Министерство высшего образования и науки Российской Федерации

Национальный научно-исследовательский университет ИТМО

Факультет программной инженерии и компьютерной техники

Лабораторная работа №5  
по дисциплине  
**«Основы профессиональной деятельности».**

Вариант №686.

Работу выполнил:

Афанасьев Кирилл Александрович,  
Студент группы P3106.  
Преподаватель:  
Афанасьев Дмитрий Борисович.

Санкт-Петербург, 2023

Оглавление

[Задание 3](#_Toc132804473)

[Текст исходной программы 3](#_Toc132804474)

[Описание программы 4](#_Toc132804475)

[Таблица трассировки выполнения команд 4](#_Toc132804476)

[Вывод 5](#_Toc132804477)

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.

Вариант 686:

1. Программа осуществляет асинхронный вывод данных на ВУ-3
2. Программа начинается с адреса 5D616. Размещаемая строка находится по адресу 5F316.
3. Строка должна быть представлена в кодировке КОИ-8.
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 0A (NL). Стоп символ является обычным символом строки и подчиняется тем же правилам расположения в памяти что и другие символы строки.

»

Текст исходной программы

*Таблица 1: Текст исходной программы.*

|  |
| --- |
| **Асинхронный вывод строки (в кодировке КОИ-8(R)) на ВУ-3 (до NL-символа).** |
| ORG 0x5D6 ; *Программа начинается с адреса 5D6*  STR: WORD 0x05F3 ; *Размещаемая строка находится по адресу 5F3*  POINTER: WORD ? ; *?? Указатель на текущие 2 символа* |
| START: CLA  LD STR  ST POINTER ; *Инициализируем указатель* |
| S1: IN 0x07 ; *Ожидание вывода первого символа*  AND #0x40 ; *Бит 6 SR == 0 («Готов» нажата?)*  BEQ S1 ; *Нет – «Спин-луп»*  LD (POINTER) ; *Иначе загрузим текущие 2 символа строки…*  OUT 0x06 ; *…И выведем первый символ на ВУ-3* |
| SXTB ; *Расширим знак, чтобы второй символ нам не мешал*  CMP #0x0A ; *Перед нами стоп-символ?*  BEQ PHLT ; *Да – на выход!* |
| S2: IN 0x07 ; *Ожидание вывода второго символа*  AND #0x40 ; *Бит 6 SR == 0 (“Готов” нажата?)*  BEQ S2 ; *Нет – «Спин-луп»*  LD (POINTER)+ ; *Иначе загрузим те же 2 символа строки, переместим указатель*  SWAB ; *Выберем второй символ в младшем байте…*  OUT 0x06 ; *…И выведем его на ВУ-3* |
| SXTB ; *Расширим знак, чтобы первый символ нам не мешал*  CMP #0x000A ; *Перед нами стоп-символ?*  BEQ PHLT ; *Да – на выход!*  JUMP S1 ; *И так далее...* |
| PHLT: HLT ; *Тот самый выход!* |
| ORG 0x5F3 ; *Строка начинается здесь*  WORD 0xD0F3 ; *Сп*  WORD 0xD3C1 ; *ас*  WORD 0xD4C9 ; *ит*  WORD 0x9AC5 ; *е*  WORD 0xD4D3 ; *ст*  WORD 0xC4D5 ; *уд*  WORD 0xCEC5 ; *ен*  WORD 0xCFD4 ; *то*  WORD 0x9AD7 ; *в*  WORD 0xC6E1 ; *Аф*  WORD 0xCEC1 ; *ан*  WORD 0xD3C1 ; *ас*  WORD 0xC5D8 ; *ье*  WORD 0xC1D7 ; *ва*  WORD 0x0A21 ; *!NL* |

*Окончание таблицы.*

Описание программы

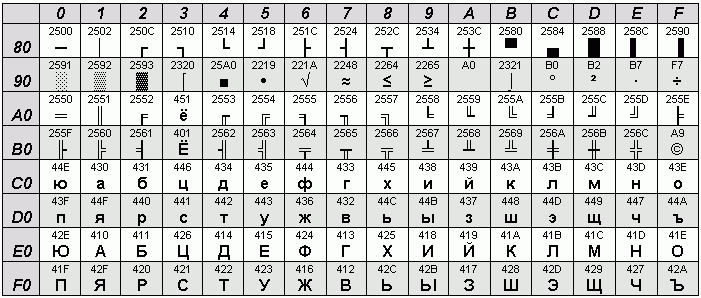
* Назначение программы: посимвольный асинхронный вывод строки, закодированной в КОИ-8(R) на ВУ-3.
* Описание исходных данных:
  + STR – Адрес 1-го элемента (начала) строки.
  + STR…0A(NL-символ) – Строка в кодировке КОИ-8(R).
  + ОПИ:
    - STR – беззнаковое 11-разрядное число.
    - Элемент строки – до 2-х восьмиразрядных символов в кодировке КОИ-8(R). Младший байт – 1-й символ элемента, старший байт – 2-й.
  + ОДЗ:
    - … Где “Длина строки” – количество символов в ней.
    - Символ строки:   
      *Рисунок 1. Таблица символов в кодировке КОИ-8(R).*
* Расположение в памяти ЭВМ программы и исходных данных:
  + Программа располагается в памяти в ячейках между адресами 5D6 и 5ED включительно (без учета строки).
  + Исходные данные должны располагаться в ячейках памяти:
    - STR – 5D6.
    - Строка: начиная с адреса STR, заканчивая символом 0x0A.
  + В программе также используется вспомогательная ячейка, находящаяся по адресу 0x5D7.
* Первая команда располагается в ячейке по адресу 5D8. Последняя – 5ED.

Таблица трассировки выполнения команд

*Таблица 2: Трассировка выполнения команд для первых двух символов.*

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Выполняемая команда | | Содержимое регистров процессора после выполнения команды | | | | | | | | | Ячейка, содержимое которой изменилось | |
| Адрес | Код команды | IP | CR | AR | DR | SP | BR | AC | NZVC | Адрес | | Новый код |
| 362 | AF40 | 363 | AF40 | 362 | 0040 | 000 | 0040 | 0040 | 0000 |  | |  |
| 363 | 0680 | 364 | 0680 | 363 | 0680 | 000 | 0363 | 4000 | 0000 |  | |  |
| 364 | 0500 | 365 | 0500 | 364 | 4000 | 000 | 0364 | 8000 | 1010 |  | |  |
| 365 | EEFB | 366 | EEFB | 361 | 8000 | 000 | FFFB | 8000 | 1010 | 361 | | 8000 |
| 366 | AF04 | 367 | AF04 | 366 | 0004 | 000 | 0004 | 0004 | 0000 |  | |  |

*Окончание таблицы.*

Вывод

Во время выполнения данной лабораторной работы я ознакомился с программой асинхронного обмена данных через внешние устройства с Базовой ЭВМ, со способами представления строк, командами ввода-вывода, а также общей организацией системы ввода-вывода в Базовой ЭВМ.