



УНИВЕРСИТЕТ ИТМО

Факультет ПИиКТ

Лабораторная работа №5 по Основам профессиональной
деятельности

Выполнение комплекса программ

Вариант 1102

Выполнил:

Давааням Баясгалан

группа Р3111

Преподаватель:

Саржевский Иван Анатольевич

г. Санкт-Петербург

2022 год

1. 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 464₁₆. Размещаемая строка находится по адресу 598₁₆.
3. Строка должна быть представлена в кодировке ISO-8859-5.
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.

2. Исходная программа

Адрес ячейки	Содержимое ячейки	Мnemonic	Описание
464 465	0598 000D	FADR STOP	Ссылка на строку. СТОП символ
466 467 468 469 46A 46B	1205 2F40 F0FD 1204 EC01 0A00	IN 5 AND #0x40 BEQ(IP-3) IN 4 ST(SP+1) RET	Ожидать сигнал готовности с ВУ-2 Считывать с ВУ-2 и сохраняем результат в стек
46C 46D 46E 46F 470 471 472 473	0200 0C00 DEF7 0800 7EF4 F00A 0680 E8F0	CLA PUSH CALL IP-9 POP CMP STOP BEQ IP+10 SWAB ST (FADR)	Считываем СИМВОЛ 1 Проверяем он СТОП символ или нет. Если он СТОП символ, то переходим в 47D Обмен символ и сохраняем.
474 475 476 477 478 479 47A 47C	0C00 DEF0 0800 7EED F003 48EA EAE9 CEF1	PUSH CALL IP-16 POP CMP STOP BEQ IP+3 ADD (FADR) ST (FADR)+ BR IP-15	Считываем СИМВОЛ 2 Проверяем он СТОП символ или нет. Если он СТОП символ, то переходим в 47D Добавим в конец строки, сохраняем и увеличиваем значение адреса. Цикл продолжает.
47D 47E 47F	48E6 EAE5 0100	ADD(FADR) ST(FADR)+ HLT	Если он СТОП символ, то его сохраняем и завершаем программу.

3. Описание программы

3.1 Назначение программы

Считываем символ с помощью ВУ-2 и записываем символы по порядку АДР1: СИМВ1 СИМВ2 АДР2: СИМВ3 СИМВ4 до СТОП символа. А СТОП символ - OD

3.2 Область представления

Ячейки результатов: 16 – разрядный знаковые:

3.3 Область допустимых значений

Ячейки результатов: $-2^{15} \leq x \leq 2^{15} - 1$

3.4 Расположение программы в памяти

Программа: 464...47F

Первый аргумент программы: 464

Второй аргумент программы: 465

Результат программы: 598...

3.5 Адреса первой и последней команды программы

Адрес первой команды: 464

Адрес последней команды: 47F

4. Код на ассемблере:

```
ORG 0x464
FADR: WORD 0x598
STOP: WORD 0xD
INPUT: IN 5
        AND    #0x40
        BEQ    INPUT
        IN     4
        ST     (SP+1)
        RET
START: CLA
FOR:   PUSH
        CALL   INPUT
        POP
        CMP    STOP
        BEQ    EXIT
        SWAB
        ST     (FADR)
        PUSH
        CALL   INPUT
        POP
        CMP    STOP
        BEQ    EXIT
        ADD    (FADR)
        ST     (FADR)+
        BR     FOR
EXIT:  ADD     (FADR)
        ST     (FADR)+
        HLT
```

Вывод

При выполнении лабораторной работы я познакомился с синтаксисом Ассемблера. Также я научился работать с ВУ-2, и освоил команды ввода и вывода.

Выполняемая команда		IP	Содержимое регистров процессора после выполнения команды								Ячейка, содержимое которой изменилось	
Адресс	Код		CR	AR	DR	SP	BR	AC	PS	NZVC	Адрес	Новый код
46C	0200	46D	0200	46C	0200	000	046C	0000	004	0100		
46D	0C00	46E	0C00	7FF	0000	7FF	046D	0000	004	0100		
46E	DEF7	466	DEF7	7FE	046F	7FE	0466	0000	004	0100	7FE	046F
466	1205	467	1205	466	1205	7FE	0466	0040	004	0100		
467	2F40	468	2F40	467	0040	7FE	0040	0040	000	0000		
468	F0FD	469	F0FD	468	F0FD	7FE	0468	0040	000	0000		
469	1204	46A	1204	469	1204	7FE	0469	0059	000	0000		
46A	EC01	46B	EC01	7FF	0059	7FE	0001	0059	000	0000	7FF	0059
46B	0A00	46F	0A00	7FE	046F	7FF	046B	0059	000	0000		
46F	0800	470	0800	7FF	0059	000	046F	0059	000	0000		
470	7EF4	471	7EF4	465	000D	000	FFF4	0059	001	0001		
471	F00A	472	F00A	471	F00A	000	0471	0059	001	0001		
472	0680	473	0680	472	0680	000	0472	5900	001	0001		
473	E8F0	474	E8F0	598	5900	000	FFF0	5900	001	0001	598	5900
474	0C00	475	0C00	7FF	5900	7FF	0474	5900	001	0001		
475	DEF0	466	DEF0	7FE	0476	7FE	0466	5900	001	0001	7FE	476
466	1205	467	1205	466	1205	7FE	0466	5940	001	0001		
467	2F40	468	2F40	467	0040	7FE	0040	0040	001	0001		
468	F0FD	469	F0FD	468	F0FD	7FE	0468	0040	001	0001		
469	1204	46A	1204	469	1204	7FE	0469	00EA	001	0001		
46A	EC01	46B	EC01	7FF	00EA	7FE	0001	00EA	001	0001	7FF	00EA
46B	0A00	476	0A00	7FE	0476	7FF	046B	00EA	001	0001		
476	0800	477	0800	7FF	00EA	000	0476	00EA	001	0001		
477	FEED	478	7EED	465	000D	000	FFED	00EA	001	0001		
478	F003	479	F003	478	F003	000	0478	0E0A	001	0001		
479	48EA	47A	48EA	598	5900	000	FFEA	59EA	000	0000		
47A	EAE9	47B	EAE9	598	59EA	000	FFE9	59EA	000	0000	598	59EA
47C	CEF1	46D	CEF1	47B	046D	000	FFF1	59EA	000	0000		
46D	0C00	46E	0C00	7FF	59EA	7FF	046D	59EA	000	0000	7FF	59EA
46E	DEF7	466	DEF7	7FE	046F	7FE	0466	59EA	000	0000	7FE	046F

466	1205	467	1205	466	1205	7FE	0466	5904	000	0000		
467	2F40	468	2F40	467	0400	7FE	0040	0040	000	0000		
468	F0FD	469	F0FD	468	F0FD	7FE	0468	0040	000	0000		
469	1204	46A	1204	469	1204	7FE	0469	000D	000	0000		
46A	EC01	46B	EC01	7FF	000D	7FE	0001	000D	000	0000	7FF	000D
46B	0A00	46F	0A00	7FE	046F	7FF	046B	000D	000	0000		
46F	0800	470	0800	7FF	000D	000	046F	000D	000	0000		
470	7EF4	471	7EF4	465	000D	000	FFF4	000D	005	0101		
471	F00A	47D	F00A	471	F00A	000	000A	000D	005	0101		
47D	48E6	47E	48E7	599	0000	000	FFE7	000D	000	0000	599	000D
47E	EAE5	47F	EAE5	599	000D	000	FFE6	000D	000	0000		
47F	0100	480	0100	47E	0100	000	047E	000D	000	0000		