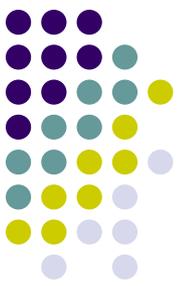


Лабораторная работа №2



Построение системы кодирования информации с использованием языка Ассемблер

Постановка задачи



В 8 ячейках памяти, начиная с адреса 800h, произвести кодирование данных по правилу

$D7 D6 D5 D4 D3 D2 D1 D0 \Rightarrow$

$\Rightarrow D_4 D_3 \overline{D_5} \overline{D_7} D3 D2 D6 \overline{D0} >$

и по правилу

$D7 D6 D5 D4 D3 D2 D1 D0 \Rightarrow$

$\Rightarrow \overline{D_5} \overline{D_4} \overline{D_3} \overline{D_2} \overline{D_1} \overline{D_0} \overline{D_7} \overline{D_6}$

Результат сохранить в следующие 8 ячеек памяти

Алгоритм решения



Указание адресов извлечения данных (HL),
адреса возврата (DE) и размера массива (C).

Извлечение данных

Проверка условия

Кодирование и сохранение результата

Переход к следующей ячейке

Операции кодирования



Циклический сдвиг

только в аккумуляторе /

–влево RLC

D7 D6 D5 D4 D3 D2 D1 D0 =>

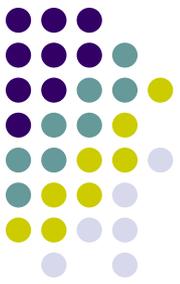
=> D6 D5 D4 D3 D2 D1 D0 D7

–вправо RRC

D7 D6 D5 D4 D3 D2 D1 D0 =>

=> D0 D7 D6 D5 D4 D3 D2 D1

Операции кодирования



Инверсия

операция XOR / $x \oplus 0 = x$, $x \oplus 1 = \bar{x}$ /

Пример : XOR 60h; $A \leftarrow A \oplus 60h$

	D7	D6	D5	D4	D3	D2	D1	D0
\oplus	0	1	1	0	0	0	0	0
<hr/>								
	D7	$\bar{D6}$	$\bar{D5}$	D4	D3	D2	D1	D0

Операции кодирования



Выделение бит

операция ANI / $x \& 0 = 0$, $x \& 1 = x$ /

Пример : ANI 0Ch ; $A \leftarrow A \& 0Ch$

	D7	D6	D5	D4	D3	D2	D1	D0
&	0	0	0	0	1	1	0	0
<hr/>								
	0	0	0	0	D3	D2	0	0

Операции кодирования



Установка бит в логический ноль
операция ANI / $x \& 0 = 0$, $x \& 1 = x$ /
Пример : ANI 0EFh

	D7	D6	D5	D4	D3	D2	D1	D0
&	1	1	1	0	1	1	1	1
<hr/>								
	D7	D6	D5	0	D3	D2	D1	D0

Операции кодирования



5) установка бит в логическую единицу

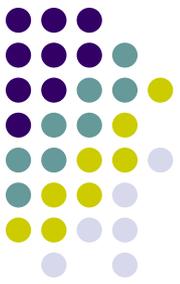
операция ORI / $x \vee 0 = x$, $x \vee 1 = 1$ /

Пример: ORI 80h

	D7	D6	D5	D4	D3	D2	D1	D0
\vee	1	0	0	0	0	0	0	0
<hr/>								
	1	D6	D5	D4	D3	D2	D1	D0

Операция проверки условия

D3>D2



XXXX **00**XX → 0000 **00**00b –
– 0000 1000b ≠ 0

XXXX **01**XX → 0000 **01**00b –
– 0000 1000b ≠ 0

XXXX **10**XX → 0000 **10**00b –
– 0000 1000b = 0

XXXX **11**XX → 0000 **11**00b –
– 0000 1000b ≠ 0

Данные в аккумуляторе

ANI 0000 1100b;

CPI 0000 1000b;

JNZ L2;

{кодирование по
первому алгоритму}

JMP L3;

L2: {кодирование по
второму алгоритму}

L3: {продолжение
программы}

Задание начальных адресов

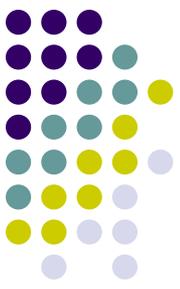


LXI H,am1 ; HL \leftarrow am1 = 0800h – начальный адрес
исходных данных

LXI D,am2 ; DE \leftarrow am2 = 0808h – начальный адрес
кодированных данных

MVI C,am2-am1 ; C \leftarrow 8 – число кодируемых
байт данных

Проверка условия



L1: MOV A,M ; A ← M(rp H) ;извлечение данных
 из памяти в аккумулятор

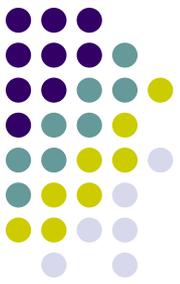
ANI 0Ch ; выделение битов D3 и D2

 A ← A & 0000 1100b = 0 0 0 0 D3 D2 0 0

CPI 08h ; A – 0000 1000b(проверка
 условия)

Если после вычитания $Z=1$ (получен ноль), то условие $D3 > D2$ выполнено и кодирование происходит по первому алгоритму, иначе ($Z=0$, не равно нулю), то кодирование происходит по второму алгоритму

Кодирование по алгоритму 1



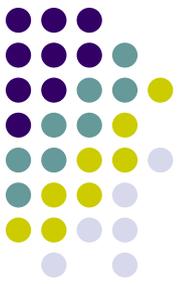
1) Инверсия бита D5 и D0

MOV A,M ; A ← M(rp H) извлечение из
памяти

XRI 21h ; A ← A xor 0010 0001b

STAX D ; M(rp D) ← A сохранение
результата

Кодирование по алгоритму 1



2) Перестановка бит D4 и D7

ANI 90h ; $A \leftarrow A \& 1001\ 0000b = D7\ 0\ 0\ D4\ 0\ 0\ 0\ 0$

JPE L3 ; проверка $D7+D4$ – четное число?

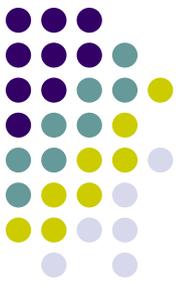
LDAX D ; $A \leftarrow M(rp\ D)$

XRI 90h ; $A \leftarrow A \text{ xor } 1001\ 0000b$ инверсия бит
D7 и D4

STAX D ; сохранение результата

L3:

Кодирование по алгоритму 1



3) Перестановка бит D6 и D1

L3: LDAX D

ANI 42h ; $A \leftarrow A \& 0100\ 0010b = 0\ D6\ 0\ 0\ 0$
0 D1 0

JPE L4 ; проверка D6+D1 – четное
число?

LDAX D ; $A \leftarrow M(rp\ D)$

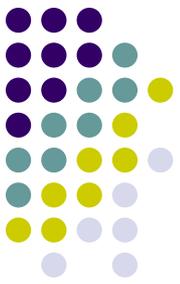
XRI 42h ; $A \leftarrow A\ xor\ 0100\ 0010b$ инверсия
бит D6 и D1

STAX D ; сохранение результата

JMP L4 ; Конец кодирования

L4: ...

Кодирование по алгоритму 2



```
L2:  MOV  A,M  ; A ← M(rp H)
      XRI  50h  ; A ← A xor 0101 0000b, инверсия
           бит D6 и D4
      RLC           ; сдвиг влево
      RLC           ; сдвиг влево
      STAX D       ; M(rp D) ← A
```

Переход к следующей ячейке памяти



L4: INX H ; $HL \leftarrow HL + 1$

INX D ; $DE \leftarrow DE + 1$

DCR C ; $C \leftarrow C - 1$

JNZ L1 ; проверка пройдены ли все
ячейки

Подбор чисел для проверки кодирования



Кодирование по первому правилу / $D3 > D2$ /

	D7	D6	D5	D4	D3	D2	D1	D0	=>
=>	D4	D1	$\overline{D5}$	D7	D3	D2	D6	$\overline{D0}$	

	D7	D6	D5	D4	D3	D2	D1	D0	
1					1	0			
2					1	0			
3					1	0			
4					1	0			

Подбор чисел для проверки кодирования



Кодирование по первому правилу / $D3 > D2$ /

	D7	D6	D5	D4	D3	D2	D1	D0	=>
=>	D4	D1	$\overline{D5}$	D7	D3	D2	D6	$\overline{D0}$	

	D7	D6	D5	D4	D3	D2	D1	D0	
1			0		1	0		1	
2			1		1	0		0	
3			0		1	0		1	
4			1		1	0		1	

Подбор чисел для проверки кодирования



Кодирование по первому правилу / $D3 > D2$ /

	D7	D6	D5	D4	D3	D2	D1	D0	=>
=>	D4	D1	$\overline{D5}$	D7	D3	D2	D6	$\overline{D0}$	

	D7	D6	D5	D4	D3	D2	D1	D0	
1	0	1	0	1	1	0	0	1	
2	1	1	1	0	1	0	1	0	
3	0	1	0	0	1	0	0	1	
4	0	1	1	0	1	0	1	1	

Подбор чисел для проверки кодирования



Кодирование по первому правилу / $D3 > D2$ /

	D7	D6	D5	D4	D3	D2	D1	D0	=>
=>	D4	D1	$\overline{D5}$	D7	D3	D2	D6	$\overline{D0}$	

	D7	D6	D5	D4	D3	D2	D1	D0	
1	0	1	0	1	1	0	0	1	59h
2	1	1	1	0	1	0	1	0	EAh
3	0	1	0	0	1	0	0	1	49h
4	0	1	1	0	1	0	1	1	6Bh

Соответствие чисел до и после кодирования



Кодирование по первому правилу / $D3 > D2$ /

	D7	D6	D5	D4	D3	D2	D1	D0	=>
=>	D4	D1	$\overline{D5}$	D7	D3	D2	D6	$\overline{D0}$	

До кодирования	После кодирования		
59h	0101 1001b	1010 1010b	AAh
EAh	1110 1010b	0101 1011b	5Bh
49h	0100 1001b	0010 1010b	2Ah
6Bh	0110 1011b	0100 1010b	4Ah

Подбор чисел для проверки кодирования

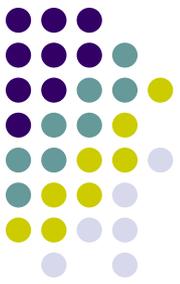


Кодирование по второму правилу / $D3 \leq D2$ /

	D7	D6	D5	D4	D3	D2	D1	D0	=>
=>	D5	$\overline{D4}$	D3	D2	D1	D0	D7	$\overline{D6}$	

	D7	D6	D5	D4	D3	D2	D1	D0	
1		1		1	1	1			
2		1		0	0	0			
3		0		1	0	1			
4		1		1	0	0			

Подбор чисел для проверки кодирования



Кодирование по второму правилу / $D3 \leq D2$ /

	D7	D6	D5	D4	D3	D2	D1	D0	=>
=>	D5	$\overline{D4}$	D3	D2	D1	D0	D7	$\overline{D6}$	

	D7	D6	D5	D4	D3	D2	D1	D0	
1	X	1	X	1	1	1	X	X	DEh
2	X	1	X	0	0	0	X	X	C3h
3	X	0	X	1	0	1	X	X	96h
4	X	1	X	1	0	0	X	X	F0h

Соответствие чисел до и после кодирования



Кодирование по второму правилу / $D3 \leq D2$ /

	D7	D6	D5	D4	D3	D2	D1	D0	=>
=>	D5	$\overline{D4}$	D3	D2	D1	D0	D7	$\overline{D6}$	

	До кодирования	После кодирования	
DEh	1101 1110b	0011 1010b	3Ah
C3h	1100 0011b	0100 1110b	4Eh
96h	1001 0110b	0001 1011b	1Bh
F0h	1111 0000b	1000 0010b	82h