

Университет ИТМО

Факультет программной инженерии и компьютерной техники
Образовательная программа системное и прикладное
программное обеспечение

Лабораторная работа №5
По дисциплине "Основы профессиональной деятельности"
Вариант 9500

Выполнил студент группы Р3109
Евграфов Артём Андреевич
Проверила:
Ткешелашвили Нино Мерабиевна

Санкт-Петербург 2025

Содержание

1. Задание варианта 9500	2
2. Описание программы	2
3. ОП и ОДЗ исходных данных и результата	2
3.1. Область представления	2
3.2. Область определения	3
4. Трассировка программы	3
5. Дополнительное задание	4
6. Вывод	6

1. Задание варианта 9500

Лабораторная работа №5

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.

Введите номер варианта

1. Программа осуществляет асинхронный ввод данных с ВУ-2
2. Программа начинается с адреса 28В₁₆. Размещаемая строка находится по адресу 5СD₁₆.
3. Строка должна быть представлена в кодировке КОИ-8.
4. Формат представления строки в памяти: АДР1: СИМВ1 СИМВ2 АДР2: СИМВ3 СИМВ4 ... 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.

2. Описание программы

Адрес	Содержимое	Мнемоника	Комментарии
28В	05СD	ADR	Ячейка для инкрементирования адреса результата
28С	0200	CLA	0000 -> AC
28D	1205	IN 5	Ввод содержимого SR ВУ2 в 6-й бит AC
28E	2F40	AND #40	Проверяем условие "6-й бит AC = 1"
28F	F0FD	BZS (IP - 3)	Переход на 0x28D, если 6-й бит = 0 (Z == 1)
290	1204	IN 4	Ввод содержимого DR ВУ2 в младший байт AC
291	7F0A	CMP #0A	Сравнение AC со стоп-символом 0A
292	F00B	BZS (IP + 11)	Переход на 0x29E, если Z == 1
293	0680	SWAB	Свап байтов AC
294	E8F6	ST (IP - 10)	Сохранение первого символа по адресу из ячейки 0x28B
295	1205	IN 5	Ввод содержимого SR ВУ2 в 6-й бит AC
296	2F40	AND #40	Проверяем условие "6-й бит AC = 1" (старший байт затирается из-за расширения знака)
297	F0FD	BZS (IP - 3)	Переход на 0x294, если 6-й бит = 0 (Z == 1)
298	1204	IN 4	Ввод содержимого DR ВУ2 в младший байт AC
299	7F0A	CMP #0A	Сравнение AC со стоп-символом 0A
29A	F005	BZS (IP + 5)	Переход на 0x29E, если Z == 1
29B	48EF	ADD (IP - 17)	Теперь в AC хранится СИМВ1 СИМВ2
29C	EAEE	ST (IP - 18)+	Сохранение второго символа по адресу из ячейки 0x28B и увеличение значения ячейки на 1
29D	CEED	JUMP (IP - 18)	Переход на адрес 0x28D (если все окей, продолжаем считывать)
29E	0680	SWAB	Свап байтов AC (так как 0A должен быть слева)
29F	CE01	JUMP (IP + 1)	Переходим на 0x2A0
2A0	48EA	ADD (IP - 22)	Теперь в AC хранится СИМВ1 СИМВ2=0A
2A1	E8E9	ST (IP - 23)	Сохранение результата с символом 0A
2A2	0100	HLT	остановка программы
5CD	0000	RES	Ячейка для сохранения символа слова (далее инкрементируется)

3. ОП и ОДЗ исходных данных и результата

3.1. Область представления

RES - 16-разрядная ячейка для сохранения 2-х символов. Старший байт - код первого символа, младший байт - код второго символа.

ADR - 11-разрядное беззнаковое число. Ячейка для хранения адреса начала символов кода.

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
8.	— 2500	┆ 2502	┆ 250C	┆ 2510	┆ 2514	┆ 2518	┆ 251C	┆ 2524	┆ 252C	┆ 2534	┆ 253C	■ 2580	■ 2584	■ 2588	■ 258C	■ 2590
9.	░ 2591	▒ 2592	▓ 2593	┆ 2320	■ 25A0	· 2219	√ 221A	≈ 2248	≤ 2264	≥ 2265		┆ A0	○ B0	2 B2	· B7	÷ F7
A.	= 2550	 2551	┆ 2552	ë 451	π 2553	┆ 2554	┆ 2555	┆ 2556	┆ 2557	┆ 2558	┆ 2559	┆ 255A	┆ 255B	┆ 255C	┆ 255D	┆ 255E
B.	 255F	┆ 2560	┆ 2561	Ë 401	 2562	 2563	┆ 2564	┆ 2565	┆ 2566	┆ 2567	┆ 2568	┆ 2569	┆ 256A	┆ 256B	┆ 256C	© A9
C.	Ю 44E	а 430	б 431	ц 446	д 434	е 435	ф 444	г 433	х 445	и 438	й 439	к 43A	л 43B	м 43C	н 43D	о 43E
D.	П 43F	я 44F	р 440	с 441	т 442	у 443	ж 436	в 432	ь 44C	ы 44B	з 437	ш 448	э 44D	щ 449	ч 447	ъ 44A
E.	Ю 42E	А 410	Б 411	Ц 426	Д 414	Е 415	Ф 424	Г 413	Х 425	И 418	Й 419	К 41A	Л 41B	М 41C	Н 41D	О 41E
F.	П 41F	Я 42F	Р 420	С 421	Т 422	У 423	Ж 416	В 412	Ь 42C	Ы 42B	З 417	Ш 428	Э 42D	Щ 429	Ч 427	Ъ 42A

3.2. Область определения

8-ричный код символа $\in [0x20; 0xFF \setminus \{0x7F\}]$.

Максимально возможное количество символов для ввода = 1126.

$$(2047(0x7FF_{16}) - 1484(0x5CD_{16})) \cdot 2 = 1126$$

4. Трассировка программы

Слово для трассировки: СМЕСЬ.

Слово в кодировке КОИ-8: F3 ED E5 F3 F8.

Слово в кодировке UTF-8 (BE): D0A1 D09C D095 D0A1 D0AC

Слово в кодировке UTF-16 (BE): 0421 041C 0415 0421 042C

Таблица трассировки первых двух символов слова СМЕСЬ:

Адр	Знач	IP	CR	AR	DR	SP	BR	AC	NZVC	Адр	Знач
28B	05CD	28B	0000	000	0000	000	0000	0000	0100		
28B	05CD	28C	05CD	28B	05CD	000	028B	0000	0100		
28C	0200	28D	0200	28C	0200	000	028C	0000	0100		
28D	1205	28E	1205	28D	1205	000	028D	0040	0100		
28E	2F40	28F	2F40	28E	0040	000	0040	0040	0000		
28F	F0FD	290	F0FD	28F	F0FD	000	028F	0040	0000		
290	1204	291	1204	290	1204	000	0290	00F3	0000		
291	7F0A	292	7F0A	291	000A	000	000A	00F3	0001		
292	F00B	293	F00B	292	F00B	000	0292	00F3	0001		
293	0680	294	0680	293	0680	000	0293	F300	1001		
294	E8F6	295	E8F6	5CD	F300	000	FFF6	F300	1001	5CD	F300
295	1205	296	1205	295	1205	000	0295	F300	1001		
296	2F40	297	2F40	296	0040	000	0040	0000	0101		
297	F0FD	295	F0FD	297	F0FD	000	FFFD	0000	0101		
295	1205	296	1205	295	1205	000	0295	0040	0101		
296	2F40	297	2F40	296	0040	000	0040	0040	0001		
297	F0FD	298	F0FD	297	F0FD	000	0297	0040	0001		
298	1204	299	1204	298	1204	000	0298	00ED	0001		
299	7F0A	29A	7F0A	299	000A	000	000A	00ED	0001		
29A	F005	29B	F00B	29A	F00B	000	029A	00ED	0001		
29B	48EF	29C	48EF	5CD	F300	000	FFEF	F3ED	1000		

29C	EAEE	29D	EAEE	5CD	F3ED	000	FFEE	F3ED	1000	28B	05CE
										5CD	F3ED
29D	CEEF	28D	CEEF	29D	028D	000	FFEF	F3ED	1000		
28D	1205	28E	1205	28D	1205	000	028D	F340	1000		
28E	2F40	28F	2F40	28E	0040	000	0040	0040	0000		
28F	F0FD	290	F0FD	28F	F0FD	000	028F	0040	0000		
290	1204	291	1204	290	1204	000	0290	000A	0000		
291	7F0A	292	7F0A	291	000A	000	000A	000A	0101		
292	F00B	29E	F00B	292	F00B	000	000B	000A	0101		
29E	0680	29F	0680	29E	0680	000	029E	0A00	0001		
29F	CE01	2A1	CE01	29F	02A1	000	0001	0A00	0001		
2A1	E8E9	2A2	E8E9	5CE	0A00	000	FFE9	0A00	0001	5CE	0A00
2A2	0100	2A3	0100	2A2	0100	000	02A2	0A00	0001		

5. Дополнительное задание

С ВУ-8 (клавиатура) вводится строка, enter - завершение ввода. После окончания ввода, на ВУ-5 (принтер) вывести коды символов в 16-ричной системе счисления через пробел. Кодировка любая.

```

1  ORG 0x000
2  ADR: WORD $RES
3  COUNTER: WORD 0x0000
4
5  BEGIN: CLA
6
7  READ: IN 0x19
8      AND #0x40
9      BZS READ
10     IN 0x18
11     CMP #0x0A
12     BZS SAVE
13     ST (ADR)+
14     LD COUNTER
15     INC
16     ST COUNTER
17     JUMP READ
18
19  SAVE: CLA
20     LD COUNTER
21     CMP #0x00
22     BZS EXIT
23     JUMP START_WRITING
24
25  START_WRITING: CLA
26     IN 0xD
27     AND #0x40
28     BZS START_WRITING
29     LD ADR
30     SUB COUNTER
31     ST ADR
32     JUMP WRITE
33
34  WRITE: CLA
35     LD (ADR)
36     ASR
37     ASR
38     ASR
39     ASR
40     PUSH

```

```

41      CALL FUNC
42      POP
43      OUT 0xC
44      LD (ADR)+
45      AND #0x0F
46      PUSH
47      CALL FUNC
48      POP
49      OUT 0xC
50      LD #0x9A
51      OUT 0xC
52      LOOP COUNTER
53      JUMP WRITE
54      HLT
55
56 FUNC: LD &1
57      CMP #0x00
58      BZS SET_0
59      CMP #0x01
60      BZS SET_1
61      CMP #0x02
62      BZS SET_2
63      CMP #0x03
64      BZS SET_3
65      CMP #0x04
66      BZS SET_4
67      CMP #0x05
68      BZS SET_5
69      CMP #0x06
70      BZS SET_6
71      CMP #0x07
72      BZS SET_7
73      CMP #0x08
74      BZS SET_8
75      CMP #0x09
76      BZS SET_9
77      CMP #0x0A
78      BZS SET_A
79      CMP #0x0B
80      BZS SET_B
81      CMP #0x0C
82      BZS SET_C
83      CMP #0x0D
84      BZS SET_D
85      CMP #0x0E
86      BZS SET_E
87      CMP #0x0F
88      BZS SET_F
89
90 SET_0: LD #0x30
91      ST &1
92      RET
93
94 SET_1: LD #0x31
95      ST &1
96      RET
97
98 SET_2: LD #0x32
99      ST &1
100     RET
101

```

```

102 SET_3: LD #0x33
103      ST &1
104      RET
105
106 SET_4: LD #0x34
107      ST &1
108      RET
109
110 SET_5: LD #0x35
111      ST &1
112      RET
113
114 SET_6: LD #0x36
115      ST &1
116      RET
117
118 SET_7: LD #0x37
119      ST &1
120      RET
121
122 SET_8: LD #0x38
123      ST &1
124      RET
125
126 SET_9: LD #0x39
127      ST &1
128      RET
129
130 SET_A: LD #0x41
131      ST &1
132      RET
133
134 SET_B: LD #0x42
135      ST &1
136      RET
137
138 SET_C: LD #0x43
139      ST &1
140      RET
141
142 SET_D: LD #0x44
143      ST &1
144      RET
145
146 SET_E: LD #0x45
147      ST &1
148      RET
149
150 SET_F: LD #0x46
151      ST &1
152      RET
153
154 ORG 0x2CD
155 RES: WORD 0x0000

```

6. Вывод

В ходе выполнения данной лабораторной работы я научился работать с ВУ-2, освоил команды ввода-вывода, а также познакомился с синтаксисом ассемблера БЭВМ-NG.