

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Лабораторная работа №5

Дисциплина: Основы профессиональной деятельности

Вариант № 14299

Выполнил: Васильев Никита
Алексеевич, студент группы Р3108

Преподаватель: Вербовой Александр
Александрович

Санкт-Петербург 2024

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.

1. Программа осуществляет асинхронный ввод данных с ВУ-2
2. Программа начинается с адреса $4CB_{16}$. Размещаемая строка находится по адресу $5EC_{16}$.
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 00 (NUL). Стоп символ является обычным символом строки и подчиняется тем же правилам расположения в памяти что и другие символы строки.

Программа на ассемблере

```
                ORG    0x4CB
result:  WORD    0x05EC
stop:    WORD    0x0000
START:   CLA
s1:      IN      5
          AND     #0x40
          BEQ     s1
          IN      4
          ST      (result)
          PUSH
          CMP     stop
          BEQ     exit
          CLA
s2:      IN      5
          AND     #0x40
          BEQ     s2
          IN      4
          SWAB
          OR      &1
          ST      (result)
          SUB     &1
          SWAB
          CMP     stop
          BEQ     exit
          LD      (result)+
          POP
          CLA
          JUMP    s1
exit:    LD      (result)+
          HLT
```

Текст программы

Адрес	Код команды	Мнемоника	Описание
4CB	05EC	result	Ссылка на результат
4CC	0000	stop	Стоп-символ
4CD	+0200	CLA	Очистка аккумулятора
4CE	1205	IN 5	Чтение регистра состояния ВУ-2
4CF	2F40	AND #0x40	Проверка на наличие введенного символа
4D0	F0FD	BEQ IP – 3 (4CF)	Если нет – возвращение к регистру состояния
4D1	1204	IN 4	Чтение регистра данных ВУ-2
4D2	E8F8	ST (IP – 8) (4CB)	Сохранение символа в результат
4D3	0C00	PUSH	Сохранение символа в стек
4D4	7EF7	CMP IP – 9 (4CC)	Установить флаги по результату: AC – 4CC
4D5	F00F	BEQ IP + 15 (4E6)	Если стоп символ – выход
4D6	0200	CLA	Очистка аккумулятора
4D7	1205	IN 5	Чтение регистра состояния ВУ-2
4D8	2F40	AND #0x40	Проверка на наличие введенного символа
4D9	F0FD	BEQ IP – 9 (4D2)	Если нет – возвращение к регистру состояния
4DA	1204	IN 4	Чтение регистра данных ВУ-2
4DB	0680	SWAB	Обмен старшего и младшего байта
4DC	3C01	OR (SP + 1) (4CD)	Логическое «ИЛИ» SP + 1 AC → AC
4DD	E8ED	ST (IP – 19) (4CB)	Сохранение символа в результат
4DE	6C01	SUB (SP + 1)	Вычитание AC – SP + 1 → AC
4DF	0680	SWAB	Обмен старшего и младшего байта
4E0	7EEA	CMP IP – 22 (4CC)	Установить флаги по результату: AC – 4CC
4E1	F003	BEQ IP + 3 (4E6)	Если стоп символ – выход
4E2	AAE8	LD (IP – 24)+ (4CB)	Инкрементируем ссылку на результат
4E3	0800	POP	Очистка символа из стека
4E4	0200	CLA	Очистка аккумулятора
4E5	CEE9	JUMP IP – 23 (4D0)	Возвращаемся в начало цикла
4E6	AAE4	LD (IP – 30)+ (4CB)	Инкрементируем ссылку на результат
4E7	0100	HLT	Останов
...			
5EC	0000	0000	Строка результата

Описание программы

Программа осуществляет посимвольный асинхронный ввод данных в ВУ-2, посимвольно записывает их в память. Программа будет получать до тех пор, пока на ВУ-2 не будет передано стоп-слово с кодировкой 0x00, который она запишет в память и прекратит свое выполнение.

Передаваемое сообщение

Слово: ФОРТУНА

В кодировке КОИ-8: E6 EF F2 F4 F5 EE E1

Область представления и область допустимых значений

Область представления:

- *result* – целое 11-ти разрядное число;
- *stop* – целое 16-ти разрядное число;
- *temp* – 16-ти разрядная ячейка;
- 5EC – ? – 16-ти разрядная ячейка, хранящая два символа в кодировке КОИ-8.

Область допустимых значений:

- *result* $\in [5EC - 2047]$
- *temp* $\in [0; 255]$
- символы $\in [00; FF]$

Расположение в памяти ЭВМ программы, исходных данных и результатов

4CD – 4E7 – расположение команд программы;

4CB – 4CE – расположение переменных программы;

5EC – ? – расположение результата.

Адреса первой и последней выполняемых команд

4CD – адрес первой выполняемой команды;

4E7 – адрес последней выполняемой команды.

Таблица трассировки:

Слово: ФОРТУНА

В кодировке КОИ-8: E6 EF F2 F4 F5 EE E1

Таблица трассировки для первых двух символов.

Выполняемая команда		Содержимое регистров процессора								Ячейка, содержимое которой изменилось после выполнения команды	
										Адрес	Новый код
Адрес	Код	IP	CR	AR	DR	SP	BR	AC	NZVC		
4CD	0200	4CE	0200	4CD	0200	000	04CD	0000	0100		
4CE	1205	4CF	1205	4CE	1205	000	04CE	0000	0100		
4CF	2F40	4D0	2F40	4CF	0040	000	0040	0000	0100		

4D0	F0FD	4CE	F0FD	4D0	F0FD	000	FFFD	0000	0100		
4CE	1205	4CF	1205	4CE	1205	000	04CE	0000	0100		
4CF	2F40	4D0	2F40	4CF	0040	000	0040	0000	0100		
4D0	F0FD	4CE	F0FD	4D0	F0FD	000	FFFD	0000	0100		
4CE	1205	4CF	1205	4CE	1205	000	04CE	0000	0100		
4CF	2F40	4D0	2F40	4CF	0040	000	0040	0000	0100		
4D0	F0FD	4CE	F0FD	4D0	F0FD	000	FFFD	0000	0100		
4CE	1205	4CF	1205	4CE	1205	000	04CE	0040	0100		
4CF	2F40	4D0	2F40	4CF	0040	000	0040	0040	0000		
4D0	F0FD	4D1	F0FD	4D0	F0FD	000	FFFD	0040	0000		
4D1	1204	4D2	1204	4D1	1204	000	04D2	00E6	0000		
4D2	E8F8	4D3	E8F8	5EC	00E6	000	FFF7	00E6	0000	5EC	00E6
4D3	0C00	4D4	0C00	7FF	00E6	7FF	04D3	00E6	0000	7FF	00E6
4D4	7EF7	4D5	7EF7	4CC	000A	7FF	FFF6	00E6	0001		
4D5	F00F	4D6	F00F	4D5	F00F	7FF	04D5	00E6	0001		
4D6	0200	4D7	0200	4D6	0200	7FF	04D6	0000	0101		
4D7	1205	4D8	1205	4D7	1205	7FF	04D7	0000	0101		
4D8	2F40	4D9	2F40	4D8	0040	7FF	0040	0000	0101		
4D9	F0FD	4D7	F0FD	4D9	F0FD	7FF	FFFD	0000	0101		
4D7	1205	4D8	1205	4D7	1205	7FF	04D7	0040	0101		
4D8	2F40	4D9	2F40	4D8	0040	7FF	0040	0040	0001		
4D9	F0FD	4DA	F0FD	4D9	F0FD	7FF	04D9	0040	0001		
4DA	1204	4DB	1204	4DA	1204	7FF	04DA	00EF	0001		
4DB	0680	4DC	0680	4DB	0680	7FF	04DB	F000	1001		
4DC	3C01	4DD	3C01	7FF	00E6	7FF	0F30	F0E6	1001		
4DD	E8ED	4DE	E8ED	5EC	F0E6	7FF	FFEC	F0E6	1001	5EC	EFE6
4DE	6C01	4DF	6C01	7FF	00E6	7FF	FFED	F000	1001		
4DF	0680	4E0	0680	4DF	0680	7FF	0084	00EF	0001		
4E0	7EEB	4E1	7EEB	4CC	000A	7FF	FFEA	00EF	0001		
4E1	F003	4E2	F003	4E1	F003	7FF	0086	00EF	0001		
4E2	AAE8	088	AAE8	5EC	F0CF	7FF	FFE7	F0E6	0001		
4E3	0800	4E4	0800	7FF	00E6	000	017B	00E6	0000		
4E4	0200	4E5	0200	4E4	0200	000	0088	0000	0100		
4E5	CEE9	4CE	CEE9	4E5	4CE	000	FFE9	0000	0100		
4CE	1205	074	1205	073	1207	000	0073	0000	0100		
4CF	2F40	075	2F40	074	0040	000	0040	0000	0100		
4D0	F0FD	073	F0FD	075	F0FD	000	FFFD	0000	0100		
4CE	1205	074	1205	073	1207	000	0073	0040	0000		
4CF	2F40	075	2F40	074	0040	000	0040	0040	0000		
4D0	F0FD	076	F0FD	075	F0FD	000	0075	0040	0000		
4D1	1204	077	1204	076	1206	000	0076	0000	0100		
4D2	E8F8	078	E8F8	563	0000	000	FFF7	0000	0100	5ED	0000
4D3	0C00	079	0C00	7FF	0000	7FF	FFF8	0000	0100		
4D4	7EF7	07A	7EF7	070	0000	7FF	FFF6	0000	0100		
4D5	F00F	08A	F00F	07A	F00F	7FF	000F	0000	0100		
4E6	AAE5	08B	AAE5	563	000A	7FF	FFE4	0000	0100	4CB	5EE
4E7	0100	08C	0100	08B	0100	7FF	008B	0000	0100		

Вывод:

При выполнении данной лабораторной работы я познакомился с реализацией ввода-вывода в БЭВМ, ассемблером и кодировкой символов.