Факультет программной инженерии и компьютерной техники

Основы профессиональной деятельности

Лабораторная работа №5

Вариант 2409

Преподаватель:

Покид Александр Сергеевич

Выполнил:

Андросов Иван Сергеевич

Р3110

Санкт-Петербург

2021

# 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 ввод данных с ВУ-2
2. Программа начинается с адреса 19E16. Размещаемая строка находится по адресу 55316.
3. Строка должна быть представлена в кодировке Windows-1251.
4. Формат представления строки в памяти: АДР1: СИМВ2 СИМВ1 АДР2: СИМВ4 СИМВ3 ... СТОП\_СИМВ.
5. Ввод или вывод строки должен быть завершен по символу c кодом 00 (NUL). Стоп символ является обычным символом строки и подчиняется тем же правилам расположения в памяти что и другие символы строки

**ПРОГРАММА:**

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | содержание | мнемоника | описание |
| 19A | 0000 | X | Переменная 1 |
| 19B | 0000 | Y | Переменная 2 |
| 19C | 0553 | address | Указатель на ячейку, в которую будет записано слово |
| 19D | 0000 | E0F | Символ конца строки |
| 19E | +0200 | CLA | Очистка аккумулятора |
| 19F | E19A | ST 19A | Очистка переменной 1 |
| 1A0 | E19B | ST 19B | Очистка переменной 2 |
| 1A1 | 1205 | IN 5 | Если ВУ-2 готово, идём дальше, если нет – ждём. |
| 1A2 | 2F40 | AND #0x0040 |
| 1A3 | F0FD | IF Z==1: IP – 3 -> IP |
| 1A4 | 1204 | IN 4 | КВУ-2 -> AC0…7  Получаем первый символ |
| 1A5 | E19A | ST 19A | Сохраняем в переменную 1 |
| 1A6 | 619D | SUB 19D | Смотрим равен ли введенный символ EOF. Если равен – сохраняем, заканчиваем, если нет, прыгаем дальше |
| 1A7 | F102 | IF Z==0: IP + 2 -> IP |
| 1A8 | EAF3 | ST (IP - 13)+ | Сохраняем со сдвигом |
| 1A9 | 0100 | HLT | Заканчиваем |
| 1AA | 0200 | CLA | Очищаем AC |
| 1AB | 1205 | IN 5 | Если ВУ-2 готово, идём дальше, если нет – ждём. |
| 1AC | 2F40 | AND #0x0040 |
| 1AD | F0FD | IF Z==1: IP – 3 -> IP |
| 1AE | 1204 | IN 4 | КВУ-2 -> AC0…7  Получаем второй символ |
| 1AF | E19B | ST 19B | Сохраняем его в переменную 2 |
| 1B0 | 0680 | SWAB | В AC меняем первые 2 и последние 2 байта (получается YY00) |
| 1B1 | 419A | ADD 19A | Складываем с первой переменной в которой хранится 00XX; В AC хранится YYXX |
| 1B2 | EAE9 | ST (IP - 23)+ | Сохраняем это значение |
| 1B3 | 0200 | CLA | Очищаем AC |
| 1B4 | A19B | LD 19B | Загружаем из второй переменной, в AC 00YY |
| 1B5 | 619D | SUB 19D | Это не конец файла ? начинаем заново : идём дальше |
| 1B6 | F1E7 | IF Z==0: IP - 25 |
| 1B7 | 0100 | HLT | Конец программы |

**Описание программы**

1. **Назначение программы и реализуемая ею функция**

Программа предназначена для ввода набора символов (строки) в кодировке на Windows-1251 по его готовности и сохранении в памяти БЭВМ. Ввод строки завершается символом 0D.

1. **Область представления исходных данных и результата:**Ячейки 553-XXX (STRING) || char[] – двух-символьные слова.

Ячейки 19A, 19B – 16 разрядные знаковые числа.

1. **Область допустимых значений исходных данных и результата:**  
   Значение каждой ячейки 553 - XXX в диапазоне Максимальная длина строки 1095 слов по два символа, т.е. 2189 символов строки + 1 символ = EOF (от 553 до 7FF + от 000 до 199)
2. **Расположение в памяти ЭВМ программы, исходных данных и результатов:**

\* Ячейки 553-XXX (STRING) – вводимая из ВУ-2 строка

\* 19A – X (используется как промежуточная переменная)

\* 19B - Y (вспомогательная переменная переменная)

\* 19C - ADDRESS (хранит адрес начала строки)

\* 19D - EOF (Завершающий символ)

**\*** Программа в ячейках 0х19E-0х1B7

1. Адреса первой и последней выполняемой команд программы:  
   Начало: 0x19E  
   Конец: 0x1B7

**Исходные данные:**

|  |  |
| --- | --- |
| **Символ** | **Кодировка windows-1251** |
| Л | CB |
| а | E0 |
| б | E1 |
| а | E0 |
| 5 | 35 |
| EOF | 0000 |

**Ожидаемые значения ячеек после выполнения программы с исходными данными:**

|  |  |
| --- | --- |
| **Ячейка** | **Значение** |
| 0х0553 | 0xE0CB |
| 0х0554 | 0xE0E1 |
| 0х0555 | 0x0035 |

**Полученные значения ячеек после выполнения программы с исходными данными:**

|  |  |
| --- | --- |
| **Ячейка** | **Значение** |
| 0х0553 | 0xE0CB |
| 0х0554 | 0xE0E1 |
| 0х0555 | 0x0035 |

*Программа работает согласно ожиданиям.*

**Трассировка** (согласно методичке, трассировка производится по двум символам из слова. Я взял символ «Л» в windows-1251 и символ EOF «00»)**:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Выполняемая команда | | Содержимое регистров процессора после выполнения команды | | | | | | | | Ячейка, содержимое которой изменилось после выполнения команды | |
| Адрес | Код | IP | CR | AR | DR | SP | BR | AC | NZVC | Адрес | Код |
| 19E | 0200 | 19F | 0200 | 19E | 0200 | 000 | 019E | 0000 | 0100 |  |  |
| 19F | E19A | 1A0 | E19A | 19A | 0000 | 000 | 019F | 0000 | 0100 | 19A | 0000 |
| 1A0 | E19B | 1A1 | E19B | 19B | 0000 | 000 | 01A0 | 0000 | 0100 | 19B | 0000 |
| 1A1 | 1205 | 1A2 | 1205 | 1A1 | 1205 | 000 | 01A1 | 0040 | 0100 |  |  |
| 1A2 | 2F40 | 1A3 | 2F40 | 1A2 | 0040 | 000 | 0040 | 0040 | 0000 |  |  |
| 1A3 | F0FD | 1A4 | F0FD | 1A3 | F0FD | 000 | 01A3 | 0040 | 0000 |  |  |
| 1A4 | 1204 | 1A5 | 1204 | 1A4 | 1204 | 000 | 01A4 | 00CB | 0000 |  |  |
| 1A5 | E19A | 1A6 | E19A | 19A | 00CB | 000 | 01A5 | 00CB | 0000 | 19A | 00CB |
| 1A6 | 619D | 1A7 | 619D | 19D | 0000 | 000 | 01A6 | 00CB | 0001 |  |  |
| 1A7 | F102 | 1AA | F102 | 1A7 | F102 | 000 | 0002 | 00CB | 0001 |  |  |
| 1AA | 0200 | 1AB | 0200 | 1AA | 0200 | 000 | 01AA | 0000 | 0101 |  |  |
| 1AB | 1205 | 1AC | 1205 | 1AB | 1205 | 000 | 01AB | 0040 | 0101 |  |  |
| 1AC | 2F40 | 1AD | 2F40 | 1AC | 0040 | 000 | 0040 | 0040 | 0001 |  |  |
| 1AD | F0FD | 1AE | F0FD | 1AD | F0FD | 000 | 01AD | 0040 | 0001 |  |  |
| 1AE | 1204 | 1AF | 1204 | 1AE | 1204 | 000 | 01AE | 0000 | 0001 |  |  |
| 1AF | E19B | 1B0 | E19B | 19B | 0000 | 000 | 01AF | 0000 | 0001 | 19B | 0000 |
| 1B0 | 0680 | 1B1 | 0680 | 1B0 | 0680 | 000 | 01B0 | 0000 | 0101 |  |  |
| 1B1 | 419A | 1B2 | 419A | 19A | 00CB | 000 | 01B1 | 00CB | 0000 |  |  |
| 1B2 | EAE9 | 1B3 | EAE9 | 553 | 00CB | 000 | FFE9 | 00CB | 0000 | 553 | 00CB |
| 1B3 | 0200 | 1B4 | 0200 | 1B3 | 0200 | 000 | 01B3 | 0000 | 0100 |  |  |
| 1B4 | A19B | 1B5 | A19B | 19B | 0000 | 000 | 01B4 | 0000 | 0100 |  |  |
| 1B5 | 619D | 1B6 | 619D | 19D | 0000 | 000 | 01B5 | 0000 | 0101 |  |  |
| 1B6 | F1E7 | 1B7 | F1E7 | 1B6 | F1E7 | 000 | 01B6 | 0000 | 0101 |  |  |
| 1B7 | 0100 | 1B8 | 0100 | 1B7 | 0100 | 000 | 01B7 | 0000 | 0101 |  |  |