

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ И СЕТЕЙ

ОЦЕНКА ОТЧЁТА:

РУКОВОДИТЕЛЬ:

Старший преподаватель		Д.В.Куртяник
должность, уч. степень, звание	подпись, дата	инициалы, фамилия

ОТЧЁТ О ЛАБОРАТОРНОЙ РАБОТЕ № 2

по курсу: ПРОГРАММИРОВАНИЕ НА ЯЗЫКАХ АССЕМБЛЕРА

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №	4143		Е.Д.Тегай
		подпись, дата	инициалы, фамилия

Цель работы

Изучение концепций и освоение технологии программирования 16-битных приложений на языке ассемблера для архитектуры процессоров семейства Intel. Закрепление навыков работы с отладчиком Turbo Debugger.

Индивидуальное задание

Индивидуальное задание изображено на рисунке 1.

18. Задан массив чисел размером в байт. Найти максимальный элемент массива.

Рисунок 1 – Индивидуальное задание

Ход работы

Для начала задаётся сам массив типа **byte**, состоящий в данном случае из 16 элементов, его длина и переменная **max**, которая и будет хранить в себе значение максимального элемента.

После этого загружается адрес начала массива с помощью команды **lea** в индексный регистр **si**. Также регистрам общего назначения присваиваются некоторые значения: 0 регистру **bx** – текущий временный индекс, длина массива – регистру **cx**, 0 регистру **dx** – индекс максимального значения.

Далее регистру **al** присваивается начальный элемент массива, а затем сразу даем переменной **max** значение начального элемента массива.

После этого двигаем адрес на следующий элемент массива с помощью записи **inc si**, где команда **inc** увеличивает **si** на единицу. Соответственно, уменьшается общее количество оставшихся элементов массива, поэтому значение **cx** уменьшается на единицу с помощью команды **dec**. Далее увеличивается текущий индекс на единицу также с помощью команды **inc**.

Затем нужно сделать проверку «состоит ли массив всего лишь из одного элемента». Это делается с помощью команды **jcxz**, которая служит для выполнения перехода в зависимости от состояния регистра **cx**. Так, если по итогу регистр **cx** хранит в себе ноль, то выполняется выход из программы с помощью метки **exit**.

Затем программа переходит в метку **search** (поиск максимального элемента массива). Ход ее работы описан далее:

Загружается текущий элемент массива в регистр **al**. Далее происходит сравнение значений переменной **max** и регистра **al**, то бишь сравнение текущего и максимального. Осуществляется это с помощью команды **cmp**. Если текущий элемент меньше или равен максимальному, то с помощью метки **next** пропускаем следующие шаги. Это выполняется с помощью команды **jle**.

Если же текущий элемент больше максимального, то текущий элемент делается максимальным (строка **max, al**), также сохраняется текущий номер в массиве. Это достигается с помощью строчки **dx, bx**.

Как было описано ранее, если текущий элемент меньше или равен максимальному, то осуществляется переход к метке **next**. Алгоритм ее работы таков: текущий индекс увеличивается, а адрес «двигается» до следующего элемента.

Таким образом с помощью функции **loop** блок кода внутри метки **search** будет выполняться столько раз, пока не будут пройдены все элементы массива.

Также ранее упоминалась метка **exit**. Содержимое внутри нее осуществляет выход из программы.

Текст программы

```
.8086
.model small
.stack 100h
.data
arr byte -50,-1,124,127,39,27,-20,-90,108,55,-111, 48,-119,86,18,3
len equ $ - arr
max byte ?

.code
Start:
```

;инициализация сегмента данных

mov ax,@data

mov ds,ax

lea si,arr ;загрузка адреса начала массива

mov bx,0 ;текущий временный индекс

mov cx,len ;длина массива

mov dx,0 ;индекс максимального элемента

mov al,[si] ;загрузка начального элемента массива

mov max,al ;сразу начальный элемент делается максимальным

inc si ;"движение" адреса на следующий элемент массива

dec cx ;уменьшение общего кол-ва оставшихся эл-тов

inc bx ;увеличение текущего индекса

jcxz exit ;проверка "состоит ли массив из одного эл-та?"

search: ;цикл поиска максимального элемента

mov al,[si] ;загрузка текущего элемента массива

cmp al,max ;сравнение текущего элемента с максимальным

jle next ;если текущий меньше или равен, то пропуск

mov max,al ;в ином случае делаем текущий элемент максимальным

mov dx,bx ;сохранение текущего номера в массиве

next:

inc bx ;увеличение текущего индекса

inc si ;"движение" адреса до следующего элемента

loop search ;повторение итераций, пока не пройдены все эл-ты

exit:

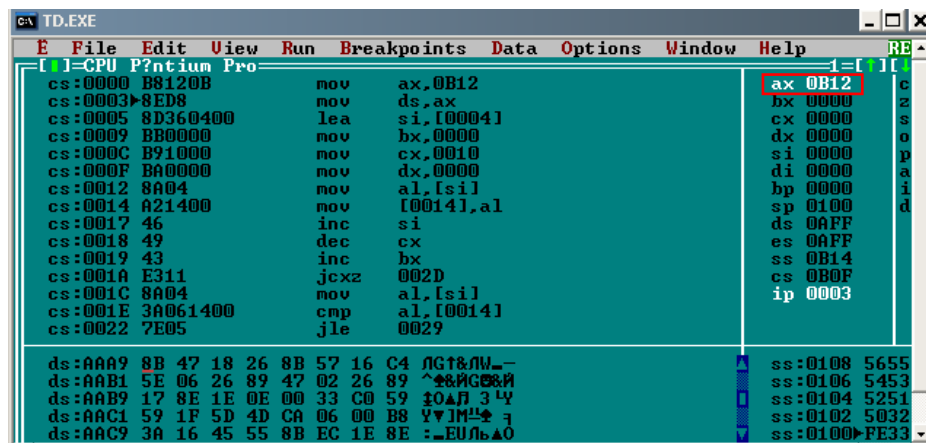
mov ah,4Ch ;функция выхода из программы

mov al,0 ;код возврата

int 21h ;прерывание доса, чтобы функция выполнялась

end Start ;конец программы, указывается точка входа

Результаты работы



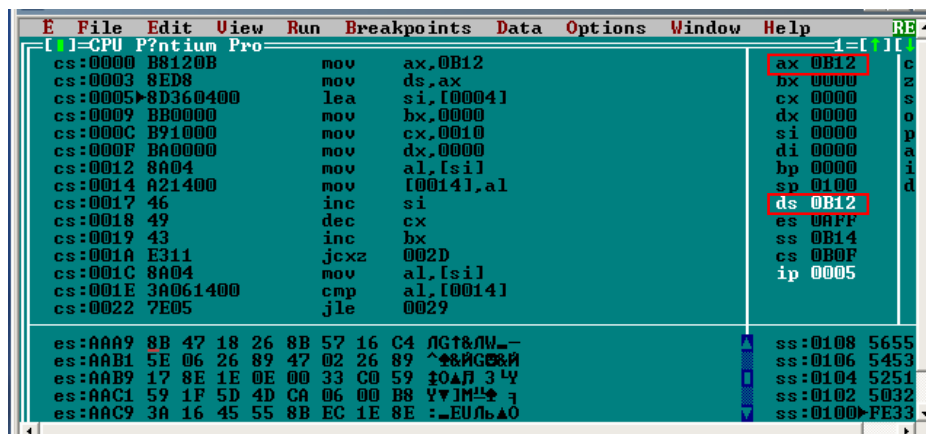
The screenshot shows a debugger window with the following assembly code and register values:

Address	Disassembly	Comment
cs:0000	B8120B	mov ax,0B12
cs:0003	8ED8	mov ds,ax
cs:0005	8D360400	lea si,[0004]
cs:0009	BB0000	mov bx,0000
cs:000C	B91000	mov cx,0010
cs:000F	BA0000	mov dx,0000
cs:0012	8A04	mov al,[si]
cs:0014	A21400	mov [0014],al
cs:0017	46	inc si
cs:0018	49	dec cx
cs:0019	43	inc bx
cs:001A	E311	jcxz 002D
cs:001C	8A04	mov al,[si]
cs:001E	3A061400	cmp al,[0014]
cs:0022	7E05	jle 0029

Register values (right side of the window):

Register	Value
ax	0B12
bx	0000
cx	0000
dx	0000
si	0000
di	0000
bp	0000
sp	0100
ds	0AFF
es	0AFF
ss	0B14
cs	0B0F
ip	0003

Рисунок 1 – Передача в ax номера сегмента, где записаны данные



The screenshot shows the same assembly code as Figure 1, but with different register values:

Address	Disassembly	Comment
cs:0000	B8120B	mov ax,0B12
cs:0003	8ED8	mov ds,ax
cs:0005	8D360400	lea si,[0004]
cs:0009	BB0000	mov bx,0000
cs:000C	B91000	mov cx,0010
cs:000F	BA0000	mov dx,0000
cs:0012	8A04	mov al,[si]
cs:0014	A21400	mov [0014],al
cs:0017	46	inc si
cs:0018	49	dec cx
cs:0019	43	inc bx
cs:001A	E311	jcxz 002D
cs:001C	8A04	mov al,[si]
cs:001E	3A061400	cmp al,[0014]
cs:0022	7E05	jle 0029

Register values (right side of the window):

Register	Value
ax	0B12
bx	0000
cx	0000
dx	0000
si	0000
di	0000
bp	0000
sp	0100
ds	0B12
es	0AFF
ss	0B14
cs	0B0F
ip	0005

Рисунок 2 – Передача в ds значения ax

E File Edit View Run Breakpoints Data Options Window Help																																		
[] CPU P?ntium Pro																																		
cs:0000	B8120B	mov	ax,0B12	ax	0B12																													
cs:0003	8ED8	mov	ds,ax	bx	0000																													
cs:0005	8D360400	lea	si,[0004]	cx	0000																													
cs:0009	BB0000	mov	bx,0000	dx	0000																													
cs:000C	B91000	mov	cx,0010	si	0004																													
cs:000F	BA0000	mov	dx,0000	di	0000																													
cs:0012	8A04	mov	al,[si]	bp	0000																													
cs:0014	A21400	mov	[0014],al	sp	0100																													
cs:0017	46	inc	si	ds	0B12																													
cs:0018	49	dec	cx	es	0AFF																													
cs:0019	43	inc	bx	ss	0B14																													
cs:001A	E311	jcxz	002D	cs	0B0F																													
cs:001C	8A04	mov	al,[si]	ip	0009																													
cs:001E	3A061400	cmp	al,[0014]																															
cs:0022	7E05	jle	0029																															
<table border="0"> <tr> <td>es:AAA9</td> <td>8B 47 18 26 8B 57 16 C4</td> <td>лГt&лW-</td> <td>ss:0108</td> <td>5655</td> </tr> <tr> <td>es:AAB1</td> <td>5E 06 26 89 47 02 26 89</td> <td>^&&ИG&&И</td> <td>ss:0106</td> <td>5453</td> </tr> <tr> <td>es:AAB9</td> <td>17 8E 1E 0E 00 33 C0 59</td> <td>т0Δл 3 4</td> <td>ss:0104</td> <td>5251</td> </tr> <tr> <td>es:AAC1</td> <td>59 1F 5D 4D CA 06 00 B8</td> <td>Y∇JИ</td> <td>ss:0102</td> <td>5032</td> </tr> <tr> <td>es:AAC9</td> <td>3A 16 45 55 8B EC 1E 8E</td> <td>:EУлбΔ0</td> <td>ss:0100</td> <td>FE33</td> </tr> </table>										es:AAA9	8B 47 18 26 8B 57 16 C4	лГt&лW-	ss:0108	5655	es:AAB1	5E 06 26 89 47 02 26 89	^&&ИG&&И	ss:0106	5453	es:AAB9	17 8E 1E 0E 00 33 C0 59	т0Δл 3 4	ss:0104	5251	es:AAC1	59 1F 5D 4D CA 06 00 B8	Y∇JИ	ss:0102	5032	es:AAC9	3A 16 45 55 8B EC 1E 8E	:EУлбΔ0	ss:0100	FE33
es:AAA9	8B 47 18 26 8B 57 16 C4	лГt&лW-	ss:0108	5655																														
es:AAB1	5E 06 26 89 47 02 26 89	^&&ИG&&И	ss:0106	5453																														
es:AAB9	17 8E 1E 0E 00 33 C0 59	т0Δл 3 4	ss:0104	5251																														
es:AAC1	59 1F 5D 4D CA 06 00 B8	Y∇JИ	ss:0102	5032																														
es:AAC9	3A 16 45 55 8B EC 1E 8E	:EУлбΔ0	ss:0100	FE33																														

Рисунок 3 – Передача в si адреса начала массива

E File Edit View Run Breakpoints Data Options Window Help																																		
[] CPU P?ntium Pro																																		
cs:0000	B8120B	mov	ax,0B12	ax	0B12																													
cs:0003	8ED8	mov	ds,ax	bx	0000																													
cs:0005	8D360400	lea	si,[0004]	cx	0000																													
cs:0009	BB0000	mov	bx,0000	dx	0000																													
cs:000C	B91000	mov	cx,0010	si	0004																													
cs:000F	BA0000	mov	dx,0000	di	0000																													
cs:0012	8A04	mov	al,[si]	bp	0000																													
cs:0014	A21400	mov	[0014],al	sp	0100																													
cs:0017	46	inc	si	ds	0B12																													
cs:0018	49	dec	cx	es	0AFF																													
cs:0019	43	inc	bx	ss	0B14																													
cs:001A	E311	jcxz	002D	cs	0B0F																													
cs:001C	8A04	mov	al,[si]	ip	000C																													
cs:001E	3A061400	cmp	al,[0014]																															
cs:0022	7E05	jle	0029																															
<table border="0"> <tr> <td>es:AAA9</td> <td>8B 47 18 26 8B 57 16 C4</td> <td>лГt&лW-</td> <td>ss:0108</td> <td>5655</td> </tr> <tr> <td>es:AAB1</td> <td>5E 06 26 89 47 02 26 89</td> <td>^&&ИG&&И</td> <td>ss:0106</td> <td>5453</td> </tr> <tr> <td>es:AAB9</td> <td>17 8E 1E 0E 00 33 C0 59</td> <td>т0Δл 3 4</td> <td>ss:0104</td> <td>5251</td> </tr> <tr> <td>es:AAC1</td> <td>59 1F 5D 4D CA 06 00 B8</td> <td>Y∇JИ</td> <td>ss:0102</td> <td>5032</td> </tr> <tr> <td>es:AAC9</td> <td>3A 16 45 55 8B EC 1E 8E</td> <td>:EУлбΔ0</td> <td>ss:0100</td> <td>FE33</td> </tr> </table>										es:AAA9	8B 47 18 26 8B 57 16 C4	лГt&лW-	ss:0108	5655	es:AAB1	5E 06 26 89 47 02 26 89	^&&ИG&&И	ss:0106	5453	es:AAB9	17 8E 1E 0E 00 33 C0 59	т0Δл 3 4	ss:0104	5251	es:AAC1	59 1F 5D 4D CA 06 00 B8	Y∇JИ	ss:0102	5032	es:AAC9	3A 16 45 55 8B EC 1E 8E	:EУлбΔ0	ss:0100	FE33
es:AAA9	8B 47 18 26 8B 57 16 C4	лГt&лW-	ss:0108	5655																														
es:AAB1	5E 06 26 89 47 02 26 89	^&&ИG&&И	ss:0106	5453																														
es:AAB9	17 8E 1E 0E 00 33 C0 59	т0Δл 3 4	ss:0104	5251																														
es:AAC1	59 1F 5D 4D CA 06 00 B8	Y∇JИ	ss:0102	5032																														
es:AAC9	3A 16 45 55 8B EC 1E 8E	:EУлбΔ0	ss:0100	FE33																														

Рисунок 4 – Передача в bx нуля (текущий временный индекс)

E File Edit View Run Breakpoints Data Options Window Help																																		
[] CPU P?ntium Pro																																		
cs:0000	B8120B	mov	ax,0B12	ax	0B12																													
cs:0003	8ED8	mov	ds,ax	bx	0000																													
cs:0005	8D360400	lea	si,[0004]	cx	0010																													
cs:0009	BB0000	mov	bx,0000	dx	0000																													
cs:000C	B91000	mov	cx,0010	si	0004																													
cs:000F	BA0000	mov	dx,0000	di	0000																													
cs:0012	8A04	mov	al,[si]	bp	0000																													
cs:0014	A21400	mov	[0014],al	sp	0100																													
cs:0017	46	inc	si	ds	0B12																													
cs:0018	49	dec	cx	es	0AFF																													
cs:0019	43	inc	bx	ss	0B14																													
cs:001A	E311	jcxz	002D	cs	0B0F																													
cs:001C	8A04	mov	al,[si]	ip	000F																													
cs:001E	3A061400	cmp	al,[0014]																															
cs:0022	7E05	jle	0029																															
<table border="0"> <tr> <td>es:AAA9</td> <td>8B 47 18 26 8B 57 16 C4</td> <td>лГt&лW-</td> <td>ss:0108</td> <td>5655</td> </tr> <tr> <td>es:AAB1</td> <td>5E 06 26 89 47 02 26 89</td> <td>^&&ИG&&И</td> <td>ss:0106</td> <td>5453</td> </tr> <tr> <td>es:AAB9</td> <td>17 8E 1E 0E 00 33 C0 59</td> <td>т0Δл 3 4</td> <td>ss:0104</td> <td>5251</td> </tr> <tr> <td>es:AAC1</td> <td>59 1F 5D 4D CA 06 00 B8</td> <td>Y∇JИ</td> <td>ss:0102</td> <td>5032</td> </tr> <tr> <td>es:AAC9</td> <td>3A 16 45 55 8B EC 1E 8E</td> <td>:EУлбΔ0</td> <td>ss:0100</td> <td>FE33</td> </tr> </table>										es:AAA9	8B 47 18 26 8B 57 16 C4	лГt&лW-	ss:0108	5655	es:AAB1	5E 06 26 89 47 02 26 89	^&&ИG&&И	ss:0106	5453	es:AAB9	17 8E 1E 0E 00 33 C0 59	т0Δл 3 4	ss:0104	5251	es:AAC1	59 1F 5D 4D CA 06 00 B8	Y∇JИ	ss:0102	5032	es:AAC9	3A 16 45 55 8B EC 1E 8E	:EУлбΔ0	ss:0100	FE33
es:AAA9	8B 47 18 26 8B 57 16 C4	лГt&лW-	ss:0108	5655																														
es:AAB1	5E 06 26 89 47 02 26 89	^&&ИG&&И	ss:0106	5453																														
es:AAB9	17 8E 1E 0E 00 33 C0 59	т0Δл 3 4	ss:0104	5251																														
es:AAC1	59 1F 5D 4D CA 06 00 B8	Y∇JИ	ss:0102	5032																														
es:AAC9	3A 16 45 55 8B EC 1E 8E	:EУлбΔ0	ss:0100	FE33																														

Рисунок 5 – Передача в cx значения длины массива

E File Edit View Run Breakpoints Data Options Window Help									
[1]-CPU P?ntium Pro									
cs:0000	B8120B	mov	ax,0B12	ax	0B12				
cs:0003	8ED8	mov	ds,ax	bx	0000				
cs:0005	8D360400	lea	si,[0004]	cx	0010				
cs:0009	BB0000	mov	bx,0000	dx	0000				
cs:000C	B91000	mov	cx,0010	si	0004				
cs:000F	BA0000	mov	dx,0000	di	0000				
cs:0012	8A04	mov	al,[si]	bp	0000				
cs:0014	A21400	mov	[0014],al	sp	0100				
cs:0017	46	inc	si	ds	0B12				
cs:0018	49	dec	cx	es	0AFF				
cs:0019	43	inc	bx	ss	0B14				
cs:001A	E311	jcxz	002D	cs	0B0F				
cs:001C	8A04	mov	al,[si]	ip	0012				
cs:001E	3A061400	cmp	al,[0014]						
cs:0022	7E05	jle	0029						
<div> <div>es:AAA9 8B 47 18 26 8B 57 16 C4 ЛГ1&ЛW-</div> <div>es:AAB1 5E 06 26 89 47 02 26 89 ^&&ИG&&И</div> <div>es:AAB9 17 8E 1E 0E 00 33 C0 59 10ΔП 3 Ч</div> <div>es:AAC1 59 1F 5D 4D CA 06 00 B8 Y∇JMΔ</div> <div>es:AAC9 3A 16 45 55 8B EC 1E 8E :-EУЛ&Δ0</div> </div> <div> <div>ss:0108 5655</div> <div>ss:0106 5453</div> <div>ss:0104 5251</div> <div>ss:0102 5032</div> <div>ss:0100 FE33</div> </div>									

Рисунок 6 – Передача в dx нуля (индекс максимума)

E File Edit View Run Breakpoints Data Options Window Help									
[1]-CPU P?ntium Pro									
cs:0000	B8120B	mov	ax,0B12	ax	0BCE				
cs:0003	8ED8	mov	ds,ax	bx	0000				
cs:0005	8D360400	lea	si,[0004]	cx	0010				
cs:0009	BB0000	mov	bx,0000	dx	0000				
cs:000C	B91000	mov	cx,0010	si	0005				
cs:000F	BA0000	mov	dx,0000	di	0000				
cs:0012	8A04	mov	al,[si]	bp	0000				
cs:0014	A21400	mov	[0014],al	sp	0100				
cs:0017	46	inc	si	ds	0B12				
cs:0018	49	dec	cx	es	0AFF				
cs:0019	43	inc	bx	ss	0B14				
cs:001A	E311	jcxz	002D	cs	0B0F				
cs:001C	8A04	mov	al,[si]	ip	0018				
cs:001E	3A061400	cmp	al,[0014]						
cs:0022	7E05	jle	0029						
<div> <div>es:AAA9 8B 47 18 26 8B 57 16 C4 ЛГ1&ЛW-</div> <div>es:AAB1 5E 06 26 89 47 02 26 89 ^&&ИG&&И</div> <div>es:AAB9 17 8E 1E 0E 00 33 C0 59 10ΔП 3 Ч</div> <div>es:AAC1 59 1F 5D 4D CA 06 00 B8 Y∇JMΔ</div> <div>es:AAC9 3A 16 45 55 8B EC 1E 8E :-EУЛ&Δ0</div> </div> <div> <div>ss:0108 5655</div> <div>ss:0106 5453</div> <div>ss:0104 5251</div> <div>ss:0102 5032</div> <div>ss:0100 FE33</div> </div>									

Рисунок 7 – «Движение» адреса на сл. элемент массива

E File Edit View Run Breakpoints Data Options Window Help									
[1]-CPU P?ntium Pro									
cs:0000	B8120B	mov	ax,0B12	ax	0BCE				
cs:0003	8ED8	mov	ds,ax	bx	0000				
cs:0005	8D360400	lea	si,[0004]	cx	000E				
cs:0009	BB0000	mov	bx,0000	dx	0000				
cs:000C	B91000	mov	cx,0010	si	0005				
cs:000F	BA0000	mov	dx,0000	di	0000				
cs:0012	8A04	mov	al,[si]	bp	0000				
cs:0014	A21400	mov	[0014],al	sp	0100				
cs:0017	46	inc	si	ds	0B12				
cs:0018	49	dec	cx	es	0AFF				
cs:0019	43	inc	bx	ss	0B14				
cs:001A	E311	jcxz	002D	cs	0B0F				
cs:001C	8A04	mov	al,[si]	ip	0019				
cs:001E	3A061400	cmp	al,[0014]						
cs:0022	7E05	jle	0029						
<div> <div>es:AAA9 8B 47 18 26 8B 57 16 C4 ЛГ1&ЛW-</div> <div>es:AAB1 5E 06 26 89 47 02 26 89 ^&&ИG&&И</div> <div>es:AAB9 17 8E 1E 0E 00 33 C0 59 10ΔП 3 Ч</div> <div>es:AAC1 59 1F 5D 4D CA 06 00 B8 Y∇JMΔ</div> <div>es:AAC9 3A 16 45 55 8B EC 1E 8E :-EУЛ&Δ0</div> </div> <div> <div>ss:0108 5655</div> <div>ss:0106 5453</div> <div>ss:0104 5251</div> <div>ss:0102 5032</div> <div>ss:0100 FE33</div> </div>									

Рисунок 8 – Уменьшение общего количества оставшихся эл-тов

```

E File Edit View Run Breakpoints Data Options Window Help
[1]-CPU Pntium Pro
cs:0000 B8120B mov ax,0B12
cs:0003 8ED8 mov ds,ax
cs:0005 8D360400 lea si,[0004]
cs:0009 BB0000 mov bx,0000
cs:000C B91000 mov cx,0010
cs:000F BA0000 mov dx,0000
cs:0012 8A04 mov al,[si]
cs:0014 A21400 mov [0014],al
cs:0017 46 inc si
cs:0018 49 dec cx
cs:0019 43 inc bx
cs:001A E311 jcxz 002D
cs:001C 8A04 mov al,[si]
cs:001E 3A061400 cmp al,[0014]
cs:0022 7E05 jle 0029

es:AAA9 8B 47 18 26 8B 57 16 C4 ЛГТ&ЛW_
es:AAB1 5E 06 26 89 47 02 26 89 ^&&ИГ&&И
es:AAB9 17 8E 1E 0E 00 33 C0 59 10АВ 3 Ч
es:AAC1 59 1F 5D 4D CA 06 00 B8 V&ИМ&&
es:AAC9 3A 16 45 55 8B EC 1E 8E :-ЕУ&&А0
ss:0108 5655
ss:0106 5453
ss:0104 5251
ss:0102 5032
ss:0100 FE33

```

Рисунок 9 – Увеличение текущего индекса

```

E File Edit View Run Breakpoints Data Options Window Help
[1]-CPU Pntium Pro
cs:0022 7E05 jle 0029
cs:0024 A21400 mov [0014],al
cs:0027 8BD3 mov dx,bx
cs:0029 43 inc bx
cs:002A 46 inc si
cs:002B E2EF loop 001C
cs:002D B44C mov ah,4C
cs:002F B000 mov al,00
cs:0031 CD21 int 21
cs:0033 00CE add dh,cl
cs:0035 FF db FF
cs:0036 7C7F jl 00B7
cs:0038 27 daa
cs:0039 1BEC sbb bp,sp
cs:003B A6 cmpsb

es:AAA9 8B 47 18 26 8B 57 16 C4 ЛГТ&ЛW_
es:AAB1 5E 06 26 89 47 02 26 89 ^&&ИГ&&И
es:AAB9 17 8E 1E 0E 00 33 C0 59 10АВ 3 Ч
es:AAC1 59 1F 5D 4D CA 06 00 B8 V&ИМ&&
es:AAC9 3A 16 45 55 8B EC 1E 8E :-ЕУ&&А0
ss:0108 5655
ss:0106 5453
ss:0104 5251
ss:0102 5032
ss:0100 FE33

```

Рисунок 10 – Прохождение по циклу

```

E File Edit View Run Breakpoints Data Options Window Help
[1]-CPU Pntium Pro
0B0F:0022 7E05 jle 0029
0B0F:0024 A21400 mov [0014],al
0B0F:0027 8BD3 mov dx,bx
0B0F:0029 43 inc bx
0B0F:002A 46 inc si
0B0F:002B E2EF loop 001C
0B0F:002D B44C mov ah,4C
0B0F:002F B000 mov al,00
0B0F:0031 CD21 int 21
0B0F:0033 00CE add dh,cl
0B0F:0035 FF db FF
0B0F:0036 7C7F jl 00B7
0B0F:0038 27 daa
0B0F:0039 1BEC sbb bp,sp
0B0F:003B A6 cmpsb

0AFF:AAA9 8B 47 18 26 8B 57 16 C4 ЛГТ&ЛW_
0AFF:AAB1 5E 06 26 89 47 02 26 89 ^&&ИГ&&И
0AFF:AAB9 17 8E 1E 0E 00 33 C0 59 10АВ 3 Ч
0AFF:AAC1 59 1F 5D 4D CA 06 00 B8 V&ИМ&&
0AFF:AAC9 3A 16 45 55 8B EC 1E 8E :-ЕУ&&А0
ss:0108 5655
ss:0106 5453
ss:0104 5251
ss:0102 5032
ss:0100 FE33

```

Рисунок 11 – Завершение работы программы

Вывод

В данной лабораторной работе были изучены концепции и освоены технологии программирования 16-битных приложений на языке ассемблера для архитектуры процессоров семейства Intel. Также закреплены навыки работы с отладчиком Turbo Debugger.