

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

Кафедра информатики и прикладной математики

Организация ЭВМ и систем

Лабораторная работа 2



Старались: **Шкаруба Н.Е.**
Суханова В.А.
Проверил: **Тропченко А.А.**
Группа: **Р3318**

2016 г

Задание:

Совместить перевод 10->2 и 2->10 в одной программе для целых и дробных чисел и разработать программы на C51 и в Ассемблере а51 для ввода и вывода двузначных чисел. Сравнить листинги .lst программ в C51 и A51 и пояснить различия в программах.

Выполнение для целых чисел:

1. Исходный текст программы на C51

```
#include <reg51.h>

main() {
    P2 = (P1 >> 4) * 10 + (P1 & 0x0f);
    P3 = ((P2 / 10) << 4) + (P2 % 10);
}
```

2. Результат выполнения C51

Name	Value	Type
P1	0x0B	uchar
P2	0x0B	uchar
P3	0x11	uchar

3. Файл листинга (файл с расширением *.lst)

C51 COMPILER V9.56.0.0 L2 10/23/2016 10:15:26 PAGE 1

C51 COMPILER V9.56.0.0, COMPILATION OF MODULE L2
OBJECT MODULE PLACED IN .\Objects\L2.obj
COMPILER INVOKED BY: C:\Keil_v5\C51\BIN\C51.EXE L2.c OPTIMIZE(8,SPEED) BROWSE
DEBUG OBJECTEXTEND CODE PRINT(.Listings\L
-2.lst) TABS(2) OBJECT(.Objects\L2.obj)

line level source

```
1      #include <reg51.h>
2
3      main() {
4 1      P2 = (P1 >> 4) * 10 + (P1 & 0x0f);
5 1      P3 = ((P2 / 10) << 4) + (P2 % 10);
6 1      }
```

*** WARNING C290 IN LINE 6 OF L2.c: missing return value

C51 COMPILER V9.56.0.0 L2 10/23/2016 10:15:26 PAGE 2

ASSEMBLY LISTING OF GENERATED OBJECT CODE

; FUNCTION main (BEGIN)

```

; SOURCE LINE # 3
; SOURCE LINE # 4
0000 E590      MOV   A,P1
0002 C4        SWAP  A
0003 540F      ANL   A,#0FH
0005 75F00A    MOV   B,#0AH
0008 A4        MUL   AB
0009 FF        MOV   R7,A
000A E590      MOV   A,P1
000C 540F      ANL   A,#0FH
000E 2F        ADD   A,R7
000F F5A0      MOV   P2,A
; SOURCE LINE # 5
0011 E5A0      MOV   A,P2
0013 75F00A    MOV   B,#0AH
0016 84        DIV   AB
0017 C4        SWAP  A
0018 54F0      ANL   A,#0F0H
001A FF        MOV   R7,A
001B E5A0      MOV   A,P2
001D 75F00A    MOV   B,#0AH
0020 84        DIV   AB
0021 E5F0      MOV   A,B
0023 2F        ADD   A,R7
0024 F5B0      MOV   P3,A
; SOURCE LINE # 6
0026 22        RET
; FUNCTION main (END)

```

MODULE INFORMATION: STATIC OVERLAYABLE

```

CODE SIZE      =   39   ----
CONSTANT SIZE   =   ----   ----
XDATA SIZE      =   ----   ----
PDATA SIZE      =   ----   ----
DATA SIZE       =   ----   ----
IDATA SIZE      =   ----   ----
BIT SIZE        =   ----   ----

```

END OF MODULE INFORMATION.

C51 COMPILATION COMPLETE. 1 WARNING(S), 0 ERROR(S)

4. Исходный текст программы на A51

```
cseg at 0          ; Place code segment at 0 address

    mov a, P1      ; Read number to convert
    anl a, #0f0h
    swap a

    mov b, #10      ; Move (A1*(10)2 + A0)2 to P2
    mul ab
    mov b, a
    mov a, P1
    anl a, #0fh
    add a, b
    mov P2, a

    mov a, P2      ; Move ((P2/10)<<4) + (P2%10) to P3
    mov b, #10
    div ab
    swap a
    add a, b        ;
    mov P3, a

end
```

5. Результат выполнения A51

Name	Value	Type
P1	0x0B	uchar
P2	0x0B	uchar
P3	0x11	uchar

Выполнение для дробных чисел:

1. Исходный текст программы на C51

```
#include <reg51.h>

main() {
    unsigned int y;

    y = (((P1 & 0xf0) >> 4) * 10 + (P1 & 0xf) << 8);
    y = y / 100;
    P2 = y;

    y = y * 10;
    P3 = ((y & 0xf00) >> 4);
    P3 += (((y & 0xff) * 10) & 0xf00) >> 8;
}
```

2. Результат выполнения C51

Name	Value	Type
P2	0x1C	uchar
P3	0x10	uchar
<Ent...		

3. Файл листинга (файл с расширением *.lst)

C51 COMPILER V9.56.0.0 LAB2FLOATC
PAGE 1

10/23/2016 11:49:14

C51 COMPILER V9.56.0.0, COMPILATION OF MODULE LAB2FLOATC
OBJECT MODULE PLACED IN .\Objects\lab2floatc.obj
COMPILER INVOKED BY: C:\Keil_v5\C51\BIN\C51.EXE lab2floatc.c OPTIMIZE(8,SPEED)
BROWSE DEBUG OBJECTEXTEND CODE PRINT(.\\Li
-stings\lab2floatc.lst) TABS(2) OBJECT(.\\Objects\lab2floatc.obj)

line level source

```
1      #include <reg51.h>
2
3      main() {
4 1      unsigned int y;
5 1
6 1      y = (((P1 & 0xf0) >> 4) * 10 + (P1 & 0xf) << 8);
7 1      y = y / 100;
8 1      P2 = y;
9 1
10 1     y = y * 10;
11 1     P3 = ((y & 0xf00) >> 4);
```

```
12 1    P3 += (((y & 0xff) * 10) & 0xf00) >> 8;
13 1    }
```

*** WARNING C290 IN LINE 13 OF lab2floatc.c: missing return value

C51 COMPILER V9.56.0.0 LAB2FLOATC

10/23/2016 11:49:14

PAGE 2

ASSEMBLY LISTING OF GENERATED OBJECT CODE

```
        ; FUNCTION main (BEGIN)
                                ; SOURCE LINE # 3
                                ; SOURCE LINE # 6
0000 E590      MOV    A,P1
0002 54F0      ANL    A,#0F0H
0004 C4        SWAP   A
0005 540F      ANL    A,#0FH
0007 75F00A    MOV    B,#0AH
000A A4        MUL    AB
000B FF        MOV    R7,A
000C E590      MOV    A,P1
000E 540F      ANL    A,#0FH
0010 7C00      MOV    R4,#00H
0012 2F        ADD    A,R7
0013 FF        MOV    R7,A
0014 EF        MOV    A,R7
;---- Variable 'y' assigned to Register 'R2/R3' ----
                                ; SOURCE LINE # 7
0015 7D64      MOV    R5,#064H
0017 7F00      MOV    R7,#00H
0019 FE        MOV    R6,A
001A 120000    E      LCALL ?C?UIDIV
001D AA06      MOV    R2,AR6
001F AB07      MOV    R3,AR7
                                ; SOURCE LINE # 8
0021 8BA0      MOV    P2,R3
                                ; SOURCE LINE # 10
0023 7C00      MOV    R4,#00H
0025 7D0A      MOV    R5,#0AH
0027 120000    E      LCALL ?C?IMUL
002A AA06      MOV    R2,AR6
002C AB07      MOV    R3,AR7
                                ; SOURCE LINE # 11
002E EA        MOV    A,R2
002F 540F      ANL    A,#0FH
0031 C4        SWAP   A
0032 F8        MOV    R0,A
0033 54F0      ANL    A,#0F0H
0035 C8        XCH    A,R0
0036 E4        CLR    A
0037 C4        SWAP   A
```

```

0038 540F      ANL  A,#0FH
003A 48        ORL  A,R0
003B F5B0      MOV  P3,A
                  ; SOURCE LINE # 12
003D 7E00      MOV  R6,#00H
003F AF03      MOV  R7,AR3
0041 120000    E   LCALL ?C?IMUL
0044 EE        MOV  A,R6
0045 540F      ANL  A,#0FH
0047 25B0      ADD  A,P3
0049 F5B0      MOV  P3,A
                  ; SOURCE LINE # 13
004B 22        RET
                  ; FUNCTION main (END)

```

MODULE INFORMATION: STATIC OVERLAYABLE
C51 COMPILER V9.56.0.0 LAB2FLOATC
PAGE 3

10/23/2016 11:49:14

```

CODE SIZE      =  76  ---
CONSTANT SIZE  =  ---  ---
XDATA SIZE     =  ---  ---
PDATA SIZE     =  ---  ---
DATA SIZE      =  ---  ---
IDATA SIZE     =  ---  ---
BIT SIZE       =  ---  ---
END OF MODULE INFORMATION.

```

C51 COMPILATION COMPLETE. 1 WARNING(S), 0 ERROR(S)

4. Исходный текст программы на A51

```

; main
MOV  A,P1
ANL  A,#0F0H
SWAP A
ANL  A,#0FH
MOV  B,#0AH
pMUL AB
MOV  R7,A
MOV  A,P1
ANL  A,#0FH
MOV  R4,#00H
ADD  A,R7

```

```

MOV R7,A
MOV A,R7
MOV R5,#064H
MOV R7,#00H
MOV R6,A
LCALL ?C?UIDIV
MOV R2,AR6
MOV R3,AR7
MOV P2,R3
MOV R4,#00H
MOV R5,#0AH
LCALL ?C?IMUL
MOV R2,AR6
MOV R3,AR7
MOV A,R2
ANL A,#0FH
SWAP A
MOV R0,A
ANL A,#0F0H
XCH A,R0
CLR A
SWAP A
ANL A,#0FH
ORL A,R0
MOV P3,A
MOV R6,#00H
MOV R7,AR3
LCALL ?C?IMUL
MOV A,R6
ANL A,#0FH
ADD A,P3
MOV P3,A
RET

```

Результат выполнения A51

Name	Value	Type
P2	0x1C	uchar
P3	0x10	uchar
<Ent...		

Вывод

Мы совместили перевод 10->2 и 2->10 в одной программе для целых и дробных чисел и разработали программы на C51 и в Ассемблере a51 для ввода и вывода двузначных чисел. Сравнили листинги .lst программ в C51 и A51.