Университет ИТМО

**Лабораторная работа №2**

Выполнил:

студент III курса группы 3125

Припадчев Артём

Проверит:

Тропченко А.А.

Санкт-Петербург

2014

Совместить перевод 10→2 и 2→10 в одной программе на С51 и в Ассемблере а51 для ввода и вывода двузначных десятичных чисел.

**Перевод целых чисел**

**Программа на C:**

#include <reg51.h>

void main()

{

//Преобразование из двоично-десятичной системы в двоичную

P2 = (P1>>4)\*10 + (P1&0x0f); //ввод 2/10 числа с порта Р1, выделение со сдвигом старшей тетрады, вывод двоичного числа в Р2

//Преобразование из двоичной системы в десятично-двоичную

P3 = ((P2/10)<<4) + (P2%10);

}

**Листинг программы**

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

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

Занимает 22 байта

**Программа на ассемблере:**

cseg at 0 ; абсолютный сегмент кода размещается с адреса 0

; в программной памяти

mov a,P1 ; ввод байта данных с порта Р1 в аккумулятор

anl a,#0f0h ; выделение старшей тетрады А1

swap a ; тетрада перемещается в младшие разряды

mov b,#10 ; 10 в регистре множителя

mul ab ; a=A1\*10, b=0

mov b,a ; сохранить младшие разряды произведения

mov a,P1 ; ввод байта данных с порта Р1 в аккумулятор

anl a,#0fh ; выделение младшей тетрады А0

add a,b ; a=(A1\*(10)2 + A0)2

mov P2,a ; вывод результата в порт Р2

mov a, P2 ; ввод байта данных с порта Р2 в аккумулятор

mov b, #10 ; 10 в регистре делителя

div ab ; a=A1/10

swap a ; тетрада перемещается в младшие разряды

add a, b ; добавление b к a

mov P3, a ; вывод результата в порт Р3

end ; псевдокоманда ассемблера- завершение текста

**Перевод дробных чисел**

**Программа на С:**

#include <reg51.h>

unsigned int y;

main()

{ while(1)

{y=(((P1&0xf0)>>4)\*10 + (P1&0xf)<<8); //дробное в масштабе M=**100\*28**

y=y/100; ; //перевод в двоичное дробное в масштабе **28**

P2=y;

y=y\*10;

P3=((y&0xf00)>>4);

P3+=(((y&0xff)\*10)&0xf00)>>8;

}

}

**Листинг программы**

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

0015 750000 R MOV y+01H,#00H

0018 F500 R MOV y,A

001A FE MOV R6,A

001B AF00 R MOV R7,y+01H

001D 7D64 MOV R5,#064H

001F 120000 E LCALL ?C?UIDIV

0022 8E00 R MOV y,R6

0024 8F00 R MOV y+01H,R7

0026 8500A0 R MOV P2,y+01H

0029 AE00 R MOV R6,y

002B AF00 R MOV R7,y+01H

002D 7C00 MOV R4,#00H

002F 7D0A MOV R5,#0AH

0031 120000 E LCALL ?C?IMUL

0034 8E00 R MOV y,R6

0036 8F00 R MOV y+01H,R7

0038 E500 R MOV A,y

003A 540F ANL A,#0FH

003C C4 SWAP A

003D F8 MOV R0,A

003E 54F0 ANL A,#0F0H

0040 C8 XCH A,R0

0041 E4 CLR A

0042 C4 SWAP A

0043 540F ANL A,#0FH

0045 48 ORL A,R0

0046 F5B0 MOV P3,A

0048 7E00 MOV R6,#00H

004A AF00 R MOV R7,y+01H

004C 120000 E LCALL ?C?IMUL

004F EE MOV A,R6

0050 540F ANL A,#0FH

0052 25B0 ADD A,P3

0054 F5B0 MOV P3,A