

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Санкт-Петербургский национальный исследовательский университет информационных
технологий, механики и оптики»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №3

по дисциплине

«Основы профессиональной деятельности»

Вариант № 3104

Выполнил:

Студент группы Р3131

Дворкин Борис

Александрович

Преподаватель:

Клименков Сергей

Викторович

Содержание

Текст задания	3
Описание программы	4
Таблица трассировки	5
Вывод	6

Текст задания

Восстановить текст заданного варианта программы, определить предназначение и описание программы, определить область представления и ОДЗ исходных данных и результата, выполнить трассировку программы.

2DE:	02F5	2EC:	F401
2DF:	A000	2ED:	CE04
2E0:	E000	2EE:	0400
2E1:	E000	2EF:	7EF1
2E2:	+ AF40	2F0:	F801
2E3:	0680	2F1:	EEEF
2E4:	0500	2F2:	82E0
2E5:	EEFB	2F3:	CEF6
2E6:	AF05	2F4:	0100
2E7:	EEF8	2F5:	0B01
2E8:	AEF5	2F6:	F200
2E9:	EEF5	2F7:	0580
2EA:	AAF4	2F8:	0000
2EB:	0480	2F9:	0480

Адрес	Код команды	Мнемоника	Комментарии
2DE	02F5	-- arr_first_elem	Адрес первого элемента
2DF	A000	-- arr_last_elem	Адрес текущего элемента (начиная с последнего)
2E0	E000	-- arr_length	Количество элементов массива
2E1	E000	-- result	Результат
2E2	AF40	LD #0x40	Прямая загрузка 040 -> AC
2E3	0680	SWAB	Обмен ст. и мл. байтов в AC
2E4	0500	ASL	Арифметический сдвиг AC влево
2E5	EEFB	ST IP-5	Прямое относительное сохранение AC -> M (2E1)
2E6	AF05	LD #0x05	Прямая загрузка 005 -> AC
2E7	EEF8	ST IP-8	Прямое относительное сохранение AC -> M (2E0)
2E8	AEF5	LD IP-B	Прямая относительная загрузка 02F5 -> AC
2E9	EEF5	ST IP-B	Прямое относительное сохранение AC -> M (2DE)
2EA	AAF4	LD (IP-C)+	Косвенная относительная автоинкрементная загрузка: 3н(2DE) += 1; 3н(2DE) -> AC
2EB	0480	ROR	Циклический сдвиг AC вправо
2EC	F401	BCS IP+1	Если C==1, то IP+1+1 -> IP
2ED	CE04	BR IP+4	IP+4+1 -> IP
2EE	0400	ROL	Циклический сдвиг AC влево
2EF	7EF1	CMP IP-F	Прямое относительное сравнение AC-M(2E1)
2F0	F801	BLT IP+1	Если N⊕V==1, то IP+1+1 -> IP
2F1	EEEF	ST IP-11	Прямое относительное сохранение AC -> M (2E0)
2F2	82E0	LOOP 0x2E0	M(2E0) - 1 -> M(2E0); Если (2E0) <= 0, то IP + 1 -> IP
2F3	CEF6	BR IP-A	Переход IP-A+1 -> IP
2F4	0100	HLT	Останов
2F5	0B01	...	“Элементы массива”
2F6	F200	...	
2F7	0580	...	
2F8	0000	...	
2F9	0480	...	

Описание программы

Программа находит максимальный нечётный элемент массива и сохраняет информацию о нём в биты ячейки результата. Формула результата:

$$\text{MEM}(2E1) = \sum_{i=0}^{\text{MEM}(2E0)} \begin{cases} 2^i & \text{если } \text{MEM}(2F5 + i) \div 2 \\ 0 & \text{если } \text{MEM}(2F5 + i) \nmid 2 \end{cases}$$

Область представления

- arr_first_elem, arr_last_elem, arr_length, result – 16-ти разрядные целые числа в прямом коде.
- arr[i] – 16-ти разрядные целые числа в дополнительном коде

Область допустимых значений

- arr_length $\in [1; 16]$ (т. к. при arr_length > 16 битов результата не будет хватать для данных о делимости элементах и он будет ошибочным)
- result $\in [0; 2^{16} - 1]$ (т. к. max = 1 + 2 + 4 + ... + 2¹⁵)
- arr_first_elem $\in [0; 2DE - \text{arr_length}] \cup [2F5; 7FF - \text{arr_length}]$
- arr_last_elem $\in [\text{arr_first_elem}; \text{arr_first_elem} + \text{arr_length} - 1]$
- Элементы массива arr[i] $\in [-32768; 32767]$ (т. е. [-2¹⁵; 2¹⁵-1])

Расположение данных в памяти

- 2DE, 2DF, 2E0, 2F5, 2F6, 2F7, 2F8, 2F9 – исходные данные;
- 2DE – промежуточный результат;
- 2E1 – итоговый результат;
- 2E2 – 2F4 – команды.

Адреса первой и последней выполняемой команды

- Адрес первой команды: 2E2
- Адрес последней команды: 2F4

Таблица трассировки

Значения:

Arr[0] = 0xFFFF, Arr[1] = 0xEDAA, Arr[2] = 0x0771, Arr[3] = 0xBAAB, Arr[4] = 0x6666, arr_length = 5, arr_first_elem = 0x02F5

Адр	Знач	IP	CR	AR	DR	SP	BR	AC	PS	NZVC	Адр	Знач
2E2	AF40	2E2	0000	000	0000	000	0000	0000	004	0100		
2E2	AF40	2E3	AF40	2E2	0040	000	0040	0040	000	0000		
2E3	0680	2E4	0680	2E3	0680	000	02E3	4000	000	0000		
2E4	0500	2E5	0500	2E4	4000	000	02E4	8000	00A	1010		
2E5	EEFB	2E6	EEFB	2E1	8000	000	FFFB	8000	00A	1010	2E1	8000
2E6	AF05	2E7	AF05	2E6	0005	000	0005	0005	000	0000		
2E7	EEF8	2E8	EEF8	2E0	0005	000	FFF8	0005	000	0000	2E0	0005
2E8	AEF5	2E9	AEF5	2D	02F5	000	FFF5	02F5	000	0000		
2E9	EEF5	2E	EEF5	2DF	02F5	000	FFF5	02F5	000	0000	2DF	02F5
2E	AAF	2EB	AAF	2F5	FFFF	000	FFF4	FFFF	008	1000	2DF	02F6
A	4		4									
2EB	0480	2EC	0480	2EB	0480	000	02EB	7FFF	003	0011		
2EC	F401	2EE	F401	2EC	F401	000	0001	7FFF	003	0011		
2EE	0400	2EF	0400	2EE	0400	000	02EE	FFFF	00A	1010		
2EF	7EF1	2F0	7EF1	2E1	8000	000	FFF1	FFFF	001	0001		
2F0	F801	2F1	F801	2F0	F801	000	02F0	FFFF	001	0001		
2F1	EEEE	2F2	EEEE	2E1	FFFF	000	FFE	FFFF	001	0001	2E1	FFFF
2F2	82E0	2F3	82E0	2E0	0004	000	0003	FFFF	001	0001	2E0	0004
2F3	CEF6	2E	CEF6	2F3	02EA	000	FFF6	FFFF	001	0001		
		A										
2E	AAF	2EB	AAF	2F6	EDAA	000	FFF4	EDAA	009	1001	2DF	02F7
A	4		4									
2EB	0480	2EC	0480	2EB	0480	000	02EB	F6D5	00A	1010		
2EC	F401	2E	F401	2EC	F401	000	02EC	F6D5	00A	1010		
		D										
2E	CE04	2F2	CE04	2E	02F2	000	0004	F6D5	00A	1010		
D				D								
2F2	82E0	2F3	82E0	2E0	0003	000	0002	F6D5	00A	1010	2E0	0003
2F3	CEF6	2E	CEF6	2F3	02EA	000	FFF6	F6D5	00A	1010		
		A										
2E	AAF	2EB	AAF	2F7	0771	000	FFF4	0771	000	0000	2DF	02F8
A	4		4									
2EB	0480	2EC	0480	2EB	0480	000	02EB	03B8	003	0011		
2EC	F401	2EE	F401	2EC	F401	000	0001	03B8	003	0011		
2EE	0400	2EF	0400	2EE	0400	000	02EE	0771	000	0000		
2EF	7EF1	2F0	7EF1	2E1	FFFF	000	FFF1	0771	000	0000		
2F0	F801	2F1	F801	2F0	F801	000	02F0	0771	000	0000		
2F1	EEEE	2F2	EEEE	2E1	0771	000	FFE	0771	000	0000	2E1	0778
2F2	82E0	2F3	82E0	2E0	0002	000	0001	0771	000	0000	2E0	0002
2F3	CEF6	2E	CEF6	2F3	02EA	000	FFF6	0771	000	0000		
		A										
2E	AAF	2EB	AAF	2F8	BAAB	000	FFF4	BAAB	008	1000	2DF	02F9
A	4		4									
2EB	0480	2EC	0480	2EB	0480	000	02EB	5D55	003	0011		
2EC	F401	2EE	F401	2EC	F401	000	0001	5D55	003	0011		
2EE	0400	2EF	0400	2EE	0400	000	02EE	BAAB	00A	1010		
2EF	7EF1	2F0	7EF1	2E1	0771	000	FFF1	BAAB	009	1001		
2F0	F801	2F2	F801	2F0	F801	000	0001	BAAB	009	1001		
2F2	82E0	2F3	82E0	2E0	0001	000	0000	BAAB	009	1001	2E0	0001
2F3	CEF6	2E	CEF6	2F3	02EA	000	FFF6	BAAB	009	1001		
		A										
2E	AAF	2EB	AAF	2F9	6666	000	FFF4	6666	001	0001	2DF	02FA
A	4		4									

2EB	0480	2EC	0480	2EB	0480	000	02EB	B333	00A	1010		
2EC	F401	2E	F401	2EC	F401	000	02EC	B333	00A	1010		
		D										
2E	CE04	2F2	CE04	2E	02F2	000	0004	B333	00A	1010		
				D								
2F2	82E0	2F4	82E0	2E0	0000	000	FFFF	B333	00A	1010	2E0	0000
2F4	0100	2F5	0100	2F4	0100	000	02F4	B333	00A	1010		

Вывод

Во время выполнения лабораторной работы я научился работать в БЭВМ с массивами, ветвлением и циклами. Я изучил прямую и косвенную адресацию и цикл выполнения таких команд, как LOOP и JUMP.