

Университет ИТМО

Вариант №3253

ЛАБОРАТОРНАЯ РАБОТА №5

По дисциплине

Основы Профессиональной Деятельности

Преподаватель:

Ткешелашвили Нино Мерабиевна

Выполнила:

Жолеу Алтынай

Группа: Р3132

Санкт-Петербург

2021

Текст задания:

1. Программа осуществляет асинхронный ввод данных с ВУ-3
2. Программа начинается с адреса $1CF_{16}$. Размещаемая строка находится по адресу 628_{16} .
3. Строка должна быть представлена в кодировке ISO-8859-5.
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 0A (NL). Стоп символ является обычным символом строки и подчиняется тем же правилам расположения в памяти что и другие символы строки.

Выполнение:

1. Исходная программа:

Адрес	Код команды	Мнемоника	Комментарии
1CD	WORD	ADDR: WORD \$ARRAY	Адрес начала массива
1CE	WORD	I: WORD 0	Адрес текущего элемента
1CF	+0200	START: CLA	Очистка аккумулятора 0 -> AC
1D0	AEFC	LD ADDR	Загрузка (прямая относительная адресация) ADDR -> AC
1D1	EEFC	ST I	Сохранение (прямая относительная адресация) ADDR -> I
1D2	1205	S1: IN 7	Ввод с устройства SR#7 REG -> AC
1D3	2F40	AND #0x40	Логическое умножение с 6 битом M & AC → AC
1D4	F0FD	BEQ S1	Переход, если равенство (Z==1)
1D5	1204	IN 6	Ввод с устройства DR#6 (ввод символа) REG -> AC
1D6	7F0A	CMP #0x0A	Сравнение с кодом символа выхода из программы (прямая загрузка) AC – 0x0A
1D7	F00C	BEQ BREAK1	Переход, если равенство (Z==1)
1D8	0680	SWAB	Обмен ст. и мл. байтов: AC7...AC0 -> AC15...AC8

1D9	E8F4	ST (I)	Сохранение ввода в массив (косвенная относительная адресация) AC -> ARRAY[I]
1DA	1205	S2: IN 7	Ввод с устройства SR#7 REG -> AC
1DB	2F40	AND #0x40	Логическое умножение с 6 битом M & AC -> AC
1DC	F0FD	BEQ S2	Переход, если равенство (Z==1)
1DD	1204	IN 6	Ввод с устройства DR#6 (ввод символа) REG -> AC
1DE	7F0A	CMP #0x0A	Сравнение с кодом символа выхода из программы (прямая загрузка) AC – 0x0A
1DF	F006	BEQ BREAK2	Переход, если равенство (Z==1)
1E0	48ED	ADD (I)	Сложение (ко 2 символу добавляем 1) (косвенная относительная адресация) AC + ARRAY[I] -> AC
1E1	0680	SWAB	Обмен ст. и мл. байтов: AC7...AC0→AC15...AC8
1E2	E8EC	ST (I)	Сохранение ввода в массив (косвенная относительная адресация) AC→ ARRAY[I]
1E3	7AEA	CMP (I)+	Постинкрементация адреса текущего элемента массива
1E4	CEEE	JUMP S1	Переход на чтение первого символа
1E5	0680	BREAK1: SWAB	Обмен ст. и мл. байтов: AC7...AC0→AC15...AC8
1E6	CE01	JUMP SAVE	Переход к сохранению последнего ввода
1E7	48E7	BREAK2: ADD(I)	Сложение (ко 2 символу добавляем 1) (косвенная относительная адресация) AC+ARRAY[I]→ AC
1E8	E8E6	SAVE: ST(I)	Сохранение ввода в массив (косвенная относительная адресация)
1E9	0100	HLT	Останов
628		ARRAY: WORD?	

2. Программа на Ассемблере:

```
ORG 0x1CD
ADDR: WORD $ARRAY
I: WORD 0
```

```
START: CLA
LD ADDR
ST I
S1:      IN 7
AND #0x40
BEQ S1
IN 6
CMP #0x0A
BEQ BREAK1
SWAB
ST (I)
```

```
S2:      IN 7
AND #0x40
BEQ S2
IN 6
CMP #0x0A
BEQ BREAK2
ADD (I)
SWAB
```

```
ST (I)
CMP (I)+
JUMP S1
```

```
BREAK1:
JUMP SAVE
BREAK2:      ADD (I)
SWAB
SAVE:  ST (I)
      HLT
```

ORG 0x628

ARRAY: WORD ?

3. Описание программы:

- Расположение в памяти ЭВМ программы, исходных данных и результатов

команды:	от 1CF до 1E9
размещаемая строка:	от 628
код символа завершающего ввод строки:	0A
размещаемая строка:	м
кодировка:	ISO-8859-5

м-DC

- Адреса первой и последней выполняемых команд программы

Адрес первой команды:	1CF
Адрес последней команды:	1E9

Вывод: изучила организации системы ввода-вывода базовой ЭВМ, команды ввода-вывода и исследовала процесс функционирования ЭВМ при обмене данными по сигналам готовности внешних устройств.

break no 1 , in 2 (1simvol) swab .s