

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ
НАПРАВЛЕНИЕ СИСТЕМНОГО И ПРИКЛАДНОГО ПРОГРАММНОГО
ОБЕСПЕЧЕНИЯ

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5
курса «Основы профессиональной деятельности»
по теме: «Исследование работы БЭВМ»
Вариант № 1035

Выполнил студент:
Тюрин Иван Николаевич
группа: Р3110

Преподаватель:
Клименков С. В.,
Ларочкин Г. И.

Санкт-Петербург, 2022 г.

Содержание

| | |
|--|----------|
| Лабораторная работа № 5. Исследование работы БЭВМ | 2 |
| 1. Задание варианта № 1035 | 2 |
| 2. Описание программы | 3 |
| 1. Назначение программы | 3 |
| 2. Область представления и допустимых значений | 3 |
| 3. Трассировка программы | 3 |
| 3. Программа | 3 |
| 4. Вывод | 3 |

Лабораторная работа № 5

Исследование работы БЭВМ

1. Задание варианта № 1035

, , ,

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-3
- Программа начинается с адреса $5D2_{16}$. Размещаемая строка находится по адресу 562_{16} .
- Строка должна быть представлена в кодировке КОИ-8.
- Формат представления строки в памяти: АДР1: СИМВ2 СИМВ1 АДР2: СИМВ4 СИМВ3 ... СТОП_СИМВ.
- Ввод или вывод строки должен быть завершен по символу с кодом $0x0A$ (NL). Стоп символ является обычным символом строки и подчиняется тем же правилам расположения в памяти что и другие символы строки.

, , ,

2. Описание программы

2. 1. Назначение программы

Описание программы представлено в таблице 1.1. [??](#). Программа осуществляет асинхронный ввод кодов символов (кодировка *KOI8-R*) через ВУ-3, и после ввода стоп-символа 0x0a=NL прекращает считывание, начинает асинхронный вывод символов соответствующих введенным кодам через ВУ-5 (текстовый принтер).

2. 2. Область представления и допустимых значений

Пользователю доступны для ввода любые однобайтовые беззнаковые числа, то есть из диапазона $[0; 255]$, соответствующие символам в кодировке *KOI8-R*, любое из них будет выведено при помощи принтера. Однако после символа с номером 10_{10} , то есть 0x0a, ввод прекращается. Так же при вводе символа с кодом 0x00 произойдет очистка вывода принтера.

Количество введенных символов не должно превышать $112_{10} = 0x5D2 - 0x562$ (с учетом стоп-символа). Это ограничение вызвано расположением программы и символов строки в памяти указанным в задании.

2. 3. Трассировка программы

Трассировка программы для введенных символов *Z*, *V*, коды которых соответственно 0x5a и 0x56, представлена в таблице 1.4.[??](#) и 1.6.[??](#).

Были введены коды символов *Z*, *V* и код стоп-символа \n эти же символы были напечатаны на принтере.

3. Программа

Была составлена программа на языке ассемблера. Она представлена в листингах [1.2](#) и [1.4](#).

4. Вывод

Научился красиво называть метки в ассемблере БЭВМ. Научился писать программы с асинхронным вводом-выводом при помощи цикла. Научился работать с УВ текстовым принтером (поработал с хотя бы каким-то принтером).

| Адрес | Метка | Мнемоника | Описание |
|-------|-------------|-----------------|---|
| 0x5d2 | endchar: | word 0x0a | Код стоп символа |
| 0x5d3 | mask: | word 0x00ff | Маска для отделения младшего байта слова |
| 0x5d4 | str: | word 0x562 | Адрес начала строки |
| 0x5d5 | str.len: | word 0x0 | Длина строки |
| 0x5d6 | str.iter: | word 0x0 | Переменная итерирования по строке |
| 0x5d7 | char: | word 0x0 | Переменная для кода полученного символа |
| 0x5d8 | START: | ld \$str | Начало программы |
| 0x5d9 | | st \$str.iter | Итератор по строке устанавливается в начало |
| 0x5da | _readstr: | cla | Цикл ввода строки |
| 0x5db | | call _readchar | Вызов подпрограммы считывания символа |
| 0x5dc | | st \$char | |
| 0x5dd | | ld \$str.len | Увеличение длины строки на 1 |
| 0x5de | | inc | |
| 0x5df | | st \$str.len | |
| 0x5e0 | | ror | |
| 0x5e1 | | bcc _oddpos | Обработка позиции символа в троке |
| 0x5e2 | | ld \$char | |
| 0x5e3 | | st (str.iter) | |
| 0x5e4 | | jump _is_end | |
| 0x5e5 | | ld \$char | Изменение положения кода символа в слове для четных позиций |
| 0x5e6 | | swab | |
| 0x5e7 | | add (str.iter) | |
| 0x5e8 | | st (str.iter)+ | Сдвиг итератора на позицию следующей пары символов |
| 0x5e9 | | ld \$char | Проверка на стоп-символ |
| 0x5ea | _is_end: | cmp endchar | |
| 0x5eb | | bne _readstr | |
| 0x5ec | | ld \$str | Итератор устанавливается на начало строки |
| 0x5ed | | st \$str.iter | |
| 0x5ee | _printstr: | ld (str.iter) | Цикл вывода строки |
| 0x5ef | | call _writechar | Вызов подпрограммы для вывода первого символа в слове |
| 0x5f0 | | ld (str.iter) | |
| 0x5f1 | | and mask | |
| 0x5f2 | | cmp endchar | Проверка на стоп-символ |
| 0x5f3 | | beq __stop | |
| 0x5f4 | | ld (str.iter) | |
| 0x5f5 | | swab | |
| 0x5f6 | | call _writechar | Вызов подпрограммы для вывода второго символа в слове |
| 0x5f7 | | ld (str.iter)+ | |
| 0x5f8 | | swab | |
| 0x5f9 | | and mask | |
| 0x5fa | | cmp endchar | |
| 0x5fb | | bne _printstr | |
| 0x5fc | __stop: | hlt | Останов |
| 0x5fd | _readchar: | in 7 | Подпрограмма для ввода символа |
| 0x5fe | | and #0x40 | Цикл ожидания ввода |
| 0x5ff | | beq _readchar | |
| 0x600 | | in 6 | Ввод кода символа |
| 0x601 | | and mask | |
| 0x602 | | ret | |
| 0x603 | _writechar: | out 0xc | Подпрограмма для вывода символа |
| 0x604 | _waitwrote: | in 0xe | Ожидание окончания вывода |
| 0x605 | | ror | |
| 0x606 | | bcs _waitwrote | |
| 0x607 | | ret | |

Таблица 1.2: Описание работы подпрограмм

| Выполняемая команда | | Содержимое регистров процессора после выполнения команды. | | | | | | | | | Ячейка, содержимое которой изменилось после выполнения команды | |
|---------------------|----------|---|------|-----|------|-----|------|------|-----|------|--|---------------|
| Адрес | Значение | IP | CR | AR | DR | SP | BR | AC | PS | NZVC | Адрес | Значение |
| 5D8 | A5D4 | 5D8 | 0000 | 000 | 0000 | 000 | 0000 | 0000 | 004 | 0100 | | |
| 5D8 | A5D4 | 5D9 | A5D4 | 5D4 | 0562 | 000 | 05D8 | 0562 | 000 | 0000 | | |
| 5D9 | E5D6 | 5DA | E5D6 | 5D6 | 0562 | 000 | 05D9 | 0562 | 000 | 0000 | 5D6 | 0562 |
| 5DA | 0200 | 5DB | 0200 | 5DA | 0200 | 000 | 05DA | 0000 | 004 | 0100 | | |
| 5DB | DE21 | 5FD | DE21 | 7FF | 05DC | 7FF | 05FD | 0000 | 004 | 0100 | 7FF | 05DC |
| 5FD | 1207 | 5FE | 1207 | 5FD | 1207 | 7FF | 05FD | 0040 | 004 | 0100 | | |
| 5FE | 2F40 | 5FF | 2F40 | 5FE | 0040 | 7FF | 0040 | 0040 | 000 | 0000 | | |
| 5FF | F0FD | 600 | F0FD | 5FF | F0FD | 7FF | 05FF | 0040 | 000 | 0000 | | |
| 600 | 1206 | 601 | 1206 | 600 | 1206 | 7FF | 0600 | 005A | 000 | 0000 | | |
| 601 | 2ED1 | 602 | 2ED1 | 5D3 | 00FF | 7FF | FFD1 | 005A | 000 | 0000 | | |
| 602 | 0A00 | 5DC | 0A00 | 7FF | 05DC | 000 | 0602 | 005A | 000 | 0000 | | |
| 5DC | E5D7 | 5DD | E5D7 | 5D7 | 005A | 000 | 05DC | 005A | 000 | 0000 | 5D7 | 005A |
| 5DD | A5D5 | 5DE | A5D5 | 5D5 | 0000 | 000 | 05DD | 0000 | 004 | 0100 | | |
| 5DE | 0700 | 5DF | 0700 | 5DE | 0700 | 000 | 05DE | 0001 | 000 | 0000 | | |
| 5DF | E5D5 | 5E0 | E5D5 | 5D5 | 0001 | 000 | 05DF | 0001 | 000 | 0000 | 5D5 | 0001 |
| 5E0 | 0480 | 5E1 | 0480 | 5E0 | 0480 | 000 | 05E0 | 0000 | 007 | 0111 | | |
| 5E1 | F503 | 5E2 | F503 | 5E1 | F503 | 000 | 05E1 | 0000 | 007 | 0111 | | |
| 5E2 | A5D7 | 5E3 | A5D7 | 5D7 | 005A | 000 | 05E2 | 005A | 001 | 0001 | | |
| 5E3 | E8F2 | 5E4 | E8F2 | 562 | 005A | 000 | FFF2 | 005A | 001 | 0001 | 562 | 005A |
| 5E4 | CE04 | 5E9 | CE04 | 5E4 | 05E9 | 000 | 0004 | 005A | 001 | 0001 | | |
| 5E9 | A5D7 | 5EA | A5D7 | 5D7 | 005A | 000 | 05E9 | 005A | 001 | 0001 | | |
| 5EA | 7EE7 | 5EB | 7EE7 | 5D2 | 000A | 000 | FFE7 | 005A | 001 | 0001 | | |
| 5EB | F1EE | 5DA | F1EE | 5EB | F1EE | 000 | FFEE | 005A | 001 | 0001 | | |
| 5DA | 0200 | 5DB | 0200 | 5DA | 0200 | 000 | 05DA | 0000 | 005 | 0101 | | |
| 5DB | DE21 | 5FD | DE21 | 7FF | 05DC | 7FF | 05FD | 0000 | 005 | 0101 | 7FF | 05DC |
| 5FD | 1207 | 5FE | 1207 | 5FD | 1207 | 7FF | 05FD | 0040 | 005 | 0101 | | |
| 5FE | 2F40 | 5FF | 2F40 | 5FE | 0040 | 7FF | 0040 | 0040 | 001 | 0001 | | |
| 5FF | F0FD | 600 | F0FD | 5FF | F0FD | 7FF | 05FF | 0040 | 001 | 0001 | | |
| 600 | 1206 | 601 | 1206 | 600 | 1206 | 7FF | 0600 | 0056 | 001 | 0001 | | |
| 601 | 2ED1 | 602 | 2ED1 | 5D3 | 00FF | 7FF | FFD1 | 0056 | 001 | 0001 | | |
| 602 | 0A00 | 5DC | 0A00 | 7FF | 05DC | 000 | 0602 | 0056 | 001 | 0001 | | |
| 5DC | E5D7 | 5DD | E5D7 | 5D7 | 0056 | 000 | 05DC | 0056 | 001 | 0001 | 5D7 | 0056 |
| 5DD | A5D5 | 5DE | A5D5 | 5D5 | 0001 | 000 | 05DD | 0001 | 001 | 0001 | | |
| 5DE | 0700 | 5DF | 0700 | 5DE | 0700 | 000 | 05DE | 0002 | 000 | 0000 | | |
| 5DF | E5D5 | 5E0 | E5D5 | 5D5 | 0002 | 000 | 05DF | 0002 | 000 | 0000 | 5D5 | 0002 |
| 5E0 | 0480 | 5E1 | 0480 | 5E0 | 0480 | 000 | 05E0 | 0001 | 000 | 0000 | | |
| 5E1 | F503 | 5E5 | F503 | 5E1 | F503 | 000 | 0003 | 0001 | 000 | 0000 | | |
| 5E5 | A5D7 | 5E6 | A5D7 | 5D7 | 0056 | 000 | 05E5 | 0056 | 000 | 0000 | | |
| 5E6 | 0680 | 5E7 | 0680 | 5E6 | 0680 | 000 | 05E6 | 5600 | 000 | 0000 | | |
| 5E7 | 48EE | 5E8 | 48EE | 562 | 005A | 000 | FFEE | 565A | 000 | 0000 | | |
| 5E8 | EAED | 5E9 | EAED | 562 | 565A | 000 | FFED | 565A | 000 | 0000 | 5D6 562 | 0563, 565A |
| 5E9 | A5D7 | 5EA | A5D7 | 5D7 | 0056 | 000 | 05E9 | 0056 | 000 | 0000 | | |
| 5EA | 7EE7 | 5EB | 7EE7 | 5D2 | 000A | 000 | FFE7 | 0056 | 001 | 0001 | | |
| 5EB | F1EE | 5DA | F1EE | 5EB | F1EE | 000 | FFEE | 0056 | 001 | 0001 | | |
| 5DA | 0200 | 5DB | 0200 | 5DA | 0200 | 000 | 05DA | 0000 | 005 | 0101 | | |
| 5DB | DE21 | 5FD | DE21 | 7FF | 05DC | 7FF | 05FD | 0000 | 005 | 0101 | 7FF | 05DC |
| 5FD | 1207 | 5FE | 1207 | 5FD | 1207 | 7FF | 05FD | 0040 | 005 | 0101 | | |
| 5FE | 2F40 | 5FF | 2F40 | 5FE | 0040 | 7FF | 0040 | 0040 | 001 | 0001 | | |
| 5FF | F0FD | 600 | F0FD | 5FF | F0FD | 7FF | 05FF | 0040 | 001 | 0001 | | |
| 600 | 1206 | 601 | 1206 | 600 | 1206 | 7FF | 0600 | 000A | 001 | 0001 | | |
| 601 | 2ED1 | 602 | 2ED1 | 5D3 | 00FF | 7FF | FFD1 | 000A | 001 | 0001 | | |
| 602 | 0A00 | 5DC | 0A00 | 7FF | 05DC | 000 | 0602 | 000A | 001 | 0001 | | |
| 5DC | E5D7 | 5DD | E5D7 | 5D7 | 000A | 000 | 05DC | 000A | 001 | 0001 | 5D7 | 000A |
| 5DD | A5D5 | 5DE | A5D5 | 5D5 | 0002 | 000 | 05DD | 0002 | 001 | 0001 | | |
| 5DE | 0700 | 5DF | 0700 | 5DE | 0700 | 000 | 05DE | 0003 | 000 | 0000 | | |
| 5DF | E5D5 | 5E0 | E5D5 | 5D5 | 0003 | 000 | 05DF | 0003 | 000 | 0000 | 5D5 | 0003 |
| 5E0 | 0480 | 5E1 | 0480 | 5E0 | 0480 | 000 | 05E0 | 0001 | 003 | 0011 | | |
| 5E1 | F503 | 5E2 | F503 | 5E1 | F503 | 000 | 05E1 | 0001 | 003 | 0011 | | |
| 5E2 | A5D7 | 5E3 | A5D7 | 5D7 | 000A | 000 | 05E2 | 000A | 001 | 0001 | | |
| 5E3 | E8F2 | 5E4 | E8F2 | 563 | 000A | 000 | FFF2 | 000A | 001 | 0001 | 563 | 000A |
| 5E4 | CE04 | 5E9 | CE04 | 5E4 | 05E9 | 000 | 0004 | 000A | 001 | 0001 | | |
| 5E9 | A5D7 | 5EA | A5D7 | 5D7 | 000A | 000 | 05E9 | 000A | 001 | 0001 | | |
| 5EA | 7EE7 | 5EB | 7EE7 | 5D2 | 000A | 000 | FFE7 | 000A | 005 | 0101 | | |
| 5EB | F1EE | 5EC | F1EE | 5EB | F1EE | 000 | 05EB | 000A | 005 | 0101 | | |
| 5EC | A5D4 | 5ED | A5D4 | 5D4 | 0562 | 000 | 05EC | 0562 | 001 | 0001 | | |
| 5ED | E5D6 | 5EE | E5D6 | 5D6 | 0562 | 000 | 05ED | 0562 | 001 | 0001 | 5D6 | 0562 |

Таблица 1.4: Трассировка программы с момента старта до момента начала цикла вывода

| Выполняемая команда | | Содержимое регистров процессора после выполнения команды. | | | | | | | | | Ячейка, содержимое которой изменилось после выполнения команды | |
|---------------------|----------|---|------|-----|------|-----|------|------|-----|------|--|----------|
| Адрес | Значение | IP | CR | AR | DR | SP | BR | AC | PS | NZVC | Адрес | Значение |
| 5EE | A8E7 | 5EF | A8E7 | 562 | 565A | 000 | FFE7 | 565A | 001 | 0001 | 7FF | 05F0 |
| 5EF | DE13 | 603 | DE13 | 7FF | 05F0 | 7FF | 0603 | 565A | 001 | 0001 | | |
| 603 | 130C | 604 | 130C | 603 | 130C | 7FF | 0603 | 565A | 001 | 0001 | | |
| 604 | 120E | 605 | 120E | 604 | 120E | 7FF | 0604 | 565A | 001 | 0001 | | |
| 605 | 0480 | 606 | 0480 | 605 | 0480 | 7FF | 0605 | AB2D | 00A | 1010 | | |
| 606 | F4FD | 607 | F4FD | 606 | F4FD | 7FF | 0606 | AB2D | 00A | 1010 | | |
| 607 | 0A00 | 5F0 | 0A00 | 7FF | 05F0 | 000 | 0607 | AB2D | 00A | 1010 | | |
| 5F0 | A8E5 | 5F1 | A8E5 | 562 | 565A | 000 | FFE5 | 565A | 000 | 0000 | | |
| 5F1 | 2EE1 | 5F2 | 2EE1 | 5D3 | 00FF | 000 | FFE1 | 005A | 000 | 0000 | | |
| 5F2 | 7EDF | 5F3 | 7EDF | 5D2 | 000A | 000 | FFDF | 005A | 001 | 0001 | | |
| 5F3 | F008 | 5F4 | F008 | 5F3 | F008 | 000 | 05F3 | 005A | 001 | 0001 | 7FF | 05F7 |
| 5F4 | A8E1 | 5F5 | A8E1 | 562 | 565A | 000 | FFE1 | 565A | 001 | 0001 | | |
| 5F5 | 0680 | 5F6 | 0680 | 5F5 | 0680 | 000 | 05F5 | 5A56 | 001 | 0001 | | |
| 5F6 | DE0C | 603 | DE0C | 7FF | 05F7 | 7FF | 0603 | 5A56 | 001 | 0001 | | |
| 603 | 130C | 604 | 130C | 603 | 130C | 7FF | 0603 | 5A56 | 001 | 0001 | | |
| 604 | 120E | 605 | 120E | 604 | 120E | 7FF | 0604 | 5A56 | 001 | 0001 | | |
| 605 | 0480 | 606 | 0480 | 605 | 0480 | 7FF | 0605 | AD2B | 00A | 1010 | | |
| 606 | F4FD | 607 | F4FD | 606 | F4FD | 7FF | 0606 | AD2B | 00A | 1010 | | |
| 607 | 0A00 | 5F7 | 0A00 | 7FF | 05F7 | 000 | 0607 | AD2B | 00A | 1010 | | |
| 5F7 | AADE | 5F8 | AADE | 562 | 565A | 000 | FFDE | 565A | 000 | 0000 | 5D6 | 0563 |
| 5F8 | 0680 | 5F9 | 0680 | 5F8 | 0680 | 000 | 05F8 | 5A56 | 000 | 0000 | | |
| 5F9 | 2ED9 | 5FA | 2ED9 | 5D3 | 00FF | 000 | FFD9 | 0056 | 000 | 0000 | | |
| 5FA | 7ED7 | 5FB | 7ED7 | 5D2 | 000A | 000 | FFD7 | 0056 | 001 | 0001 | | |
| 5FB | F1F2 | 5EE | F1F2 | 5FB | F1F2 | 000 | FFF2 | 0056 | 001 | 0001 | | |
| 5EE | A8E7 | 5EF | A8E7 | 563 | 000A | 000 | FFE7 | 000A | 001 | 0001 | | |
| 5EF | DE13 | 603 | DE13 | 7FF | 05F0 | 7FF | 0603 | 000A | 001 | 0001 | 7FF | 05F0 |
| 603 | 130C | 604 | 130C | 603 | 130C | 7FF | 0603 | 000A | 001 | 0001 | | |
| 604 | 120E | 605 | 120E | 604 | 120E | 7FF | 0604 | 000A | 001 | 0001 | | |
| 605 | 0480 | 606 | 0480 | 605 | 0480 | 7FF | 0605 | 8005 | 00A | 1010 | | |
| 606 | F4FD | 607 | F4FD | 606 | F4FD | 7FF | 0606 | 8005 | 00A | 1010 | | |
| 607 | 0A00 | 5F0 | 0A00 | 7FF | 05F0 | 000 | 0607 | 8005 | 00A | 1010 | | |
| 5F0 | A8E5 | 5F1 | A8E5 | 563 | 000A | 000 | FFE5 | 000A | 000 | 0000 | | |
| 5F1 | 2EE1 | 5F2 | 2EE1 | 5D3 | 00FF | 000 | FFE1 | 000A | 000 | 0000 | | |
| 5F2 | 7EDF | 5F3 | 7EDF | 5D2 | 000A | 000 | FFDF | 000A | 005 | 0101 | | |
| 5F3 | F008 | 5FC | F008 | 5F3 | F008 | 000 | 0008 | 000A | 005 | 0101 | | |
| 5FC | 0100 | 5FD | 0100 | 5FC | 0100 | 000 | 05FC | 000A | 005 | 0101 | | |

Таблица 1.6: Трассировка программы с момента начала цикла вывода до конца

```

1 org          0x5d2
2 endchar:     word    0x0a
3 mask:        word    0x00ff
4
5 str:         word    0x562
6 str.len:     word    0x0
7 str.iter:    word    0x0
8 char:        word    0x0
9
10 START:
11 ld  $str
12 st  $str.iter
13
14 _readstr:
15     cla
16     call    _readchar
17     st      $char
18     ld      $str.len
19     inc
20     st      $str.len
21     ror
22     bcc     _oddpos
23     ld      $char
24     st      (str.iter)
25     jump   _is_end
26     _oddpos:
27     ld      $char
28     swab
29     add     (str.iter)
30     st      (str.iter)+
31     _is_end:
32     ld      $char
33     cmp     endchar
34     bne     _readstr
35
36 ld      $str
37 st      $str.iter

```

Листинг 1.2: Код программы на языке ассемблена БЭВМ: параметры программы, переменные и цикл ввода символов


```

1  _printstr:
2      ld      (str.iter)
3      call   _writechar
4      ld      (str.iter)
5      and     mask
6      cmp     endchar
7      beq     __stop
8
9      ld      (str.iter)
10     swab
11     call   _writechar
12     ld      (str.iter)+
13     swab
14     and     mask
15     cmp     endchar
16     bne     _printstr
17 __stop:
18 hlt
19
20 _readchar:
21     in      7
22     and     #0x40
23     beq     _readchar
24     in      6
25     and     mask
26     ret
27
28 _writechar:
29     out     0xc
30     _waitwrote:
31     in      0xe
32     ror
33     bcs     _waitwrote
34     ret

```

Листинг 1.4: Код программы на языке ассемблена БЭВМ: цикл вывода символов и подпрограммы для асинхронного ввода и вывода символов