*Федеральное государственное бюджетное образовательное*

*учреждение высшего профессионального образования*

*«Санкт-Петербургский национальный исследовательский*

*университет информационных технологий, механики и оптики»*

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

## **ЛАБОРАТОРНАЯ РАБОТА №5**

по дисциплине

«Основы профессиональной деятельности»

Вариант № 11516

***Выполнил:***

Студент группы P3113

Бободжонов Комронджон

***Преподаватель:***

Ткешелашвили Н.М.

Санкт-Петербург, 2023 г.

## Содержание

[Текст задания 3](#_Toc127641422)

[Описание программы 3](#_Toc127641423)

[Вывод 6](#_Toc127641424)

[Таблица трассировки 7](#_Toc127641425)

# 

# 

# Текст 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.

1. Программа осуществляет асинхронный вывода данных с ВУ-1
2. Программа начинается с адреса 58F16. Размещаемая строка находится по адресу 5F316.
3. Строка должна быть представлена в кодировке ISO-8859-5.
4. Формат представления строки в памяти: АДР0: ДЛИНА АДР1: СИМВ1 СИМВ2 АДР2: СИМВ3 СИМВ4 ..., где ДЛИНА - 16 разрядное слово, где значащими являются 8 младших бит.
5. Вывод строки начинается со вывода количества символов (1 байт), и должен быть завершен по выводу их необходимого количества.

Код на ассемблере

ORG 0x58F

START: CLA

LINE\_LENGTH\_OUT: IN 3

AND #0x40

BEQ LINE\_LENGTH\_OUT

LD (LENGTH\_ADDR)

OUT 2

FIRST\_OUT: IN 3

AND #0x40

BEQ FIRST\_OUT

LD (LINE\_ADDR)+

SWAB

OUT 2

SWAB

ST RES

LD (LENGTH\_ADDR)

DEC

CMP 0x0

BEQ EXIT

ST (LENGTH\_ADDR)

SECOND\_OUT: IN 3

AND #0x40

BEQ SECOND\_OUT

LD RES

OUT 2

CLA

LOOP (LENGTH\_ADDR)

JUMP FIRST\_OUT

HLT

EXIT: HLT

LENGTH\_ADDR: WORD 0x5F3

LINE\_ADDR: WORD 0x5F4

RES: WORD ?

ORG 0x5F3

WORD 0x0005

WORD 0xC4B0

WORD 0xB7B0

WORD 0xBD00

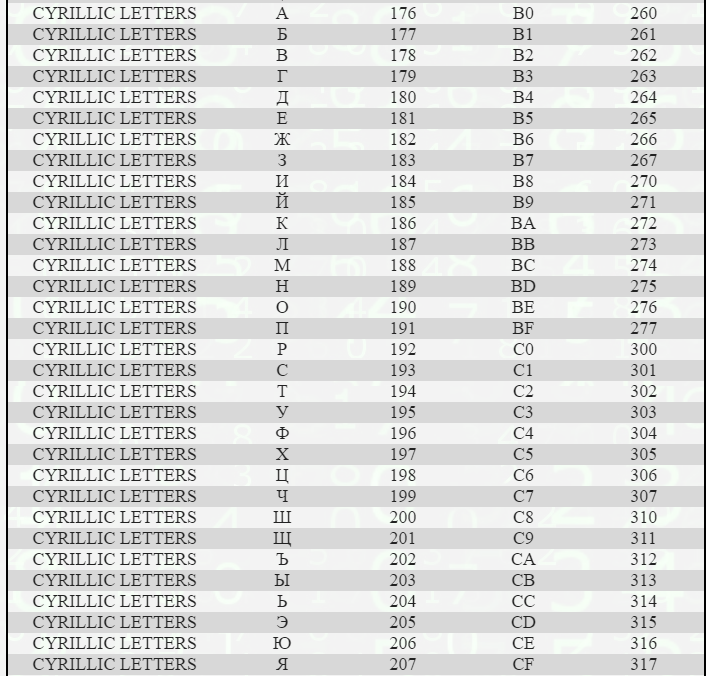
**Описание программы**

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Содержимое | Мнемоника | Описание |
| 58F | 0200 | CLA | 0->AC |
| 590 | 1203 | IN 3 | Записывает младший байт из SR ВУ-1 в AC |
| 591 | 2F40 | AND #0x40 | M & AC -> AC |
| 592 | F0FD | BEQ IP+FD | IF Z == 1 => IP + FD -> IP |
| 593 | A818 | LD (IP+18) | Косвенная относительная загрузка:  (IP+18) -> AC |
| 593 | 1302 | OUT 2 | Выводит младший бaйт AC в ВУ-1 |
| 594 | 1203 | IN 3 | Записывает младший байт из SR ВУ-1 в AC |
| 595 | 2F40 | AND #0x40 | M & AC -> AC |
| 596 | F0FD | BEQ IP+FD | IF Z == 1 => IP + FD -> IP |
| 597 | AA14 | LD (IP+14)+ | Косвенная автоинкрементная загрузка:  (IP+14) -> AC, IP+14+1 -> IP |
| 598 | 0680 | SWAB | Меняет местами младший и старший байт AC и записывает в  AC |
| 599 | 1302 | OUT 2 | Выводит младший бaйт AC в ВУ-1 |
| 600 | 0680 | SWAB | Меняет местами младший и старший байт AC и записывает в  AC |
| 601 | EE11 | ST M(IP+11) | AC -> M(IP+11) |
| 602 | A80E | LD (IP+0E) | Косвенная относительная загрузка:  (IP+0E) -> AC |
| 603 | 0740 | DEC | AC – 1 -> AC |
| 604 | 7000 | CMP 0x0 | AC – M -> N, Z, V, C |
| 605 | F00A | BEQ IP+0A | IF Z == 1 => IP + 0A -> IP |
| 606 | E80A | ST (IP+0A) | AC -> (IP+0A) |
| 607 | 1203 | IN 3 | Записывает младший байт из SR ВУ-1 в AC |
| 608 | 2F40 | AND #0x40 | M & AC -> AC |
| 609 | F0FD | BEQ IP+FD | IF Z == 1 => IP + FD -> IP |
| 60A | AE08 | LD M(IP+8) | Прямая относительная загрузка:  M(IP+8) -> AC |
| 60B | 1302 | OUT 2 | Выводит младший бaйт AC в ВУ-1 |
| 60C | 0200 | CLA | 0 -> AC |
| 60D | 8803 | LOOP (IP+3) | (IP+3) – 1 -> (IP+3), IF (IP+3) <= 0 => IP + 1 -> IP |
| 60E | CEEB | JUMP IP+EB | Прямой абсолютный прыжок IP+EB -> IP |
| 60F | 0100 | HLT | Останов |
| 610 | 0100 | HLT | Останов |
| 611 | 05AD | 5F3 | Адрес длины слова |
| 612 | 05AE |  | Текущий адрес пары букв |
| 5F3 | 0005 |  | Длина слова |
| 5F4 | C4B0 |  | Первая пара букв слова |
| 5F5 | B7B0 |  | Вторая пара букв слова |
| 5F6 | BD00 |  | Третья пара букв слова |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# ОП и ОДЗ исходных данных и результата

**Область представления:**

* RES – 16-разрядная ячейка для хранения 2х символов. Старший байт – код первого символа, младший байт – код второго символа
* ADR – 11-разрядное беззнаковое число. Ячейка для хранения адреса начала символов слова



**Область определения:**

* 8-ричный код символа для ввода (0x20; 0x7F) \ (0xA0; 0xFF)



(Исключены служебные символы)

Длина строки: Начало по адресу 0x5AD, length ∈ [0, 2\*(7FF-5AD-1)]

Кодировка ISO-8859-5:

Ф - 1100 0100  
А - 1011 0000  
З - 1011 0111  
А - 1011 0000  
Н - 1011 1101

Трассировки(строчка: ФАЗАН)

ORG 0x58F

START: CLA

LINE\_LENGTH\_OUT: IN 3

AND #0x40

BEQ LINE\_LENGTH\_OUT

LD (LENGTH\_ADDR)

OUT 2

FIRST\_OUT: IN 3

AND #0x40

BEQ FIRST\_OUT

LD (LINE\_ADDR)+

SWAB

OUT 2

SWAB

ST RES

LD (LENGTH\_ADDR)

DEC

CMP 0x0

BEQ EXIT

ST (LENGTH\_ADDR)

SECOND\_OUT: IN 3

AND #0x40

BEQ SECOND\_OUT

LD RES

OUT 2

CLA

LOOP (LENGTH\_ADDR)

JUMP FIRST\_OUT

HLT

EXIT: HLT

LENGTH\_ADDR: WORD 0x5F3

LINE\_ADDR: WORD 0x5F4

RES: WORD ?

ORG 0x5F3

WORD 0x0005

WORD 0xC4B0

WORD 0xB7B0

WORD 0xBD00

*Трассировка производится для первых двух символов*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Выполняемая команда | | Содержимое регистров после выполнения команды | | | | | | | | Ячейка, содержимое которой изменилось после выполнения команды | | | |
| Адрес | Содержимое | IP | CR | AR | DR | SP | BR | AC | NZVC | | Адрес | Содержимое |
| xxx | xxxx | xxx | xxxx | xxx | xxxx | xxx | xxxx | xxxx | xxxx | | xxx | xxxx |

58F 0200 59F 0200 58F 0200 58F 0000 0100

590 1203 591 1203 591 1203 590 0000 0100

591 2F40 592 2F40 591 0040 0000 0040

592 FOFD 593 FOFD 592 FOFD 0000 0040

593 A818 594 A818 593 A818 0000 0000 0000

594 1302 595 1302 594 1302 0000 0000 0000

595 1203 596 1203 595 1203 0000 0000 0001

596 2F40 597 2F40 596 0040 0000 0040

597 FOFD 598 FOFD 597 FOFD 0000 0040

598 AA14 599 AA14 598 C4B0 0000 C4B0 1000 5AD 05F5

599 0680 59A 0680 599 0680 0000 BOC4 1000

59A 1302 59B 1302 59A 1302 0000 C4B0 1000

59B 0680 59C 0680 59B 0680 0000 C4B0 1000

59C EE11 59D EE11 59C C4B0 0000 C4B0 1000 5AE C4B0

59D A80E 59E A80E 59D 5F3 0005 0005 5AE C4B0

59E 0740 59F 0740 59E 0740 0000 0004 0001

59F 7000 5A0 7000 59F 7000 0000 0004 0001

5A0 F00A 5A1 F00A 5A0 F00A 0001 0001

5A1 E80A 5A2 E80A 5A1 E80A 0001 5F3 0004

5A2 1203 5A3 1203 5A2 1203 0001 0001

5A3 2F40 5A4 2F40 5A3 0040 0001 0040

5A4 FOFD 5A5 FOFD 5A4 FOFD 0000 0040

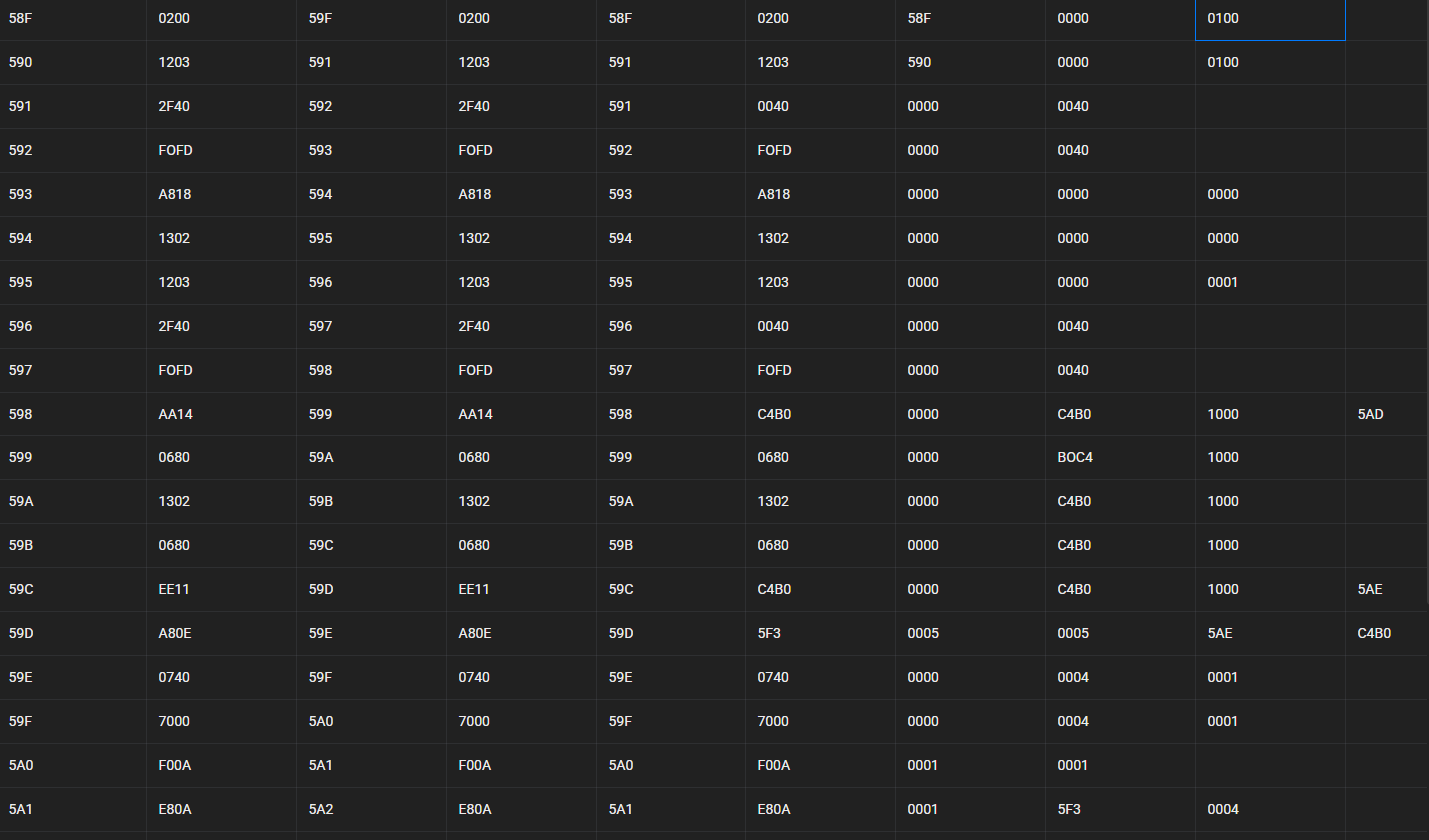
5A5 AE08 5A6 AE08 5A5 C4B0 0000 C4B0 1001

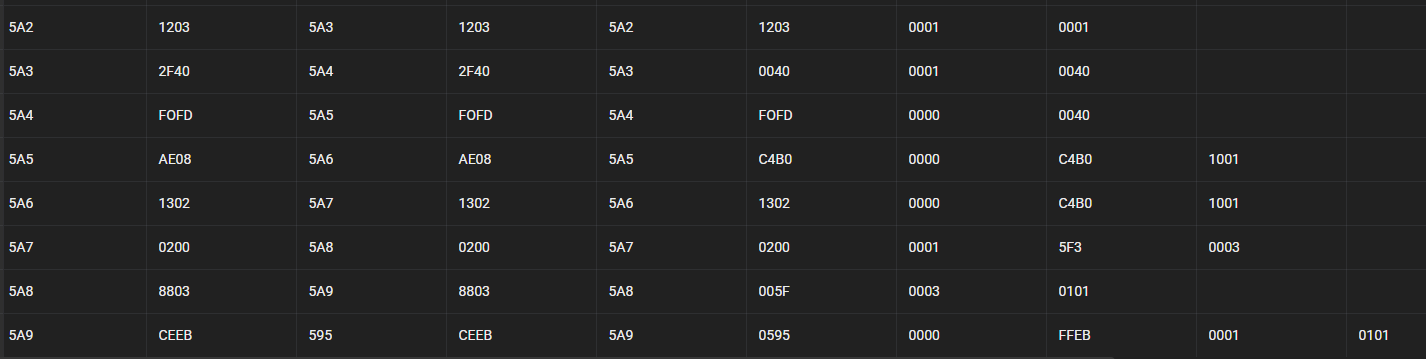
5A6 1302 5A7 1302 5A6 1302 0000 C4B0 1001

5A7 0200 5A8 0200 5A7 0200 0001 5F3 0003

5A8 8803 5A9 8803 5A8 005F 0003 0101

5A9 CEEB 595 CEEB 5A9 0595 0000 FFEB 0001 0101



0