

Санкт-Петербургский Национальный Исследовательский
Университет ИТМО
Факультет программной инженерии и компьютерной техники

Лабораторная работа №3
По «Основы Профессиональной Деятельности»
Вариант 43052

Выполнил:
Студент группы Р3106
Рубцов Арсений Дмитриевич
Проверил:
Вербовой Александр Александрович

Оглавление

Текст задания.....	3
Описание программы.....	4
Область представления.....	5
Область допустимых значений.....	5
Расположение данных в памяти.....	5
Таблица трассировки.....	6
Вывод.....	7

Текст задания

По выданному преподавателем варианту восстановить текст заданного варианта программы, определить предназначение и составить описание программы, определить область представления и область допустимых значений исходных данных и результата, выполнить трассировку программы.

2B2:	02C4		2C0:	2AF4
2B3:	0200		2C1:	82B4
2B4:	E000		2C2:	CEF9
2B5:	E000		2C3:	0100
2B6:	+ 0200		2C4:	0380
2B7:	EEFD		2C5:	A2B6
2B8:	AF05		2C6:	0E01
2B9:	EEFA		2C7:	0000
2BA:	4EF7		2C8:	0380
2BB:	EEF7			
2BC:	ABF6			
2BD:	0480			
2BE:	F402			
2BF:	0400			

Адрес	Код команды	Мнемоника	Комментарии	Описание программы
2B2	02C4	A	Адрес первого элемента массива	-
2B3	0200	B	Адрес текущего элемента массива	-
2B4	E000	C	Длина массива	-
2B5	E000	R	Счетчик	Результат
2B6	0200	CLA	Очистка AC; AC = 0	
2B7	EEFD	ST IP - 3	Прямая относительная $ST\ IP + 1 - 3 = ST\ 2B5$	Загружаем значение AC = 0000 в ячейку 2B5; $MEM(2B5) = 0000$
2B8	AF05	LD 05	Прямая загрузка AC = 0005	В аккумулятор загружаем значение 0005
2B9	EEFA	ST IP - 6	Прямая относительная $ST\ IP + 1 - 6 = ST\ 2B4$	Записываем значение AC = 0005 в ячейку с адресом 2B4
2BA	4EF7	ADD IP - 9	Прямая относительная $ADD\ IP + 1 - 9 = ADD\ 2B2$	Складываем AC и значение ячейки 2B2 и записываем в AC; AC = 02C9
2BB	EEF7	ST IP - 9	Прямая относительная	Записываем значение AC =

			ST IP + 1 - 9 = ST 2B3	02C9 в ячейку с адресом 2B3
2BC	ABF6	LD --MEM(2B3)	Косвенная автодекрементная; LD --MEM(2B3)	Записали в AR = IP + 1 - 10 Ячейку 2B3 уменьшили на 1, после загрузили адрес DR(2B3) - 1
2BD	0480	ROR	Циклический сдвиг вправо	$AC_0 \rightarrow C$; $C \rightarrow AC_{15}$
2BE	F402	BCC 2C1	Если C=0, то IP + 2 + 1 → IP	Если при циклическом сдвиге элемента массива, флаг Carry устанавливается в 0, то переходим на ячейку 2C1, в противном случае на ячейку 2BF
2BF	0400	ROL	Циклический сдвиг влево	$AC_{15} \rightarrow C$; $C \rightarrow AC_0$
2C0	2AF4	AND --MEM(2B5)	Косвенная автодекрементная; AND --MEM(2B5)	Увеличиваем счетчик на 1
2C1	82B4	LOOP 2B4		Повторяем программу с ячейки 2BC, пока не пройдем все элементы массива
2C2	CEF9	JUMP IP - 7	Прямая относительная JUMP IP + 1 - 7	
2C3	0100	HLT	ОСТАНОВ	
2C4	0380	D	Элементы массива	-
2C5	A2B6	E		-
2C6	0E01	F		-
2C7	0000	G		-
2C8	0380	H		-

Описание программы

Программа находит четные элементы массива и увеличивает ячейку, отвечающую за количество таких элементов.

Область представления

- A, B — 11-ти разрядные, адрес в памяти БЭВМ
- C, R — 16-ти разрядные целые числа, беззнаковые.
- D, E, F, G, H — 16-ти разрядные беззнаковые целые числа.

Область допустимых значений

- **ОДЗ для количества элементов массива** можно получить из инструкции с прямой загрузкой операнда (2B8) AFxx, где xx будет записан в AC (знак расширен). Так как количество элементов будет проходить через инструкцию LOOP => необходимо сделать указанное количество итераций, а если число, записанное в младшие 7 разрядов будет отрицательным, то цикл LOOP сразу же закончится и будет вызвана инструкция HLT (Останов). Тогда максимальное количество элементов в массиве соответствует числу $7F = 127$
- **ОДЗ для результата программы(2B5):** так как результат представлен в виде счетчика (беззнаковое число) и из предыдущего условия следует, что массив может содержать 127 элементов, то и ячейка результата содержится в диапазоне: $[0; 127]$
- **ОДЗ для элементов массива:** Так как элементы массива представлены в виде беззнаковых 16-ти разрядных чисел, то ОДЗ для них: $[0; 2^{16}]$

Расположение данных в памяти

- 2B6 — 2C3 — исполняемая программа, инструкции
- 2B5 — результат
- 2C4 — 2C8 — элементы массива
- 2B2 — 2B4 — адрес первого и текущего элемента массива и его длина

Таблица трассировки

I = 3

D = 3138

E = DEAD

F = B1BA

Адрес	Знач	IP	CR	AR	DR	SP	BR	AC	PS	NZVC	Адрес	Знач
2B6	0200	2B7	0200	2B6	0200	000	02B6	0000	004	0100		
2B7	EEFD	2B8	EEFD	2B5	0000	000	FFFD	0000	004	0100	2B5	0000
2B8	AF03	2B9	AF03	2B8	0003	000	0003	0003	000	0000		
2B9	EEFA	2BA	EEFA	2B4	0003	000	FFFA	0003	000	0000	2B4	0003
2BA	4EF7	2BB	4EF7	2B2	02C4	000	FFF7	02C7	000	0000		
2BB	EEF7	2BC	EEF7	2B3	02C7	000	FFF7	02C7	000	0000	2B3	02C7
2BC	ABF6	2BD	ABF6	2C6	B1BA	000	FFF6	B1BA	008	1000	2B3	02C6
2BD	0480	2BE	0480	2BD	0480	000	02BD	58DD	000	0000		
2BE	F402	2BF	F402	2BE	F402	000	02BE	58DD	000	0000		
2BF	0400	2C0	0400	2BF	0400	000	02BF	B1BA	00A	1010		
2C0	2AF4	2C1	2AF4	000	0000	000	FFF4	0000	004	0100	2B5	0001
2C1	82B4	2C2	82B4	2B4	0002	000	0001	0000	004	0100	2B4	0002
2C2	CEF9	2C3	CEF9	2C2	02BC	000	FFF9	0000	004	0100		
2BC	ABF6	2BD	ABF6	2C5	DEAD	000	FFF6	DEAD	008	1000	2B3	02C5
2BD	0480	2BE	0480	2BD	0480	000	02BD	6F56	003	0011		
2BE	F402	2BF	F402	2BE	F402	000	0002	6F56	003	0011		
2C1	82B4	2C2	82B4	2B4	0001	000	0000	6F56	003	0011	2B4	0001
2C2	CEF9	2C3	CEF9	2C2	02BC	000	FFF9	6F56	003	0011		
2BC	ABF6	2BD	ABF6	2C4	3138	000	FFF6	3138	001	0001	2B3	02C4
2BD	0480	2BE	0480	2BD	0480	000	02BD	989C	00A	1010		
2BE	F402	2BF	F402	2BE	F402	000	02BE	989C	00A	1010		
2BF	0400	2C0	0400	2BF	0400	000	02BF	3138	003	0011		
2C0	2AF4	2C1	2AF4	001	0000	000	FFF4	0000	005	0101	2B5	0002
2C1	82B4	2C2	82B4	2B4	0000	000	FFFF	0000	005	0101	2B4	0000
2C3	0100	2C3	0100	2C3	0100	000	02C3	0000	005	0101		

Вывод

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