Санкт-Петербургский национальный исследовательский университет

информационных технологий, механики и оптики,

факультет программной инженерии и компьютерной техники

# Лабораторная работа по основам

# профессиональной деятельности №5

# «Асинхронный обмен с данными в ВУ»

Группа: Р3133

Выполнила: Агадилова Малика

Преподаватель: Лабушев Тимофей Михайлович

Вариант: 3305

Санкт-Петербург

2022

Вариант 3305

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 вывод данных на ВУ-1
2. Программа начинается с адреса (1A8)16. Размещаемая строка находится по адресу (64E)16.
3. Строка должна быть представлена в кодировке ISO-8859-5.
4. Формат представления строки в памяти: АДР1: СИМВ1 СИМВ2 АДР2: СИМВ3 СИМВ4 ... СТОП\_СИМВ.
5. Ввод или вывод строки должен быть завершен по символу c кодом 0A (NL). Стоп символ является обычным символом строки и подчиняется тем же правилам расположения в памяти что и другие символы строки.

# 

# 

# 

# 

# 

# 

# 

# 

# 

# Описание программы

| Адрес | Код команды | Мнемоника | Комментарии |
| --- | --- | --- | --- |
| 1A6 | 064E |  | Адрес строки |
| 1A7 | 000A |  | Стоп символ |
|  |  |  |  |
| 1A8 | 0200 | CLA | Очистка аккумулятора |
| 1A9 | A1A6 | LD 1A6 | Загрузка в аккумулятор строки для проверки и вывода первой буквы |
| 1AA | 0С00 | PUSH | Помещаем символы в стек |
| 1AB | D200 | CALL 200 | Вызов подпрограммы |
| 1AС | 0800 | POP | Проверка равна ли ячейка стоп-символу |
| 1AD | 71A7 | CMP 1A7 |
| 1AE | F008 | BEQ (IP + 8) |
| 1AF | AAF6 | LD (IP-A)+ | Загружаем строку в аккумулятор, чтобы проверить и вывести второй символ, содержащийся в ней |
| 1B0 | 0680 | SWAB | Обмен старшего и младшего байтов  Помещаем символ, который идет следующим по порядку в слове, в младший байт |
| 1B1 | 0C00 | PUSH | Помещаем неизмененную строку в стек |
| 1B2 | D200 | CALL 200 | Вызов подпрограммы |
| 1B3 | 0800 | POP | Проверка равна ли ячейка стоп-символу |
| 1B4 | 71A7 | CMP 1A7 |
| 1B5 | F001 | BEQ (IP + 1) |
| 1B6 | C1A8 | JUMP 1A8 | Переход к следующему действию для проверки следующей ячейки, содержащей символы |
| 1B7 | 0100 | HLT | Отключение ТГ, переход в пультовый режим |

Подпрограмма, отвечающая за вывод символов ВУ-1

| 200 | 1203 | IN 3 | Проверка готовности внешнего устройства |
| --- | --- | --- | --- |
| 201 | 2F40 | AND #0x40 | Если внешнее устройство готово (т.е. результат логического И числа 0x40 с аккумулятором не равен нулю), то продолжение выполнения подпрограммы, если нет – возврат в начало и новая проверка готовности |
| 202 | F0FD | BEQ (IP-3) |
| 203 | AC01 | LD &1 | Загрузка в аккумулятор текущих символов |
| 204 | 0600 | SXTB | Расширение знака младшего байта |
| 205 | 1302 | OUT 2 | Вывод во внешнее устройство кода символа |
| 206 | EC01 | ST &1 | Загрузка в стек измененной строки |
| 207 | 0A00 | RET | Возврат из подпрограммы |

Программа

ORG 0x1A6

FIRST:WORD 0x64E Начало строки

STOP:WORD 0x0A Cтоп-символ

START:CLA

NEXT:LD FIRST

PUSH

CALL 0x200

POP

CMP STOP

BEQ EXIT

LD (FIRST)+

SWAB

PUSH

CALL 0x200

POP

CMP STOP

BEQ EXIT

JUMP NEXT

EXIT: HLT

ORG 0x200

INPUT: IN 3

AND #0x40

BEQ INPUT

LD &1

SXTB

OUT 2

ST &1

RET

Вывод во внешнее устройство кодов символов заданного слова.

«byйт» ISO-8859-5: 62 79 D9 E2

UTF-8: D062 D079 D0B9 D182

UTF-16: 0062 0079 0439 0442

# 

# Область представления исходных данных:

FIRST, STOP → знаковые числа в 16-разрядном формате

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

1A8 – 1B7, 200 – 207 – команды

1A6 – 1A7 – исходные данные

*Адреса первой и последней выполняемых команд программы:*

Адрес первой выполняемой команды: 1A8

Адрес последней выполняемой команды: 1B7

# Трассировка

| **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 |  |
| 1A8 | 0200+ | 1A9 | 0200 | 1A8 | 0200 | 000 | 01A8 | 0000 | 0100 |  |  |  |
| 1A9 | A1A6 | 1AA | A1A6 | 1A6 | 064E | 000 | 01A9 | 7962 | 0000 |  |  |  |
| 1AA | 0С00 | 1AB | 0С00 | 7FF | 7962 | 7FF | 01AA | 7962 | 0000 | 7FF | 7962 |  |
| 1AB | D200 | 200 | D200 | 7FE | 1AC | 7FE | D200 | 7962 | 0000 | 7FE | 01AC |  |
| 200 | 1203 | 201 | 1203 | 200 | 1203 | 7FE | 0200 | 7962 | 0000 |  |  |  |
| 201 | 2F40 | 202 | 2F40 | 201 | 2F40 | 7FE | 0040 | 0040 | 0000 |  |  |  |
| 202 | F0FD | 203 | F0FD | 202 | F0FD | 7FE | 0202 | 0040 | 0000 |  |  |  |
| 203 | AC01 | 204 | AC01 | 7FF | 7962 | 7FE | 0001 | 7962 | 0000 |  |  |  |
| 204 | 0600 | 205 | 0600 | 204 | 0600 | 7FE | 0204 | 0062 | 0000 |  |  |  |
| 205 | 1302 | 206 | 1302 | 205 | 1302 | 7FE | 0205 | 0062 | 0000 |  |  |  |
| 206 | EC01 | 207 | EC01 | 206 | EC01 | 7FE | 0206 | 0062 | 0000 | 7FF | 0062 |  |
| 207 | 0A00 | 1AC | 0A00 | 207 | 01AC | 7FF | 0207 | 0062 | 0000 |  |  |  |
| 1AC | 0800 | 1AD | 0800 | 7FF | 0062 | 000 | 01AC | 0062 | 0000 |  |  |  |
| 1AD | 71A7 | 1AE | 71A7 | 1A7 | 000A | 000 | 01AD | 0058 | 0000 |  |  |  |
| 1AE | F008 | 1AF | F007 | 1AE | F007 | 000 | 01AE | 0058 | 0000 |  |  |  |
| 1AF | AAF6 | 1B0 | AAF6 | 1AF | 064E | 000 | FFF6 | 7962 | 0000 | 1A6 | 064F |  |
| 1B0 | 0680 | 1B1 | 0680 | 1B0 | 0680 | 000 | 01B0 | 6279 | 0000 |  |  |  |
| 1B1 | 0C00 | 1B2 | 0C00 | 7FF | 6279 | 7FF | 01B1 | 6279 | 0000 | 7FF | 6279 |  |
| 1B2 | D200 | 200 | D200 | 7FE | 1B3 | 7FE | D200 | 6279 | 0000 | 7FE | 01B3 |  |
| 200 | 1203 | 201 | 1203 | 200 | 1203 | 7FE | 0200 | 6279 | 0000 |  |  |  |
| 201 | 2F40 | 202 | 2F40 | 201 | 2F40 | 7FE | 0040 | 0040 | 0000 |  |  |  |
| 202 | F0FD | 203 | F0FD | 202 | F0FD | 7FE | 0202 | 0040 | 0000 |  |  |  |
| 203 | AC01 | 204 | AC01 | 7FF | 6279 | 7FE | 0001 | 6279 | 0000 |  |  |  |
| 204 | 0600 | 205 | 0600 | 204 | 0600 | 7FE | 0204 | 0079 | 0000 |  |  |  |
| 205 | 1302 | 206 | 1302 | 205 | 1302 | 7FE | 0205 | 0079 | 0000 |  |  |  |
| 206 | EC01 | 207 | EC01 | 206 | EC01 | 7FE | 0206 | 0079 | 0000 | 7FF | 0079 |  |
| 207 | 0A00 | 01B3 | 0A00 | 207 | 01AC | 7FF | 0207 | 0079 | 0000 |  |  |  |
| 1B3 | 0800 | 1B4 | 0800 | 7FF | 0079 | 000 | 01B3 | 0079 | 0000 |  |  |  |
| 1B4 | 71A7 | 1B5 | 71A7 | 1A7 | 000A | 000 | 01B4 | 006F | 0000 |  |  |  |
| 1B5 | F001 | 1B6 | F001 | 1B5 | F001 | 000 | 01B5 | 006F | 0000 |  |  |  |
| 1B6 | C1A8 | 1A8 | C1A8 | 1B6 | C1A8 | 000 | 01B6 | 006F | 0000 |  |  |  |
| 1A8 | 0200+ | 1A9 | 0200 | 1A8 | 0200 | 000 | 01A8 | 0000 | 0100 |  |  |  |
| 1A9 | A1A6 | 1AA | A1A6 | 1A6 | 064F | 000 | 01A9 | E2D9 | 1000 |  |  |  |
| 1AA | 0С00 | 1AB | 0С00 | 7FF | E2D9 | 7FF | 01AA | E2D9 | 1000 | 7FF | E2D9 |  |
| 1AB | D200 | 200 | D200 | 7FE | 1AC | 7FE | D200 | E2D9 | 1000 | 7FE | 01AC |  |
| 200 | 1203 | 201 | 1203 | 200 | 1203 | 7FE | 0200 | E2D9 | 1000 |  |  |  |
| 201 | 2F40 | 202 | 2F40 | 201 | 2F40 | 7FE | 0040 | 0040 | 1000 |  |  |  |
| 202 | F0FD | 203 | F0FD | 202 | F0FD | 7FE | 0202 | 0040 | 1000 |  |  |  |
| 203 | AC01 | 204 | AC01 | 7FF | 7962 | 7FE | 0001 | E2D9 | 1000 |  |  |  |
| 204 | 0600 | 205 | 0600 | 204 | 0600 | 7FE | 0204 | FFD9 | 1000 |  |  |  |
| 205 | 1302 | 206 | 1302 | 205 | 1302 | 7FE | 0205 | FFD9 | 1000 |  |  |  |
| 206 | EC01 | 207 | EC01 | 206 | EC01 | 7FE | 0206 | FFD9 | 1000 | 7FF | FFD9 |  |
| 207 | 0A00 | 1AC | 0A00 | 207 | 01AC | 7FF | 0207 | FFD9 | 1000 |  |  |  |
| 1AC | 0800 | 1AD | 0800 | 7FF | FFD9 | 000 | 01AC | FFD9 | 1000 |  |  |  |
| 1AD | 71A7 | 1AE | 71A7 | 1A7 | 000A | 000 | 01AD | FFCF | 1000 |  |  |  |
| 1AE | F008 | 1AF | F007 | 1AE | F007 | 000 | 01AE | FFCF | 1000 |  |  |  |
| 1AF | AAF6 | 1B0 | AAF6 | 1AF | 064F | 000 | FFF6 | E2D9 | 1000 | 1A6 | 0650 |  |
| 1B0 | 0680 | 1B1 | 0680 | 1B0 | 0680 | 000 | 01B0 | D9E2 | 1000 |  |  |  |
| 1B1 | 0C00 | 1B2 | 0C00 | 7FF | D9E2 | 7FF | 01B1 | D9E2 | 1000 | 7FF | D9E2 |  |
| 1B2 | D200 | 200 | D200 | 7FE | 1B3 | 7FE | D200 | D9E2 | 1000 | 7FE | 01B3 |  |
| 200 | 1203 | 201 | 1203 | 200 | 1203 | 7FE | 0200 | D9E2 | 1000 |  |  |  |
| 201 | 2F40 | 202 | 2F40 | 201 | 2F40 | 7FE | 0040 | 0040 | 0000 |  |  |  |
| 202 | F0FD | 203 | F0FD | 202 | F0FD | 7FE | 0202 | 0040 | 0000 |  |  |  |
| 203 | AC01 | 204 | AC01 | 7FF | D9E2 | 7FE | 0001 | D9E2 | 1000 |  |  |  |
| 204 | 0600 | 205 | 0600 | 204 | 0600 | 7FE | 0204 | FFE2 | 1000 |  |  |  |
| 205 | 1302 | 206 | 1302 | 205 | 1302 | 7FE | 0205 | FFE2 | 1000 |  |  |  |
| 206 | EC01 | 207 | EC01 | 206 | EC01 | 7FE | 0206 | FFE2 | 1000 | 7FF | FFE2 |  |
| 207 | 0A00 | 01B3 | 0A00 | 207 | 01AC | 7FF | 0207 | FFE2 | 1000 |  |  |  |
| 1B3 | 0800 | 1B4 | 0800 | 7FF | FFE2 | 000 | 01B3 | FFE2 | 1000 |  |  |  |
| 1B4 | 71A7 | 1B5 | 71A7 | 1A7 | 000A | 000 | 01B4 | FFD8 | 1000 |  |  |  |
| 1B5 | F001 | 1B6 | F001 | 1B5 | F001 | 000 | 01B5 | FFD8 | 1000 |  |  |  |
| 1B6 | C1A8 | 1A8 | C1A8 | 1B6 | C1A8 | 000 | 01B6 | FFD8 | 1000 |  |  |  |
| 1A8 | 0200+ | 1A9 | 0200 | 1A8 | 0200 | 000 | 01A8 | 0000 | 0100 |  |  |  |
| 1A9 | A1A6 | 1AA | A1A6 | 1A6 | 0650 | 000 | 01A9 | 000A | 0000 |  |  |  |
| 1AA | 0С00 | 1AB | 0С00 | 7FF | 000A | 7FF | 01AA | 000A | 0000 | 7FF | 000A |  |
| 1AB | D200 | 200 | D200 | 7FE | 1AC | 7FE | D200 | 000A | 0000 | 7FE | 01AC |  |
| 200 | 1203 | 201 | 1203 | 200 | 1203 | 7FE | 0200 | 000A | 0000 |  |  |  |
| 201 | 2F40 | 202 | 2F40 | 201 | 2F40 | 7FE | 0040 | 0040 | 0000 |  |  |  |
| 202 | F0FD | 203 | F0FD | 202 | F0FD | 7FE | 0202 | 0040 | 0000 |  |  |  |
| 203 | AC01 | 204 | AC01 | 7FF | 7962 | 7FE | 0001 | 000A | 0000 |  |  |  |
| 204 | 0600 | 205 | 0600 | 204 | 0600 | 7FE | 0204 | 000A | 0000 |  |  |  |
| 205 | 1302 | 206 | 1302 | 205 | 1302 | 7FE | 0205 | 000A | 0000 |  |  |  |
| 206 | EC01 | 207 | EC01 | 206 | EC01 | 7FE | 0206 | 000A | 0000 | 7FF | 000A |  |
| 207 | 0A00 | 1AC | 0A00 | 207 | 01AC | 7FF | 0207 | 000A | 0000 |  |  |  |
| 1AC | 0800 | 1AD | 0800 | 7FF | 000A | 000 | 01AC | 000A | 0000 |  |  |  |
| 1AD | 71A7 | 1AE | 71A7 | 1A7 | 000A | 000 | 01AD | 0000 | 0100 |  |  |  |
| 1AE | F008 | 1B7 | F007 | 1AE | F007 | 000 | 01AE | 0000 | 0100 |  |  |  |
| 1B7 | 0100 | 1B8 | 0100 | 1B7 | 0100 | 000 | 01B7 | 0000 | 0100 |  |  |  |

# Дополнительное задание

1. Добрый вечер! В качестве дополнительного задания предлагаю вам написать на ассемблере БЭВМ программу для подсчета количества символов в строке в кодировке UTF-8 (обратите внимание на то, что число символов в строке может отличаться от числа байт в ней). На вход программе подается адрес строки в памяти. В каждую ячейку начиная с указанного адреса записывается по два байта строки, конец строки определяется по стоп-символу с кодом 00. Если строка не соответствует UTF-8 — например, в первом байте символа значится, что в символе N байт, но последующие N-1 байт не начинаются с 10 — то число символов выставляется в -1, работа программы прекращается.

<http://samag.ru/archive/article/2653>

ORG 0x333

FIRST:WORD 0x64E Начало строки

COUNT: WORD 0x0000 счетчик слов

CHECK: WORD 0X0009 переход к 2 байту

WTF: WORD 0xFFFF

START: CLA

NEXT: LD (FIRST)+

ROL

BCS UWU c=1 проверяется 15 бит

BEQ STOP

LOADCOUNT: LD COUNT

INC

ST COUNT подсчет символов

JUMP NEXT

UWU: ROL

LOOP CHECK переход к 2 байту

JUMP UWU

BCS CHECK1

JUMP STOP

CHECK1: ROL

BCC LOADCOUNT с=0 при не свопадениии 110\*\*\* 10\*\*\*\*

LD WTF

ST COUNT

STOP: HLT

Вывод

В процессе выполнения данной лабораторной работы был изучен асинхронный обмен данными с ВУ-1.