

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерных техники

Лабораторная работа по ОПД №5

Вариант 9965

Выполнил: Баукин Максим Александрович

Группа: Р3132

Принимающий: Данилов П. Ю.

Должность: Лаборант факультета ПИиКТ

Г. Санкт-Петербург, 2024 г.

## Оглавление

Описание задания .....	3
Ход работы .....	3
Текст программы на ассемблере.....	5
Описание программы .....	6

## Описание задания

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 на **ВУ-2**

Программа начинается с адреса **4D3<sub>16</sub>**. Размещаемая строка находится по адресу **5AA<sub>16</sub>**.

Строка должна быть представлена в кодировке КОИ-8.

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.

## Ход работы

Адрес	Код команды	Мнемоника	Комментарии
4D3	0200	CLA	Очистка аккумулятора
4D4	AE17	LD (IP+17)	Загрузка адреса массива из вводных данных
4D5	EE17	ST (IP+17)	Сохранение адреса массива
4D6	1205	IN 5	Спин-луп по флагу готов ВУ-2
4D7	2F40	AND #0X40	
4D8	F0FD	BEQ 4D6	
4D9	1204	IN 4	Чтение размера массива из ВУ-2

<b>4DA</b>	<b>EA12</b>	<b>ST (IP+12)+</b>	<b>Автоинкрементное сохранение длины в первый элемент массива</b>
<b>4DB</b>	<b>EE12</b>	<b>ST (IP+12)</b>	<b>Сохранение длины в ячейку цикла</b>
<b>4DC</b>	<b>1205</b>	<b>IN 5</b>	<b>Спин-луп по флагу готов ВУ-2</b>
<b>4DD</b>	<b>2F40</b>	<b>AND #40</b>	
<b>4DE</b>	<b>F0FD</b>	<b>BEQ 4DC</b>	
<b>4DF</b>	<b>1204</b>	<b>IN 4</b>	<b>Чтение элемента массива из ВУ-2</b>
<b>4E0</b>	<b>0680</b>	<b>SWAB</b>	<b>Обмен старшего и младшего байтов</b>
<b>4E1</b>	<b>EE0D</b>	<b>ST (IP+13)</b>	<b>Сохранение аккумулятора в буфер</b>
<b>4E2</b>	<b>EB0B</b>	<b>ST -(IP+11)</b>	<b>Автодекрементное сохранение длины массива</b>
<b>4E3</b>	<b>1205</b>	<b>IN 5</b>	<b>Спин-луп по флагу готов ВУ-2</b>
<b>4E4</b>	<b>2F40</b>	<b>AND #40</b>	
<b>4E5</b>	<b>F0FD</b>	<b>BEQ 4E2</b>	
<b>4E6</b>	<b>AE08</b>	<b>LD (IP+8)</b>	<b>Чтение значение старшего байта из буфера</b>
<b>4E7</b>	<b>1204</b>	<b>IN 4</b>	<b>Чтение элемента массива из ВУ-2</b>
<b>4E8</b>	<b>EA04</b>	<b>ST (IP+4)+</b>	<b>Автоинкрементное сохранение аккумулятора в массиве</b>
<b>4E9</b>	<b>8E04</b>	<b>LOOP (IP+4)</b>	<b>Цикл по длине массива</b>
<b>4EA</b>	<b>CEF2</b>	<b>JUMP 4DC</b>	
<b>4EB</b>	<b>0100</b>	<b>HLT</b>	<b>Останов</b>

## Текст программы на ассемблере

**ORG 0x4D3**

**START:     CLA**

**LD   RES\_ADDR**

**ST   RES\_ADDR\_BUF**

**S1:   IN     5**

**AND  #0X40**

**BEQ  S1**

**IN     4**

**ST    (RES\_ADDR\_BUF)+**

**ST    ARR\_LEN**

**S2:   IN     5**

**AND  #0x40**

**BEQ         S2**

**IN     4**

**SWAB**

**ST    BUF**

**ST    -(ARR\_LEN)**

**S3:   IN     5**

**AND  #0x40**

**BEQ  S3**

**LD    BUF**

**IN     4**

**ST    (RES\_ADDR\_BUF)+**

**LOOP       ARR\_LEN**

**JUMP       S2**

## HLT

RES_ADDR:	WORD	0X5AA
RES_ADDR_BUF:	WORD	?
ARR_LEN:	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  $\in [00; FF]$

RES\_ADDR  $\in [0010; 4D2-ARR\_LEN] \cup [4F0; 7FE-ARR\_LEN]$

### *Расположение программы и данных:*

Программа располагается в ячейках с 4D3 по 4EB

Исходные данные располагаются:

- RES\_ADDR – в ячейке 4EC
- ARR\_LEN – вводится в ВУ-2 сначала

Результат располагается в ячейках с (5AA) по (5AA) + ARR\_LEN