
| **BPL** | **CHANGE** |
| **CMP** | **COND\_MIN** |
| **BMI** | **CHANGE** |
| **RET** |  |
| **CHANGE:** | **AND** | **#0x00** |
| **ADD** | **COND\_MAX** |
| **RET** |  |
| **FIRSTSYMB:** | **WORD** | **0x0610** |
| **CURSYMB:** | **WORD** | **0x0610** |
| **COUNTER:** | **WORD** | **0x0000** |
| **COND\_MAX:** | **WORD** | **0x003A** |
| **COND\_MIN:** | **WORD** | **0x0030** |
| **OFFSET:** | **WORD** | **0x0000** |

# Вывод

В процессе выполнения лабораторной работы я ознакомился с принципами создания комплекса программ в БЭВМ, способом связи между ними и получением общего результата работы. Узнал, как работает стек в БЭВМ и как в процессе выполнения программы узнать его состояние. На практике применил изученные приемы и детально обработал выданную в варианте программу.