МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО 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 №3483

##### ***Выполнил:*** Студент группы P3134 Баянов Равиль Динарович

#### Преподаватель:

##### Бострикова Дарья

##### Константиновна

Санкт-Петербург, 2023

**Оглавление**

[**Задание** 3](#_Toc132752526)

[**Текст исходной программы** 4](#_Toc132752527)

[**Описание программы** 5](#_Toc132752528)

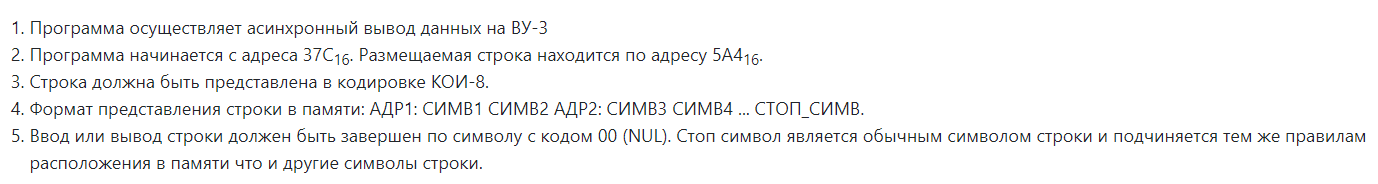
[**Трассировка** 7](#_Toc132752564)

[**Дополнительное задание** 9](#_Toc132752565)

[**Вывод** 11](#_Toc132752566)

# **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.



# **Текст исходной программы**

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Комментарии** |
| 37C |  | C: WORD 0x05A4 | Текущий элемент строки |
| 37D |  | STOP: WORD 0x0000 | Стоп-слово (конец строки) |
| 37E |  | L: WORD 0x0000 | Младший байт текущего элемента |
| 37F |  | B: WORD 0x0000 | Старший байт текущего элемента |
| 380 |  | M: WORD 0x00FF | Слово для выделения младшего байта |
| 381 | +0200 | START: CLA | Обнулить аккумулятор AC |
| 382 | AAF9 | LD (C)+ | Записать значение ячеек 5A4, 5A5… в аккумулятор AC |
| 383 | EEFA | ST L | Загрузка в ячейку 37E значение аккумулятора AC (младшего байта L) |
| 384 | 0680 | SWAB | Обмен старшего и младшего байта |
| 385 | EEF9 | ST B | Загрузка в ячейку 37F значение аккумулятора AC (старшего байта B) |
| 386 | 1207 | S1: IN #7 | Ввод регистра состояния в аккумулятор AC |
| 387 | 2F40 | AND #0x0040 | Побитовое И регистра состояния и слова для выявления готовности |
| 388 | F0FD | BEQ S1 | Переход, если Z==1 на ячейку 386 (S1) |
| 389 | AEF5 | LD B | Загрузка старшего байта (B) в аккумулятор AC |
| 38A | 2EF5 | AND M | Побитовое И старшего байта и слова для выделения (чтобы выделить старший байт от младшего) |
| 38B | 7EF1 | CMP STOP | Установка регистров состояния (проверка не является ли символ стоп-словом) |
| 38C | F00A | BEQ F | Переход, если Z==1 на ячейку 397 (F) |
| 38D | 1306 | OUT #6 | Вывод старшего байта (B) |
| 38E | 1207 | S2: IN #7 | Ввод регистра состояния в аккумулятор AC |
| 38F | 2F40 | AND #0x0040 | Побитовое И регистра состояния и слова для выявления готовности |
| 390 | F0FD | BEQ S2 | Переход, если Z==1 на ячейку 38E (S2) |
| 391 | AEE6 | LD L | Загрузка младшего байта (L) в аккумулятор AC |
| 392 | 2EE7 | AND M | Побитовое И младшего байта и слова для выделения (чтобы выделить младший байт от старшего) |
| 393 | 7EDD | CMP STOP | Установка регистров состояния (проверка не является ли символ стоп-словом) |
| 394 | F002 | BEQ F | Переход, если Z==1 на ячейку 397 (F) |
| 395 | 1306 | OUT #6 | Вывод младшего байта (L) |
| 396 | CEDE | JUMP START | Безусловный переход в ячейку 381 (START) |
| 397 | 0100 | F: HLT | Остановка программы |

# **Описание программы**

# Программа выводит символы на ВУ-3, представленные в памяти в формате:

*АДР1: СИМВ1 СИМВ2 АДР2: СИМВ3 СИМВ4 ... СТОП\_СИМВ.*

# **Ложка**

# КОИ-8 – EC CF D6 CB C1

***Расположение в памяти БЭВМ программы, исходных данных и результатов:***

381 - 397 – программа.

37С (С) - адрес первого символа строки.

37В (STOP) – адрес стоп-слова.

37E (L) – младший байт переменной (второй символ в одном адресе).

37F (B) – старший байт переменной (первый символ в одном адресе).

380 (M) – значение для выделения младшего байта.

5A4 – (5A4 + L16 – 1) – выводимая строка, где L16 - длина выводимой строки.

#### Область представления:

#### 37С – 37F – символ строки КОИ-8.

#### 380 – набор 16 логических значений.

#### 5A4 – (5A4 + L16 – 1) – два символа из строки в КОИ-8.

#### 37C – 11-ти разрядное беззнаковое число.

#### Область допустимых значений:

#### Все ячейки могут принимать любые значения, кроме 37C.

#### Так как программа начинается на ячейке 37C и заканчивается на ячейке 37F, то ОДЗ для 37С (С):

#### Строка лежит в [000, 37C] или [397, 7FF]

# **Код на ассемблере:**

# ORG 0x05A4

# WORD 0xECCF, 0xD6CB, 0xC100

# ORG 0x37C

# C: WORD 0x5A4

# STOP: WORD 0x0

# L: WORD 0x0

# B: WORD 0x0

# M: WORD 0xFF

# START: CLA

# LD (C)+

# ST L

# SWAB

# ST B

# S1: IN 0x7

# AND #0x40

# BEQ S1

# LD B

# AND M

# CMP STOP

# BEQ F

# OUT 0x6

# S2: IN 0x7

# AND #0x40

# BEQ S2

# LD L

# AND M

# CMP STOP

# BEQ F

# OUT 0x6

# JUMP START

# F: HLT

# **Трассировка**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Выполненная**  **Команда** | |  | | **Содержание аккумуляторов процессора после выполнение команды** | | | | | | | | **Ячейка, содержимое которой изменилось после выполнения команды** | |
| Адр | Знчн | IP | CR | | AR | DR | SP | BR | AC | PS | NZVC | Адр | Знчн |
| Адр | Знчн | IP | CR | | AR | DR | SP | BR | AC | PS | NZVC | Адр | Знчн |
| 37C | 05A4 | 37C | 0000 | | 000 | 0000 | 000 | 0000 | 0000 | 004 | 0100 |  |  |
| 37C | 05A4 | 37D | 05A4 | | 37C | 05A4 | 000 | 037C | 0000 | 004 | 0100 |  |  |
| 37D | 0000 | 37E | 0000 | | 37D | 0000 | 000 | 037D | 0000 | 004 | 0100 |  |  |
| 37E | 0000 | 37F | 0000 | | 37E | 0000 | 000 | 037E | 0000 | 004 | 0100 |  |  |
| 37F | 0000 | 380 | 0000 | | 37F | 0000 | 000 | 037F | 0000 | 004 | 0100 |  |  |
| 380 | 00FF | 381 | 00FF | | 380 | 00FF | 000 | 0380 | 0000 | 004 | 0100 |  |  |
| 381 | 0200 | 382 | 0200 | | 381 | 0200 | 000 | 0381 | 0000 | 004 | 0100 |  |  |
| 382 | AAF9 | 383 | AAF9 | | 5A4 | ECCF | 000 | FFF9 | ECCF | 008 | 1000 | 37C | 05A5 |
| 383 | EEFA | 384 | EEFA | | 37E | ECCF | 000 | FFFA | ECCF | 008 | 1000 | 37E | ECCF |
| 384 | 0680 | 385 | 0680 | | 384 | 0680 | 000 | 0384 | CFEC | 008 | 1000 |  |  |
| 385 | EEF9 | 386 | EEF9 | | 37F | CFEC | 000 | FFF9 | CFEC | 008 | 1000 | 37F | CFEC |
| 386 | 1207 | 387 | 1207 | | 386 | 1207 | 000 | 0386 | CF00 | 008 | 1000 |  |  |
| 387 | 2F40 | 388 | 2F40 | | 387 | 0040 | 000 | 0040 | 0000 | 004 | 0100 |  |  |
| 388 | F0FD | 386 | F0FD | | 388 | F0FD | 000 | FFFD | 0000 | 004 | 0100 |  |  |
| 386 | 1207 | 387 | 1207 | | 386 | 1207 | 000 | 0386 | 0000 | 004 | 0100 |  |  |
| 387 | 2F40 | 388 | 2F40 | | 387 | 0040 | 000 | 0040 | 0000 | 004 | 0100 |  |  |
| 388 | F0FD | 386 | F0FD | | 388 | F0FD | 000 | FFFD | 0000 | 004 | 0100 |  |  |
| 386 | 1207 | 387 | 1207 | | 386 | 1207 | 000 | 0386 | 0000 | 004 | 0100 |  |  |
| 387 | 2F40 | 388 | 2F40 | | 387 | 0040 | 000 | 0040 | 0000 | 004 | 0100 |  |  |
| 388 | F0FD | 386 | F0FD | | 388 | F0FD | 000 | FFFD | 0000 | 004 | 0100 |  |  |
| 386 | 1207 | 387 | 1207 | | 386 | 1207 | 000 | 0386 | 0000 | 004 | 0100 |  |  |
| 387 | 2F40 | 388 | 2F40 | | 387 | 0040 | 000 | 0040 | 0000 | 004 | 0100 |  |  |
| 388 | F0FD | 386 | F0FD | | 388 | F0FD | 000 | FFFD | 0000 | 004 | 0100 |  |  |
| 386 | 1207 | 387 | 1207 | | 386 | 1207 | 000 | 0386 | 0040 | 004 | 0100 |  |  |
| 387 | 2F40 | 388 | 2F40 | | 387 | 0040 | 000 | 0040 | 0040 | 000 | 0000 |  |  |
| 388 | F0FD | 389 | F0FD | | 388 | F0FD | 000 | 0388 | 0040 | 000 | 0000 |  |  |
| 389 | AEF5 | 38A | AEF5 | | 37F | CFEC | 000 | FFF5 | CFEC | 008 | 1000 |  |  |
| 38A | 2EF5 | 38B | 2EF5 | | 380 | 00FF | 000 | FFF5 | 00EC | 000 | 0000 |  |  |
| 38B | 7EF1 | 38C | 7EF1 | | 37D | 0000 | 000 | FFF1 | 00EC | 001 | 0001 |  |  |
| 38C | F00A | 38D | F00A | | 38C | F00A | 000 | 038C | 00EC | 001 | 0001 |  |  |
| 38D | 1306 | 38E | 1306 | | 38D | 1306 | 000 | 038D | 00EC | 001 | 0001 |  |  |
| 38E | 1207 | 38F | 1207 | | 38E | 1207 | 000 | 038E | 0000 | 001 | 0001 |  |  |
| 38F | 2F40 | 390 | 2F40 | | 38F | 0040 | 000 | 0040 | 0000 | 005 | 0101 |  |  |
| 390 | F0FD | 38E | F0FD | | 390 | F0FD | 000 | FFFD | 0000 | 005 | 0101 |  |  |
| 38E | 1207 | 38F | 1207 | | 38E | 1207 | 000 | 038E | 0000 | 005 | 0101 |  |  |
| 38F | 2F40 | 390 | 2F40 | | 38F | 0040 | 000 | 0040 | 0000 | 005 | 0101 |  |  |
| 390 | F0FD | 38E | F0FD | | 390 | F0FD | 000 | FFFD | 0000 | 005 | 0101 |  |  |
| 38E | 1207 | 38F | 1207 | | 38E | 1207 | 000 | 038E | 0000 | 005 | 0101 |  |  |
| 38F | 2F40 | 390 | 2F40 | | 38F | 0040 | 000 | 0040 | 0000 | 005 | 0101 |  |  |
| 390 | F0FD | 38E | F0FD | | 390 | F0FD | 000 | FFFD | 0000 | 005 | 0101 |  |  |
| 38E | 1207 | 38F | 1207 | | 38E | 1207 | 000 | 038E | 0000 | 005 | 0101 |  |  |
| 38F | 2F40 | 390 | 2F40 | | 38F | 0040 | 000 | 0040 | 0000 | 005 | 0101 |  |  |
| 390 | F0FD | 38E | F0FD | | 390 | F0FD | 000 | FFFD | 0000 | 005 | 0101 |  |  |
| 38E | 1207 | 38F | 1207 | | 38E | 1207 | 000 | 038E | 0000 | 005 | 0101 |  |  |
| 38F | 2F40 | 390 | 2F40 | | 38F | 0040 | 000 | 0040 | 0000 | 005 | 0101 |  |  |
| 390 | F0FD | 38E | F0FD | | 390 | F0FD | 000 | FFFD | 0000 | 005 | 0101 |  |  |
| 38E | 1207 | 38F | 1207 | | 38E | 1207 | 000 | 038E | 0000 | 005 | 0101 |  |  |
| 38F | 2F40 | 390 | 2F40 | | 38F | 0040 | 000 | 0040 | 0000 | 005 | 0101 |  |  |
| 390 | F0FD | 38E | F0FD | | 390 | F0FD | 000 | FFFD | 0000 | 005 | 0101 |  |  |
| 38E | 1207 | 38F | 1207 | | 38E | 1207 | 000 | 038E | 0040 | 005 | 0101 |  |  |
| 38F | 2F40 | 390 | 2F40 | | 38F | 0040 | 000 | 0040 | 0040 | 001 | 0001 |  |  |
| 390 | F0FD | 391 | F0FD | | 390 | F0FD | 000 | 0390 | 0040 | 001 | 0001 |  |  |
| 391 | AEEC | 392 | AEEC | | 37E | ECCF | 000 | FFEC | ECCF | 009 | 1001 |  |  |
| 392 | 2EED | 393 | 2EED | | 380 | 00FF | 000 | FFED | 00CF | 001 | 0001 |  |  |
| 393 | 7EE9 | 394 | 7EE9 | | 37D | 0000 | 000 | FFE9 | 00CF | 001 | 0001 |  |  |
| 394 | F002 | 395 | F002 | | 394 | F002 | 000 | 0394 | 00CF | 001 | 0001 |  |  |
| 395 | 1306 | 396 | 1306 | | 395 | 1306 | 000 | 0395 | 00CF | 001 | 0001 |  |  |
| 396 | CEEA | 381 | CEEA | | 396 | 0381 | 000 | FFEA | 00CF | 001 | 0001 |  |  |

# **Дополнительное задание**

Написать программу на ассемблере, которая вводит с клавиатуры (ВУ-8) символы в кодировке КОИ-8 и затем по нажатию кнопки Tab запускает вывод этих символов транслитом на текстовый принтер (ВУ-5).

**Описание программы:**

|  |  |
| --- | --- |
| **Код команд** | **Комментарии** |
| ORG 0x100 | Размещение программы в памяти с ячейки 100 |
| CURRENT1: WORD 0x10 | Текущий символ при вводе |
| CURRENT2: WORD 0x10 | Текущий символ при выводе |
| TAB: WORD 0x9 | Символ “Tab”, который запустит вывод символов |
| START: CLA | Начало программы (обнуление аккумулятора) |
| S1: IN 0x19 | Ожидание ввода символа |
| AND #0x40 | Проверка SR на наличие 6 бита |
| BEQ S1 | Нет – “спин-луп” |
| IN 0x18 | Ввод символа в аккумулятор AC |
| CMP TAB | Сравнение введённого символа с символом “Tab” |
| BEQ OUTPUT | Если введённый символ равен символу “Tab”, то переход к выводу символов |
| ST (CURRENT1)+ | Сохраняем символы в ячейки 10, 11, 12…(CURRENT1) |
| JUMP START | Безусловный переход к началу программы, для ввода следующего символа |
| OUTPUT: CLA | Начало вывода символов (обнуление аккумулятора) |
| S2: IN 0xD | Ожидание готовности ВУ-5 (готовности вывода символа) |
| AND #0x40 | Проверка SR на наличие 6 бита |
| BEQ S2 | Нет – “спин-луп” |
| LD (CURRENT2)+ | Загрузка символов в ячейках 10, 11, 12… в аккумулятор AC |
| AND #0x7F | Сброс 8 бита, чтобы вывести символы транслитом на КОИ-8 |
| CMP 0x0 | Сравнение загруженного в аккумулятор AC символа с символом 0, для того чтобы определить коне строки и выйти из программы |
| BEQ FINISH | Если всё-так символ равен символу 0, то выход из программы (FINISH) |
| OUT 0xC | Вывод символа на ВУ-5 |
| JUMP OUTPUT | Безусловный переход к началу вывода символа, чтобы вывести следующий символ в строке |
| FINISH: HLT | Завершение программы |

**Код на ассемблере:**

ORG 0x100

CURRENT1: WORD 0x10

CURRENT2: WORD 0x10

TAB: WORD 0x9

START: CLA

S1: IN 0x19

AND #0x40

BEQ S1

IN 0x18

CMP TAB

BEQ OUTPUT

ST (CURRENT1)+

JUMP START

OUTPUT: CLA

S2: IN 0xD

AND #0x40

BEQ S2

LD (CURRENT2)+

AND #0x7F

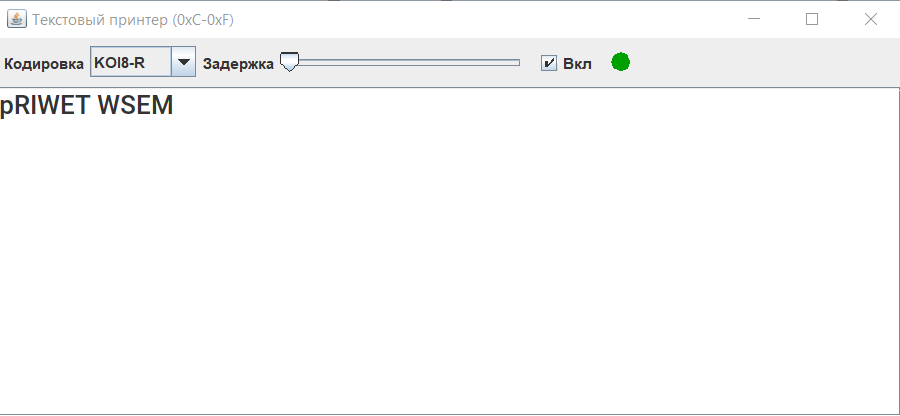
CMP 0x0

BEQ FINISH

OUT 0xC

JUMP OUTPUT

FINISH: HLT

**Вывод программы при вводе строки “Привет Всем”:**

# **Вывод**

Изучил организации системы ввода-вывода базовой ЭВМ, команды ввода-вывода (IN, OUT) и исследовал процесс функционирования ЭВМ при обмене данными по сигналам готовности внешних устройств (ВУ).