Федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский национальный исследовательский университет ИТМО»

**Институт ПКиКТ**

**Дисциплина: Основы Профессиональной Деятельности**

**Лабораторная работа №5**

**«Асинхронный обмен данными с ВУ»**

**Вариант №1323**

Выполнил: Сиразетдинов Азат Ниязович

Группа: Р3116

Преподаватель: Афанасьев

Дмитрий Борисович

2023г.

Оглавление

[Задание 3](#_Toc133406820)

[Выполнение работы 4](#_Toc133406821)

[Текст исходной программы: 4](#_Toc133406822)

[Текст исходной программы на ассемблере 5](#_Toc133406823)

[Описание программы 6](#_Toc133406824)

[Назначение программы 6](#_Toc133406825)

[Область представления: 6](#_Toc133406826)

[Область допустимых значений: 6](#_Toc133406827)

[Расположение программы и данных: 6](#_Toc133406828)

# 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.

Программа осуществляет асинхронный ввод данных с **ВУ-2**

Программа начинается с адреса **3F116**. Размещаемая строка находится по адресу **63B16**. Строка должна быть представлена в кодировке **Windows-1251**.

Формат представления строки в памяти: АДР0: ДЛИНА АДР1: СИМВ2 СИМВ1 АДР2:

СИМВ4 СИМВ3 ..., где ДЛИНА - 16 разрядное слово, где значащими являются 8 младших бит.

Ввод строки начинается со ввода количества символов (1 байт), и должен быть завершен по вводу их необходимого количества.

# Выполнение работы

## Текст исходной программы:

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарии |
| 3F1 | 0200 | CLA | Очистка аккумулятора |
| 3F2 | AE16 | LD (IP+16) | Загрузка адреса массива из вводных данных |
| 3F3 | EE16 | ST (IP+16) | Сохранение адреса массива |
| 3F4 | 1205 | IN 5 | Спин-луп по флагу готов ВУ-2 |
| 3F5 | 2F40 | AND #0X40 |
| 3F6 | F0FD | BEQ 3F4 |
| 3F7 | 1204 | IN 4 | Чтение размера массива из ВУ-2 |
| 3F8 | EA11 | ST (IP+11)+ | Постинкрементное сохранение длины в первый элемент массива |
| 3F9 | EE11 | ST (IP+11) | Сохранение длины в ячейку цикла |
| 3FA | 1205 | IN 5 | Спин-луп по флагу готов ВУ-2 |
| 3FB | 2F40 | AND #40 |
| 3FC | F0FD | BEQ 3FA |
| 3FD | 1204 | IN 4 | Чтение элемента массива из ВУ-2 |
| 3FE | 0680 | SWAB | Обмен старшего и младшего байтов |
| 3FF | EE0C | ST (IP+C) | Сохранение аккумулятора в буфер |
| 400 | 1205 | IN 5 | Спин-луп по флагу готов ВУ-2 |
| 401 | 2F40 | AND #40 |
| 402 | F0FD | BEQ 400 |
| 403 | AE08 | LD (IP+8) | Чтение значение старшего байта из буфера |
| 404 | 1204 | IN 4 | Чтение элемента массива из ВУ-2 |
| 405 | EA04 | ST (IP+4)+ | Постинкрементное сохранение аккумулятора в массиве |
| 406 | 8E04 | LOOP (IP+4) | Цикл по длине массива |
| 407 | CEF2 | JUMP 3FA |
| 408 | 0100 | HLT | Останов |

## Текст исходной программы на ассемблере

ORG 0x3F1

START**:** CLA

LD RES\_ADDR

|  |  |  |
| --- | --- | --- |
|  | **ST** | RES\_ADDR\_BUF |
|  | | |

;Read count of letters

S1**:** **IN** 5

**AND** #0X40

BEQ S1 ; spin-loop

**IN** 4

|  |  |  |
| --- | --- | --- |
|  | **ST** | **(**RES\_ADDR\_BUF**)+** |
|  | **ST** | LOOP\_CNT |
|  |

;Read highest byte

S2**:** **IN** 5

**AND** #0x40

BEQ S2 ; spin-loop

**IN** 4

SWAB

|  |  |  |
| --- | --- | --- |
|  | **ST** | BUF |
|  | | |

;Read lowest byte

S3**:** **IN** 5

**AND** #0x40

BEQ S3 ; spin-loop

LD BUF **IN** 4

|  |  |  |
| --- | --- | --- |
|  | **ST** | **(**RES\_ADDR\_BUF**)+** |
| **LOOP** LOOP\_CNT | | |

JUMP S2

**HLT**

RES\_ADDR**:** **WORD** 0X63B ; ссылка на адрес первого элемента

RES\_ADDR\_BUF**:** **WORD** **?**

LOOP\_CNT**:** **WORD** **?**

BUF**:** **WORD** **?**

# Описание программы

## Назначение программы

Программа осуществляет асинхронный ввод данных с ВУ-2

## Область представления:

RES\_ADDR – беззнаковое целое 16 разрядное число

ARR\_LEN – беззнаковое целое 16 разрядное число, значащими являются 8 младших бит

## Область допустимых значений:

ARR\_LEN   ∈ [00;  FF]

RES\_ADDR   ∈ [0010;  3F0-ARR\_LEN] ∪ [413; 7FE-ARR\_LEN]

## Расположение программы и данных:

Программа располагается в ячейках с 3F1 по 408

Исходные данные располагаются:

* RES\_ADDR – в ячейке 409
* ARR\_LEN – вводится в ВУ-2 сначала

Результат располагается в ячейках с (409) по (409) + ARR\_LEN