

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ ИТМО”**

Лабораторная работа №5

ФИО студента, вариант: Туляков Евгений Викторович, 1906

Направление подготовки(специальность): 09.03.04

Группа: P3119

ФИО преподавателя:

Лабушев Тимофей Михайлович

Санкт-Петербург, 2021

Текст задания

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 1906

1. Программа осуществляет асинхронный вывод данных на ВУ-1
2. Программа начинается с адреса 4BB₁₆. Размещаемая строка находится по адресу 5DF₁₆.
3. Стока 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 0D (CR). Стоп символ является обычным символом строки и подчиняется тем же правилам расположения в памяти что и другие символы строки.

Код программы

Метка	Мнемоника	Параметр	Описание
	ORG	0x4BB	
CUR:	WORD	0x5DF	Адрес текущего элемента в массиве
STOPW:	WORD	0x0D	Стоп-слово
BIG:	WORD	0x0000	Старший байт текущего элемента
LITTLE:	WORD	0x0000	Младший байт текущего элемента
MASK:	WORD	0x00FF	Маска для выделения младшего байта
START:	CLA		Начало программы
	LD	(CUR)+	Загрузка в Little младшего байта
	ST	LITTLE	
	SWAB		Загрузка в Big старшего байта
	ST	BIG	
S1:	IN	3	Ожидание готовности ВУ к выводу старшего байта
	AND	#0x40	
	BEQ	S1	
	LD	BIG	Проверка что старший байт не стоп-слово
	AND	MASK	
	CMP	STOPW	
	BEQ	FINISH	
	OUT	2	Вывод старшего байта
S2:	IN	3	Ожидание готовности ВУ к выводу младшего байта
	AND	#0x40	
	BEQ	S2	

Метка	Мнемоника	Параметр	Описание
	LD	LITTLE	Проверка что младший байт не стоп-слово
	AND	MASK	
	CMP	STOPW	
	BEQ	FINISH	
	OUT	2	Вывод младшего байта
	JUMP	START	Переход к новому элементу массива
FINISH:	HLT		Остановка

Описание программы

Назначение программы:

Вывод текста сохранённого в массиве в формате АДР1: СИМВ1 СИМВ2 АДР2: СИМВ3 СИМВ4 ...
СТОП_СИМВ

Расположение в памяти БЭВМ программы, исходных данных и результатов:

Ячейки 0x4BB-0x4D6 — код программы

Ячейка CUR 0x4BB — адрес первого символа строки

Ячейка STOPW 0x4BC — код символа окончания строки

Ячейка BIG 0x4BD — переменная (значение старшего байта)

Ячейка LITTLE 0x4BE — переменная (значение младшего байта)

Ячейка MASK 0x4BF — битовая маска для выделения младшего байта

Ячейки 0x5DF-(0x5DF + X - 1) — выводимая строка, если X - ее длина в 16СС

Область представления:

Ячейка 0x4BB — 11 разрядное целое беззнаковое число с фиксированной запятой

Ячейки 0x4BC - 0x4BE — символ строки в кодировке КОИ-8

Ячейка 0x4BF — набор из 16 логических значений

Ячейки 0x5DF-(0x5DF + X - 1) — два символа строки в кодировке КОИ-8

Область допустимых значений:

Все ячейки кроме CUR 0x4BB и констант MASK 0x4BF, STOPW 0x4BC могут иметь любые значения.

Ячейка 0x4BB хранит адрес первого элемента массива, следовательно массив должен лежать в промежутках [0x000, 0x4BA] или [0x4D7, 0x7FF].

Т.е., если X - это длина строки:

$$\left[\begin{array}{l} \left\{ \begin{array}{l} CUR > 0 \\ CUR + X - 1 < 0x4BB \end{array} \right. \\ \left\{ \begin{array}{l} CUR > 0x4D6 \\ CUR + X - 1 < 0x7FF \end{array} \right. \end{array} \right]$$

$$\left[\begin{array}{l} 0x000 < CUR < 0x4BB - X + 1 \\ 0x4D6 < CUR < 0x7FF - X + 1 \end{array} \right]$$

Кодировка

Символ	КОИ-8	UTF-8	UTF-16
е	0065	0065	0065
н	006E	006E	006E
с	0063	0063	0063
о	006F	006F	006F
д	0064	0064	0064
и	00C9	D0B8	0438
н	00CE	D0BD	043D
г	00C7	D0B3	0433

Трассировка с данными числами

Адр	Знчн	IP	CR	AR	DR	SP	BR	AC	NZVC	Адр	Знчн
4C0	0200	4C1	0200	4C0	0200	000	04C0	0000	0100		
4C1	AAF9	4C2	AAF9	5DF	0065	000	FFF9	0065	0000	4BB	05E0
4C2	EEFB	4C3	EEFB	4BE	0065	000	FFFB	0065	0000	4BE	0065
4C3	0680	4C4	0680	4C3	0680	000	04C3	6500	0000		
4C4	EEF8	4C5	EEF8	4BD	6500	000	FFF8	6500	0000	4BD	6500
4C5	1203	4C6	1203	4C5	1203	000	04C5	6540	0000		
4C6	2F40	4C7	2F40	4C6	0040	000	0040	0040	0000		
4C7	F0FD	4C8	F0FD	4C7	F0FD	000	04C7	0040	0000		
4C8	AEF4	4C9	AEF4	4BD	6500	000	FFF4	6500	0000		
4C9	2EF5	4CA	2EF5	4BF	00FF	000	FFF5	0000	0100		
4CA	7EF1	4CB	7EF1	4BC	000D	000	FFF1	0000	1000		
4CB	F00A	4CC	F00A	4CB	F00A	000	04CB	0000	1000		
4CC	1302	4CD	1302	4CC	1302	000	04CC	0000	1000		
4CD	1203	4CE	1203	4CD	1203	000	04CD	0040	1000		
4CE	2F40	4CF	2F40	4CE	0040	000	0040	0040	0000		
4CF	F0FD	4D0	F0FD	4CF	F0FD	000	04CF	0040	0000		
4D0	AEED	4D1	AEED	4BE	0065	000	FFED	0065	0000		
4D1	2EED	4D2	2EED	4BF	00FF	000	FFED	0065	0000		
4D2	7EE9	4D3	7EE9	4BC	000D	000	FFE9	0065	0001		
4D3	F002	4D4	F002	4D3	F002	000	04D3	0065	0001		
4D4	1302	4D5	1302	4D4	1302	000	04D4	0065	0001		

Адр	Знчн	IP	CR	AR	DR	SP	BR	AC	NZVC	Адр	Знчн
4D5	CEEA	4C0	CEEA	4D5	04C0	000	FFEA	0065	0001		
4C0	0200	4C1	0200	4C0	0200	000	04C0	0000	0101		
4C1	AAF9	4C2	AAF9	5E0	006E	000	FFF9	006E	0001	4BB	05E1
4C2	EEFB	4C3	EEFB	4BE	006E	000	FFF8	006E	0001	4BE	006E
4C3	0680	4C4	0680	4C3	0680	000	04C3	6E00	0001		
4C4	EEF8	4C5	EEF8	4BD	6E00	000	FFF8	6E00	0001	4BD	6E00
4C5	1203	4C6	1203	4C5	1203	000	04C5	6E40	0001		
4C6	2F40	4C7	2F40	4C6	0040	000	0040	0040	0001		
4C7	F0FD	4C8	F0FD	4C7	F0FD	000	04C7	0040	0001		
4C8	AEF4	4C9	AEF4	4BD	6E00	000	FFF4	6E00	0001		
4C9	2EF5	4CA	2EF5	4BF	00FF	000	FFF5	0000	0101		
4CA	7EF1	4CB	7EF1	4BC	000D	000	FFF1	0000	1000		
4CB	F00A	4CC	F00A	4CB	F00A	000	04CB	0000	1000		
4CC	1302	4CD	1302	4CC	1302	000	04CC	0000	1000		
4CD	1203	4CE	1203	4CD	1203	000	04CD	0040	1000		
4CE	2F40	4CF	2F40	4CE	0040	000	0040	0040	0000		
4CF	F0FD	4D0	F0FD	4CF	F0FD	000	04CF	0040	0000		
4D0	AEED	4D1	AEED	4BE	006E	000	FFED	006E	0000		
4D1	2EED	4D2	2EED	4BF	00FF	000	FFED	006E	0000		
4D2	7EE9	4D3	7EE9	4BC	000D	000	FFE9	006E	0001		
4D3	F002	4D4	F002	4D3	F002	000	04D3	006E	0001		
4D4	1302	4D5	1302	4D4	1302	000	04D4	006E	0001		
4D5	CEEA	4C0	CEEA	4D5	04C0	000	FFEA	006E	0001		
4C0	0200	4C1	0200	4C0	0200	000	04C0	0000	0101		
4C1	AAF9	4C2	AAF9	5E1	000D	000	FFF9	000D	0001	4BB	05E2
4C2	EEFB	4C3	EEFB	4BE	000D	000	FFF8	000D	0001	4BE	000D
4C3	0680	4C4	0680	4C3	0680	000	04C3	0D00	0001		
4C4	EEF8	4C5	EEF8	4BD	0D00	000	FFF8	0D00	0001	4BD	0D00
4C5	1203	4C6	1203	4C5	1203	000	04C5	0D40	0001		
4C6	2F40	4C7	2F40	4C6	0040	000	0040	0040	0001		
4C7	F0FD	4C8	F0FD	4C7	F0FD	000	04C7	0040	0001		
4C8	AEF4	4C9	AEF4	4BD	0D00	000	FFF4	0D00	0001		
4C9	2EF5	4CA	2EF5	4BF	00FF	000	FFF5	0000	0101		
4CA	7EF1	4CB	7EF1	4BC	000D	000	FFF1	0000	1000		
4CB	F00A	4CC	F00A	4CB	F00A	000	04CB	0000	1000		

Адр	Знчн	IP	CR	AR	DR	SP	BR	AC	NZVC	Адр	Знчн
4CC	1302	4CD	1302	4CC	1302	000	04CC	0000	1000		
4CD	1203	4CE	1203	4CD	1203	000	04CD	0040	1000		
4CE	2F40	4CF	2F40	4CE	0040	000	0040	0040	0000		
4CF	F0FD	4D0	F0FD	4CF	F0FD	000	04CF	0040	0000		
4D0	AEED	4D1	AEED	4BE	000D	000	FFED	000D	0000		
4D1	2EED	4D2	2EED	4BF	00FF	000	FFED	000D	0000		
4D2	7EE9	4D3	7EE9	4BC	000D	000	FFE9	000D	0101		
4D3	F002	4D6	F002	4D3	F002	000	0002	000D	0101		
4D6	0100	4D7	0100	4D6	0100	000	04D6	000D	0101		

Вывод

В процессе выполнения лабораторной работы был получен опыт работы с подпрограммами и стеком, разобрался, каким образом реализован стек в БЭВМ, изучил принцип действия команд PUSH, POP, CALL, RET.