

Федеральное государственное автономное образовательное

учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Направление подготовки: 09.03.04 – Системное и прикладное программное обеспечение

Дисциплина «Основы профессиональной деятельности»

**Отчёт по лабораторной работе №5**

**Асинхронный обмен данными с ВУ**

Вариант №15515

Выполнил

Галак Екатерина Анатольевна

P3115

Проверил

Блохина Елена Николаевна

Санкт – Петербург, 2025

Оглавление

[Задание 3](#_Toc195010291)

[Текст исходной программы и данные 3](#_Toc195010292)

[Программа на языке Ассемблера БЭВМ 4](#_Toc195010293)

[Назначение программы 4](#_Toc195010294)

[Описание и назначение исходных данных, область представления и область допустимых значений исходных данных и результата 4](#_Toc195010295)

[Расположение в памяти ЭВМ программы, исходных данных и результатов 5](#_Toc195010296)

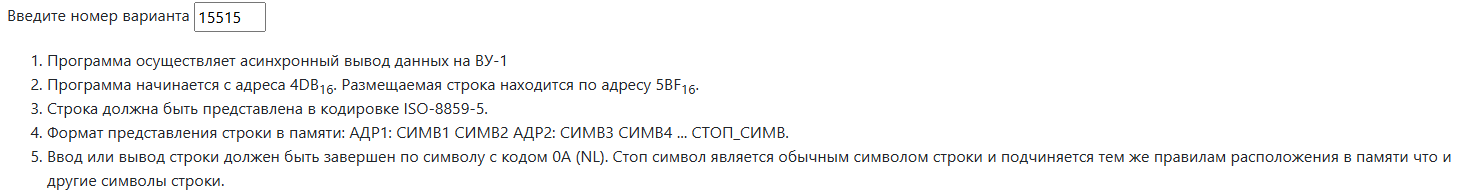
[Адреса первой и последней выполняемой команд программы 5](#_Toc195010297)

[Трассировка 5](#_Toc195010298)

[Заключение 7](#_Toc195010299)

# 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.



Исходная строка: Береза(

|  |  |  |  |
| --- | --- | --- | --- |
| Символ | ISO-8859-5 | UTF-8 | UTF-16 |
| Б | B1 | DO 91 | 04 11 |
| е | D5 | D0 B5 | 04 35 |
| р | E0 | D1 80 | 04 40 |
| е | D5 | D0 B5 | 04 35 |
| з | D7 | D0 B7 | 04 37 |
| а | D0 | D0 B0 | 04 30 |
| ( | 28 | 00 28 | 00 28 |

# Текст исходной программы и данные

Данные:

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Содержимое ячейки памяти | Обозначение | Комментарии |
| 4D7 | 05BF | ADDR | Адрес начала строки |
| 4D8 | 0000 | I | Адрес текущей ячейки с символами |
| 4D9 | 0000 | BUFF | Ячейка для временного хранения данных |
| 4DA | 000A | STOP\_C | Стоп-символ |

Текст исходной программы:

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарии |
| 4DB | 0200 | CLA | Очистка AC |
| 4DC | AEFA | LD (IP – 6) | Загрузка в AC переменной ADDR, расположенную по адресу (4D7) |
| 4DD | EEFA | ST (IP – 6) | Загрузка в переменную I адреса текущего символа строки (первый символ) |
| 4DE | 1203 | IN 3 | Ожидание готовности первого символа (SR (#3) -> AC) |
| 4DF | 2F40 | AND #40 | Бит 6 SR == 0 (“Готов” нажата?) (M & AC -> AC) |
| 4E0 | F0FD | BEQ (IP – 3) | Нет - «Спин-луп» (IF Z == 1 then IP – 3 + 1 -> IP) |
| 4E1 | AAF6 | LD (IP - 10)+ | Загрузка в AC значения по адресу, записанном в ячейке 4D8 (I). К значению ячейки 4D8 прибавляем единицу |
| 4E2 | EEF6 | ST (IP - 10) | Загрузка AC в BUFF (4D9) |
| 4E3 | 0680 | SWAB | Обмен старшего и младшего байта |
| 4E4 | 0600 | SXTB | Расширение знака младшего байта |
| 4E5 | 7EF4 | CMP (IP – 12) | Установка флагов по результату AC – (4DA) (стоп-символ) |
| 4E6 | F00A | BEQ (IP + 10) | IF Z == 1 then IP + 10 + 1 -> IP (если элемент равен стоп-символу, переходим в ячейку 4F1) |
| 4E7 | 1302 | OUT 2 | Вывод первого символа (младший байт AC) в ВУ-1 |
| 4E8 | 1203 | IN 3 | Ожидание готовности первого символа (SR (#3) -> AC) |
| 4E9 | 2F40 | AND #40 | Бит 6 SR == 0 (“Готов” нажата?) (M & AC -> AC) |
| 4EA | F0FD | BEQ (IP – 3) | Нет - «Спин-луп» (IF Z == 1 then IP – 3 + 1 -> IP) |
| 4EB | AEED | LD (IP - 19) | Загрузка в AC значения из BUFF (4D9) |
| 4EC | 0600 | SXTB | Расширение знака младшего байта |
| 4ED | 7EEC | CMP (IP – 20) | Установка флагов по результату AC – (4DA) (стоп-символ) |
| 4EE | F002 | BEQ (IP + 2) | IF Z == 1 then IP + 2 + 1 -> IP (если элемент равен стоп-символу, переходим в ячейку 4F1) |
| 4EF | 1302 | OUT 2 | Вывод второго символа (младший байт AC) в ВУ-1 |
| 4F0 | CEED | JUMP (IP – 19) | Переход в ячейку 4DE |
| 4F1 | 0100 | HLT | Останов |

# Программа на языке Ассемблера БЭВМ

ORG 0x4D7

ADDR: **WORD** $ARRAY ; string address

I: **WORD** 0 ; current chars of the string

BUFF: **WORD** 0 ; buffer cell

STOP\_C: **WORD** 0x0A ; EOF char

START: CLA ; clear

LD ADDR

ST I ; I = ADDR

S1: **IN** 3 ; \

**AND** #0X40 ; -> SPIN-LOOP while not ready

BEQ S1 ; /

LD (I)+ ; increment of the pointer to the current chars

ST BUFF ; load 2 chars (2 chars in one cell) into a buffer cell (BUFF)

SWAB ; \

SXTB ; -> sign extension (get the first char)

**CMP** STOP\_C ; compare with the stop char (STOP\_C)

BEQ STOP ; if equal, then end the output

**OUT** 2 ; output the first char

S2: **IN** 3 ; \

**AND** #0X40 ; -> SPIN-LOOP while not ready

BEQ S2 ; /

LD BUFF ; load the buffer cell

SXTB ; sign extension (get the second char)

**CMP** STOP\_C ; compare with the stop char (STOP\_C)

BEQ STOP ; if equal, then end the output

**OUT** 2 ; output the second char

JUMP S1 ; move on to the output of a new cell (2 chars)

STOP: **HLT** ; exit program

ORG 0x5BF

ARRAY: **WORD** 0xB1D5

**WORD** 0xE0D5

**WORD** 0xD7D0

**WORD** 0x280A

# Назначение программы

Назначение программы:

Асинхронный вывод данных на ВУ-1 (строка выводится на ВУ-1).

Строка представлена в кодировке ISO-8859-5, формат представления в памяти: АДР1: СИМВ1 СИМВ2, АДР2: СИМВ3 СИМВ4 … СТОП\_СИМВЛ. Признаком завершения вывода является символ 0A (NL).

# Описание и назначение исходных данных, область представления и область допустимых значений исходных данных и результата

Описание и назначение исходных данных:

* ADDR – адрес начала строки в памяти БЭВМ.
* I – адрес текущей ячейки памяти с символами строки (изначально равна ADDR).
* BUFF – ячейка для временного хранения данных.
* STOP\_C – стоп-символ (константа).
* СИМВ – символ строки.

Область представления:

* СИМВ – 8-ми разрядное целое беззнаковое число

[0; 28 – 1], но символы хранятся в 16-ти разрядных ячейках (беззнаковое 16-ти разрядное число) по два в каждом.

* I, ADDR – 11-разрядные беззнаковые числа (адреса БЭВМ)
* BUFF, STOP\_C – 16-ти разрядные беззнаковые числа

[0; 216 – 1]

Область допустимых значений (ОДЗ):

# Расположение в памяти ЭВМ программы, исходных данных и результатов

* Программа расположена в ячейках **4DB** – **4F1**
* Исходные данные для программы расположены в ячейках **4D7 – 4DA**
* Исходная строка расположена в ячейках **5BF** – **7FF**

# Адреса первой и последней выполняемой команд программы

* Первая – **4DB**
* Последняя – **4F1**

# Трассировка

Таблица трассировки программы для вывода первых двух символов строки (Б = B1, е = D5):

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** | **Новый код** |
| 4DB | 0200 | 4DC | 0200 | 4DB | 0200 | 000 | 04DB | 0000 | 0100 |  |  |
| 4DC | AEFA | 4DD | AEFA | 4D7 | 05BF | 000 | FFFA | 05BF | 0000 |  |  |
| 4DD | EEFA | 4DE | EEFA | 4D8 | 05BF | 000 | FFFA | 05BF | 0000 | 4D8 | 05BF |
| 4DE | 1203 | 4DF | 1203 | 4DE | 1203 | 000 | 04DE | 0540 | 0000 |  |  |
| 4DF | 2F40 | 4E0 | 2F40 | 4DF | 0040 | 000 | 0040 | 0040 | 0000 |  |  |
| 4E0 | F0FD | 4E1 | F0FD | 4E0 | F0FD | 000 | 04E0 | 0040 | 0000 |  |  |
| 4E1 | AAF6 | 4E2 | AAF6 | 5BF | B1D5 | 000 | FFF6 | B1D5 | 1000 | 4D8 | 05C0 |
| 4E2 | EEF6 | 4E3 | EEF6 | 4D9 | B1D5 | 000 | FFF6 | B1D5 | 1000 | 4D9 | B1D5 |
| 4E3 | 0680 | 4E4 | 0680 | 4E3 | 0680 | 000 | 04E3 | D5B1 | 1000 |  |  |
| 4E4 | 0600 | 4E5 | 0600 | 4E4 | 0600 | 000 | 04E4 | FFB1 | 1000 |  |  |
| 4E5 | 7EF4 | 4E6 | 7EF4 | 4DA | 000A | 000 | FFF4 | FFB1 | 1001 |  |  |
| 4E6 | F00A | 4E7 | F00A | 4E6 | F00A | 000 | 04E6 | FFB1 | 1001 |  |  |
| 4E7 | 1302 | 4E8 | 1302 | 4E7 | 1302 | 000 | 04E7 | FFB1 | 1001 | ВУ-1 | B1 |
| 4E8 | 1203 | 4E9 | 1203 | 4E8 | 1203 | 000 | 04E8 | FF40 | 1001 |  |  |
| 4E9 | 2F40 | 4EA | 2F40 | 4E9 | 0040 | 000 | 0040 | 0040 | 0001 |  |  |
| 4EA | F0FD | 4EB | F0FD | 4EA | F0FD | 000 | 04EA | 0040 | 0001 |  |  |
| 4EB | AEED | 4EC | AEED | 4D9 | B1D5 | 000 | FFED | B1D5 | 1001 |  |  |
| 4EC | 0600 | 4ED | 0600 | 4EC | 0600 | 000 | 04EC | FFD5 | 1001 |  |  |
| 4ED | 7EEC | 4EE | 7EEC | 4DA | 000A | 000 | FFEC | FFD5 | 1001 |  |  |
| 4EE | F002 | 4EF | F002 | 4EE | F002 | 000 | 04EE | FFD5 | 1001 |  |  |
| 4EF | 1302 | 4F0 | 1302 | 4EF | 1302 | 000 | 04EF | FFD5 | 1001 | ВУ-1 | D5 |
| 4E0 | CEED | 4DE | CEED | 4F0 | 04DE | 000 | FFED | FFD5 | 1001 |  |  |

# Заключение

Во время выполнения лабораторной работы я познакомилась с внешними устройствами и с их регистрами, асинхронным обменом данных в БЭВМ и с командами IN и OUT.

Модернизировать программу следующим образом: вывод осуществляется на бегущую строку. ВАЖНО строка должна быть читаемой, заглавные и строчные буквы должны различаться. (Оказывается, нужно было вывести только сообщение про березу, но моя программа универсальная (написана не самым изящным способом, но работает), может вывести русские буквы в двух регистрах и некоторые символы)