

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Национальный исследовательский университет ИТМО»

*Факультет программной инженерии и компьютерной техники*

**Лабораторная работа №5**

по дисциплине

«Основы профессиональной деятельности»

Вариант №2315

Выполнил:

Студент группы Р3113

Султанов Артур Радикович

Проверил:

Блохина Елена Николаевна

г. Санкт-Петербург

2023г.

## Оглавление

<b>Оглавление</b>	<b>2</b>
<b>Задание</b>	<b>3</b>
Часть 2. Описание программы	6
Назначение программы	6
ОПИ, ОДЗ	6
Расположение данных	6
Адреса первой и последней выполняемой команды	6
Часть 3. Трассировка программы	7
<b>Заключение</b>	<b>8</b>

## 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.

1. Программа осуществляет асинхронный ввод данных с ВУ-2
2. Программа начинается с адреса 420. Размещаемая строка находится по адресу 5С7.
3. Строка должна быть представлена в кодировке КОИ-8.
4. Формат представления строки в памяти: АДР1: СИМВ2 СИМВ1 АДР2: СИМВ4 СИМВ3 ... СТОП\_СИМВ.
5. Ввод или вывод строки должен быть завершен по символу с кодом 0А (NL). Стоп символ является обычным символом строки и подчиняется тем же правилам расположения в памяти что и другие символы строки.

## Часть 1. Текст исходной программы

Сообщение: Молоко!

В КОИ-8: ED CF CC CF CB CF 21

11101101 11001111 11001100 11001111 11001011 11001111  
00100001

В UTF-8: D09C D0BE D0BB D0BE D0BA D0BE 21

В UTF-16 (BE): 041C 043E 043B 043E 043A 043E 0021

Текст программы на языке ассемблера БЭВМ:

```
                                org  0x420
START:                          CLA
                                ST   IS_EVEN
                                LD   STR_BEGIN      ; initialization
                                ST   STR_PTR
READ_LOOP:                     NOP
DEV_WAIT:                      IN    0x5
                                AND   #0x40        ; spin loop
                                BEQ   DEV_WAIT
                                IN    0x4          ; read `c` from dev
                                ST    C
                                LD    IS_EVEN
                                AND   #1           ; determine the way
to write `c` on this iteration
                                BEQ   PREPARE_ODD_VALUE
                                JUMP  PREPARE_EVEN_VALUE
```

```

PREPARE_ODD_VALUE: LD    C                      ; if (is_even == 0)
-> `c`

JUMP WRITE_VALUE

PREPARE_EVEN_VALUE: LD    C
SWAB                ; if (is_even == 0)
-> `c << 8`

WRITE_VALUE:       OR     (STR_PTR)             ; ptr |= val
ST     (STR_PTR)
LD     STR_PTR
ADD    IS_EVEN          ; +1 each 2nd
iteration

ST     STR_PTR
LD     #1
SUB    IS_EVEN          ; is_even = 1 -
is_even;

ST     IS_EVEN
LD     C
CMP    #0xA             ; if (c == '\n')
goto READ_LOOP

BNE    READ_LOOP
HLT

C:        WORD ?
IS_EVEN:  WORD 0         ; is it even
iteration?

STR_BEGIN: WORD 0x5C7
STR_PTR:  WORD ?

```

org 0x5C7

STR:

WORD ?

Текст исходной программы:

Адрес	Код команды	Мнемоника	Комментарии
420	0200	CLA	Очистка аккумулятора
421	EE1D	ST IP+0x1D	Сохранение 0 в IS_EVEN
422	AE1D	LD IP+0x1D	Сохранение адреса начала строки STR_BEGIN в STR_PTR
423	EE1D	ST IP+0x1D	
424	0000	NOP	Ничего, начало цикла
425	1205	IN 0x5	Spin loop - ожидание готовности BY-2
426	2F40	AND #0x40	
427	F0FD	BEQ IP-3	
428	1204	IN 0x4	Чтение данных с BY-2 в ячейку C
429	EE14	ST IP+0x14	
42A	AE14	LD IP+0x14	Проверка на четность итерации цикла (IS_EVEN)
42B	2F01	AND #1	
42C	F001	BEQ IP+1	Если итерация нечетная, то переход на соответствующую обработку по адресу 0x42E
42D	CE02	JUMP IP+2	Если итерация нечетная, то переход на соответствующую обработку по адресу 0x430
42E	AE0F	LD IP+0xF	Обработка нечетной итерации: загрузка C в аккумулятор. Переход на операцию записи символа в строку
42F	CE02	JUMP IP+2	
430	AE0D	LD IP+0xD	Обработка четной итерации: загрузка C в аккумулятор и обмен старшего и младших байтов аккумулятора. Переход на операцию записи символа в строку
431	0680	SWAB	

432	380E	OR (IP+0xE)	Запись получившейся в зависимости формы знака (старший/младший байт) по адресу, хранящемуся в STR_PTR
433	E80D	ST (IP+0xD)	
434	AE0C	LD IP+0xC	Увеличение адреса, хранящегося в STR_PTR на значение, равное IS_EVEN (то есть +1 каждую вторую итерацию). Необходимо, так как в одной ячейке хранится 2 символа, а значит адрес должен увеличиваться на 1 после записи 2 символов.
435	4E09	ADD IP+0x9	
436	EE0A	ST IP+0xA	
437	AF01	LD #1	IS_EVEN = 1 - IS_EVEN. Так как изначально IS_EVEN = 0, то фактически IS_EVEN зациклен между 0 и 1.
438	6E06	SUB IP+0x6	
439	EE05	ST IP+0x5	
43A	AE03	LD IP+0x3	Если введенный символ равен стоп-символу - '\n' (0xA), то цикл завершается. Если введенный символ не равен стоп-символу, цикл продолжается.
43B	7F0A	CMP #0xA	
43C	F1E7	BNE IP-0x19	
43D	0100	HLT	Операция “Остановка”

## Часть 2. Описание программы

Назначение программы

Асинхронное считывание символов (в кодировке КОИ-8) с ВУ-2 и их запись в строку, по два на ячейку. Считывание происходит до тех пор, пока введенный символ не будет равен стоп-символу '\n' (0xA).

ОПИ, ОДЗ

$C \in [0; 0xff]$  (однобайтовый символ с ВУ-2)

$IS\_EVEN \in [0; 1]$  (используется как булева переменная, изначально задается 0)

Строка начинается на 0x5C7, тогда:

$STR\_BEGIN = 0x5C7$

$STR\_PTR \in [0x5C7; 0x7FF]$

Но если же рассматривать все варианты ее расположения, то:

$STR\_BEGIN \in [0; 0x41F] \cup [0x442; 0x7FF]$  (изначально равен 0x5C7)

$STR\_PTR \in [0; 0x41F] \cup [0x442; 0x7FF]$

Учитывая, что строка начинается на 0x5C7, то максимальное кол-во символов в строке (вместе с '\n') равно:

$$(0x7FF - 0x5C7 + 1) * 2 = 568 * 2 = 1136$$

Расположение данных

Адрес	Значение
43E	С - получаемый с устройства символ
43F	IS_EVEN - переменная, в которой хранится четность итерации цикла
440	STR_BEGIN - указатель на начало строки
441	STR_PTR - указатель на элемент строки
5C7	STR - строка

Адреса первой и последней выполняемой команды

Адрес первой выполняемой команды	Адрес последней выполняемой команды
420	43D



### Часть 3. Трассировка программы

Адрес	Код	IP	CR	AR	DR	SP	BR	AC	NZVC	Адрес	Новый код
420	0200	421	0200	420	0200	000	0420	0000	0100		
421	EE1D	422	EE1D	43F	0000	000	001D	0000	0100	43F	0000
422	AE1D	423	AE1D	440	05C7	000	001D	05C7	0000		
423	EE1D	424	EE1D	441	05C7	000	001D	05C7	0000	441	05C7
424	0000	425	0000	424	0000	000	0424	05C7	0000		
425	1205	426	1205	425	1205	000	0425	0540	0000		
426	2F40	427	2F40	426	0040	000	0040	0040	0000		
427	F0FD	428	F0FD	427	F0FD	000	0427	0040	0000		
428	1204	429	1204	428	1204	000	0428	00ED	0000		
429	EE14	42A	EE14	43E	00ED	000	0014	00ED	0000	43E	00ED
42A	AE14	42B	AE14	43F	0000	000	0014	0000	0100		
42B	2F01	42C	2F01	42B	0001	000	0001	0000	0100		
42C	F001	42E	F001	42C	F001	000	0001	0000	0100		
42E	AE0F	42F	AE0F	43E	00ED	000	000F	00ED	0000		
42F	CE02	432	CE02	42F	0432	000	0002	00ED	0000		
432	380E	433	380E	5C7	0000	000	FF12	00ED	0000		
433	E80D	434	E80D	5C7	00ED	000	000D	00ED	0000	5C7	00ED
434	AE0C	435	AE0C	441	05C7	000	000C	05C7	0000		
435	4E09	436	4E09	43F	0000	000	0009	05C7	0000		
436	EE0A	437	EE0A	441	05C7	000	000A	05C7	0000	441	05C7
437	AF01	438	AF01	437	0001	000	0001	0001	0000		
438	6E06	439	6E06	43F	0000	000	0006	0001	0001		
439	EE05	43A	EE05	43F	0001	000	0005	0001	0001	43F	0001
43A	AE03	43B	AE03	43E	00ED	000	0003	00ED	0001		
43B	7F0A	43C	7F0A	43B	000A	000	000A	00ED	0001		
43C	F1E7	424	F1E7	43C	F1E7	000	FFE7	00ED	0001		
424	0000	425	0000	424	0000	000	0424	00ED	0001		
425	1205	426	1205	425	1205	000	0425	0040	0001		
426	2F40	427	2F40	426	0040	000	0040	0040	0001		
427	F0FD	428	F0FD	427	F0FD	000	0427	0040	0001		
428	1204	429	1204	428	1204	000	0428	00CF	0001		

429	EE14	42A	EE14	43E	00CF	000	0014	00CF	0001	43E	00CF
42A	AE14	42B	AE14	43F	0001	000	0014	0001	0001		
42B	2F01	42C	2F01	42B	0001	000	0001	0001	0001		
42C	F001	42D	F001	42C	F001	000	042C	0001	0001		
42D	CE02	430	CE02	42D	0430	000	0002	0001	0001		
430	AE0D	431	AE0D	43E	00CF	000	000D	00CF	0001		
431	0680	432	0680	431	0680	000	0431	CF00	1001		
432	380E	433	380E	5C7	00ED	000	3012	CFED	1001		
433	E80D	434	E80D	5C7	CFED	000	000D	CFED	1001	5C7	CFED
434	AE0C	435	AE0C	441	05C7	000	000C	05C7	0001		
435	4E09	436	4E09	43F	0001	000	0009	05C8	0000		
436	EE0A	437	EE0A	441	05C8	000	000A	05C8	0000	441	05C8
437	AF01	438	AF01	437	0001	000	0001	0001	0000		
438	6E06	439	6E06	43F	0001	000	0006	0000	0101		
439	EE05	43A	EE05	43F	0000	000	0005	0000	0101	43F	0000
43A	AE03	43B	AE03	43E	00CF	000	0003	00CF	0001		
43B	7F0A	43C	7F0A	43B	000A	000	000A	00CF	0001		
43C	F1E7	424	F1E7	43C	F1E7	000	FFE7	00CF	0001		
424	0000	425	0000	424	0000	000	0424	00CF	0001		
425	1205	426	1205	425	1205	000	0425	0040	0001		
426	2F40	427	2F40	426	0040	000	0040	0040	0001		
427	F0FD	428	F0FD	427	F0FD	000	0427	0040	0001		
428	1204	429	1204	428	1204	000	0428	00CC	0001		
429	EE14	42A	EE14	43E	00CC	000	0014	00CC	0001	43E	00CC
42A	AE14	42B	AE14	43F	0000	000	0014	0000	0101		
42B	2F01	42C	2F01	42B	0001	000	0001	0000	0101		
42C	F001	42E	F001	42C	F001	000	0001	0000	0101		
42E	AE0F	42F	AE0F	43E	00CC	000	000F	00CC	0001		
42F	CE02	432	CE02	42F	0432	000	0002	00CC	0001		
432	380E	433	380E	5C8	0000	000	FF33	00CC	0001		
433	E80D	434	E80D	5C8	00CC	000	000D	00CC	0001	5C8	00CC
434	AE0C	435	AE0C	441	05C8	000	000C	05C8	0001		
435	4E09	436	4E09	43F	0000	000	0009	05C8	0000		

436	EE0A	437	EE0A	441	05C8	000	000A	05C8	0000	441	05C8
437	AF01	438	AF01	437	0001	000	0001	0001	0000		
438	6E06	439	6E06	43F	0000	000	0006	0001	0001		
439	EE05	43A	EE05	43F	0001	000	0005	0001	0001	43F	0001
43A	AE03	43B	AE03	43E	00CC	000	0003	00CC	0001		
43B	7F0A	43C	7F0A	43B	000A	000	000A	00CC	0001		
43C	F1E7	424	F1E7	43C	F1E7	000	FFE7	00CC	0001		
424	0000	425	0000	424	0000	000	0424	00CC	0001		
425	1205	426	1205	425	1205	000	0425	0040	0001		
426	2F40	427	2F40	426	0040	000	0040	0040	0001		
427	F0FD	428	F0FD	427	F0FD	000	0427	0040	0001		
428	1204	429	1204	428	1204	000	0428	00CF	0001		
429	EE14	42A	EE14	43E	00CF	000	0014	00CF	0001	43E	00CF
42A	AE14	42B	AE14	43F	0001	000	0014	0001	0001		
42B	2F01	42C	2F01	42B	0001	000	0001	0001	0001		
42C	F001	42D	F001	42C	F001	000	042C	0001	0001		
42D	CE02	430	CE02	42D	0430	000	0002	0001	0001		
430	AE0D	431	AE0D	43E	00CF	000	000D	00CF	0001		
431	0680	432	0680	431	0680	000	0431	CF00	1001		
432	380E	433	380E	5C8	00CC	000	3033	CFCC	1001		
433	E80D	434	E80D	5C8	CFCC	000	000D	CFCC	1001	5C8	CFCC
434	AE0C	435	AE0C	441	05C8	000	000C	05C8	0001		
435	4E09	436	4E09	43F	0001	000	0009	05C9	0000		
436	EE0A	437	EE0A	441	05C9	000	000A	05C9	0000	441	05C9
437	AF01	438	AF01	437	0001	000	0001	0001	0000		
438	6E06	439	6E06	43F	0001	000	0006	0000	0101		
439	EE05	43A	EE05	43F	0000	000	0005	0000	0101	43F	0000
43A	AE03	43B	AE03	43E	00CF	000	0003	00CF	0001		
43B	7F0A	43C	7F0A	43B	000A	000	000A	00CF	0001		
43C	F1E7	424	F1E7	43C	F1E7	000	FFE7	00CF	0001		
424	0000	425	0000	424	0000	000	0424	00CF	0001		
425	1205	426	1205	425	1205	000	0425	0040	0001		
426	2F40	427	2F40	426	0040	000	0040	0040	0001		

427	F0FD	428	F0FD	427	F0FD	000	0427	0040	0001		
428	1204	429	1204	428	1204	000	0428	00CB	0001		
429	EE14	42A	EE14	43E	00CB	000	0014	00CB	0001	43E	00CB
42A	AE14	42B	AE14	43F	0000	000	0014	0000	0101		
42B	2F01	42C	2F01	42B	0001	000	0001	0000	0101		
42C	F001	42E	F001	42C	F001	000	0001	0000	0101		
42E	AE0F	42F	AE0F	43E	00CB	000	000F	00CB	0001		
42F	CE02	432	CE02	42F	0432	000	0002	00CB	0001		
432	380E	433	380E	5C9	0000	000	FF34	00CB	0001		
433	E80D	434	E80D	5C9	00CB	000	000D	00CB	0001	5C9	00CB
434	AE0C	435	AE0C	441	05C9	000	000C	05C9	0001		
435	4E09	436	4E09	43F	0000	000	0009	05C9	0000		
436	EE0A	437	EE0A	441	05C9	000	000A	05C9	0000	441	05C9
437	AF01	438	AF01	437	0001	000	0001	0001	0000		
438	6E06	439	6E06	43F	0000	000	0006	0001	0001		
439	EE05	43A	EE05	43F	0001	000	0005	0001	0001	43F	0001
43A	AE03	43B	AE03	43E	00CB	000	0003	00CB	0001		
43B	7F0A	43C	7F0A	43B	000A	000	000A	00CB	0001		
43C	F1E7	424	F1E7	43C	F1E7	000	FFE7	00CB	0001		
424	0000	425	0000	424	0000	000	0424	00CB	0001		
425	1205	426	1205	425	1205	000	0425	0040	0001		
426	2F40	427	2F40	426	0040	000	0040	0040	0001		
427	F0FD	428	F0FD	427	F0FD	000	0427	0040	0001		
428	1204	429	1204	428	1204	000	0428	00CF	0001		
429	EE14	42A	EE14	43E	00CF	000	0014	00CF	0001	43E	00CF
42A	AE14	42B	AE14	43F	0001	000	0014	0001	0001		
42B	2F01	42C	2F01	42B	0001	000	0001	0001	0001		
42C	F001	42D	F001	42C	F001	000	042C	0001	0001		
42D	CE02	430	CE02	42D	0430	000	0002	0001	0001		
430	AE0D	431	AE0D	43E	00CF	000	000D	00CF	0001		
431	0680	432	0680	431	0680	000	0431	CF00	1001		
432	380E	433	380E	5C9	00CB	000	3034	CFCB	1001		
433	E80D	434	E80D	5C9	CFCB	000	000D	CFCB	1001	5C9	CFCB

434	AE0C	435	AE0C	441	05C9	000	000C	05C9	0001		
435	4E09	436	4E09	43F	0001	000	0009	05CA	0000		
436	EE0A	437	EE0A	441	05CA	000	000A	05CA	0000	441	05CA
437	AF01	438	AF01	437	0001	000	0001	0001	0000		
438	6E06	439	6E06	43F	0001	000	0006	0000	0101		
439	EE05	43A	EE05	43F	0000	000	0005	0000	0101	43F	0000
43A	AE03	43B	AE03	43E	00CF	000	0003	00CF	0001		
43B	7F0A	43C	7F0A	43B	000A	000	000A	00CF	0001		
43C	F1E7	424	F1E7	43C	F1E7	000	FFE7	00CF	0001		
424	0000	425	0000	424	0000	000	0424	00CF	0001		
425	1205	426	1205	425	1205	000	0425	0040	0001		
426	2F40	427	2F40	426	0040	000	0040	0040	0001		
427	F0FD	428	F0FD	427	F0FD	000	0427	0040	0001		
428	1204	429	1204	428	1204	000	0428	0021	0001		
429	EE14	42A	EE14	43E	0021	000	0014	0021	0001	43E	0021
42A	AE14	42B	AE14	43F	0000	000	0014	0000	0101		
42B	2F01	42C	2F01	42B	0001	000	0001	0000	0101		
42C	F001	42E	F001	42C	F001	000	0001	0000	0101		
42E	AE0F	42F	AE0F	43E	0021	000	000F	0021	0001		
42F	CE02	432	CE02	42F	0432	000	0002	0021	0001		
432	380E	433	380E	5CA	0000	000	FFDE	0021	0001		
433	E80D	434	E80D	5CA	0021	000	000D	0021	0001	5CA	0021
434	AE0C	435	AE0C	441	05CA	000	000C	05CA	0001		
435	4E09	436	4E09	43F	0000	000	0009	05CA	0000		
436	EE0A	437	EE0A	441	05CA	000	000A	05CA	0000	441	05CA
437	AF01	438	AF01	437	0001	000	0001	0001	0000		
438	6E06	439	6E06	43F	0000	000	0006	0001	0001		
439	EE05	43A	EE05	43F	0001	000	0005	0001	0001	43F	0001
43A	AE03	43B	AE03	43E	0021	000	0003	0021	0001		
43B	7F0A	43C	7F0A	43B	000A	000	000A	0021	0001		
43C	F1E7	424	F1E7	43C	F1E7	000	FFE7	0021	0001		
424	0000	425	0000	424	0000	000	0424	0021	0001		
425	1205	426	1205	425	1205	000	0425	0040	0001		

426	2F40	427	2F40	426	0040	000	0040	0040	0001		
427	F0FD	428	F0FD	427	F0FD	000	0427	0040	0001		
428	1204	429	1204	428	1204	000	0428	000A	0001		
429	EE14	42A	EE14	43E	000A	000	0014	000A	0001	43E	000A
42A	AE14	42B	AE14	43F	0001	000	0014	0001	0001		
42B	2F01	42C	2F01	42B	0001	000	0001	0001	0001		
42C	F001	42D	F001	42C	F001	000	042C	0001	0001		
42D	CE02	430	CE02	42D	0430	000	0002	0001	0001		
430	AE0D	431	AE0D	43E	000A	000	000D	000A	0001		
431	0680	432	0680	431	0680	000	0431	0A00	0001		
432	380E	433	380E	5CA	0021	000	F5DE	0A21	0001		
433	E80D	434	E80D	5CA	0A21	000	000D	0A21	0001	5CA	0A21
434	AE0C	435	AE0C	441	05CA	000	000C	05CA	0001		
435	4E09	436	4E09	43F	0001	000	0009	05CB	0000		
436	EE0A	437	EE0A	441	05CB	000	000A	05CB	0000	441	05CB
437	AF01	438	AF01	437	0001	000	0001	0001	0000		
438	6E06	439	6E06	43F	0001	000	0006	0000	0101		
439	EE05	43A	EE05	43F	0000	000	0005	0000	0101	43F	0000
43A	AE03	43B	AE03	43E	000A	000	0003	000A	0001		
43B	7F0A	43C	7F0A	43B	000A	000	000A	000A	0101		
43C	F1E7	43D	F1E7	43C	F1E7	000	043C	000A	0101		
43D	0100	43E	0100	43D	0100	000	043D	000A	0101		

Программа с изменениями:

```

                org 0x420
; READ
START:         CLA
                ST   IS_EVEN
                LD   STR_BEGIN      ; initialization
                ST   STR_PTR

```

```

READ_LOOP:      NOP

DEV2_WAIT:      IN    0x5
                 AND   #0x40                ; spin loop
                 BEQ   DEV2_WAIT

                 IN    0x4                ; read `c` from dev
                 ST    C

                 LD    IS_EVEN
                 AND   #1                ; determine the way
to write `c` on this iteration
                 BEQ   PREPARE_ODD_VAL0
                 JUMP  PREPARE_EVEN_VAL0

PREPARE_ODD_VAL0: LD    C                ; if (is_even
== 0) -> `c`
                 JUMP  SAVE_VAL

PREPARE_EVEN_VAL0: LD    C
                 SWAB                ; if (is_even == 0)
-> `c << 8`

SAVE_VAL:       OR    (STR_PTR)          ; ptr |= val
                 ST    (STR_PTR)

                 LD    STR_PTR

```

```

iteration
    ADD    IS_EVEN        ;    +1    each    2nd
iteration
    ST     STR_PTR

    LD     #1
    SUB    IS_EVEN        ;    is_even    =    1    -
is_even;
    ST     IS_EVEN

    LD     C
    CMP    #0xA           ;    if    (c    ==    '\n')
goto READ_LOOP
    BNE    READ_LOOP

; WRITE

    CLA
    ST     IS_EVEN
    LD     STR_BEGIN      ;    str_ptr    =
str_begin
    ST     STR_PTR

WRITE_LOOP:    NOP

    LD     IS_EVEN
    AND    #1             ;    determine the way
to write `str[i]` on this iteration

```



```

        BEQ  PREPARE_ODD_VAL1
        JUMP PREPARE_EVEN_VAL1

```

```

PREPARE_ODD_VAL1:  LD    (STR_PTR)          ; if (is_even
== 0) -> `str[i]`
                    JUMP EXTRACT_VAL

```

```

PREPARE_EVEN_VAL1: LD    (STR_PTR)
                    SWAB                    ; if (is_even == 0)
-> `str[i] << 8`

```

```

EXTRACT_VAL:      AND    MAX_BYTE          ; c = str[i] &
0xff;
                    ST    C

                    CMP    MIN_RU_SYMBOL
                    BGE    DEV5_WAIT        ; 0xC0 <= arg <=
0xff
                    JUMP  NEXT_ITER_PREP1

```

```

DEV5_WAIT:        IN     0xD
                    AND    #0x40            ; spin loop
                    BEQ    DEV5_WAIT

                    LD     C
                    OUT    0xC

```

```

NEXT_ITER_PREP1: LD    STR_PTR
                  ADD   IS_EVEN          ;   +1   each   2nd
iteration
                  ST    STR_PTR

                  LD    #1
                  SUB   IS_EVEN          ;   is_even   =   1   -
is_even;
                  ST    IS_EVEN

                  LD    C
                  CMP   #0xA             ;   if   (c   ==   '\n')
goto WRITE_LOOP
                  BNE   WRITE_LOOP

                  HLT

; DATA

C:                WORD ?
IS_EVEN:          WORD 0                 ;   is   it   an   even
iteration?
STR_BEGIN:        WORD 0x5C7
STR_PTR:          WORD ?
MAX_BYTE:         WORD 0x00FF

```

MIN\_RU\_SYMBOL: WORD 0x00C0

org 0x5C7

STR: WORD ?

## Заключение

В рамках данной лабораторной работы я познакомился с вводом-выводом в БЭВМ, взаимодействием и структурой внешних устройств БЭВМ, а также различными кодировками.