Университет ИТМО  
Кафедра вычислительной техники

Организация ЭВМ и систем

**Лабораторная работа №1**

*Вариант 9*

Выполнил:

Студент группы P3317

Сорокин Ю.Б.

Преподаватель

Скорубский В.И.

Санкт-Петербург

2016 г.

**Задание**

9. Десятичное сложение (вычитание) в неупакованных форматах, положение запятой фиксировано

“56,54” +” 75, 56” =” 32,10”

**Выполнение**

*Описание алгоритма:*

Выделяется память объемом на 1 байт больше исходной строки.

В цикле проходим по двум данным строкам и суммируем элементы. Встречается символ «,» пишем его

*Листинг C51 кода с прямым доступом к памяти*

Program Size: data=10.0 xdata=6 code=243

#include <reg51.h>

char code str\_a?[]="25,82";

char code str\_b[]="77,39";

char xdata str\_c[6];

main()

{

char i = 0, char\_a=0, char\_b=0, buff = 0;

for( i = 4; i>=0; i-- )

{

if(str\_a[i]==',')

str\_c[i+1]=str\_a[i];

else{

char\_a = str\_a[i]-0x30;

char\_b = str\_b[i]-0x30;

if(char\_a+char\_b+buff>=0x0A){

str\_c[i+1] = char\_a+char\_b+buff+0x26;

buff=0x1;

}else{

str\_c[i+1] = char\_a+char\_b+buff+0x30;

}

}

if(i==0 & buff==0x01)

str\_c[i]=0x31;

}

while(1);

}

*Листинг C51 кода с косвенным доступом к памяти*

Program Size: data=18.0 xdata=0 code=393

#include <reg51.h>

char code \* str\_a="25,82";

char code \* str\_b="77,39";

char xdata \* pointer\_c;

main()

{

char i = 0, char\_a=0, char\_b=0, buff = 0;

for( i = 4; i>=0; i--)

{

if(\*(str\_a+i)==',')

\*(pointer\_c+i+1)=\*(str\_a+i);

else{

char\_a = \*(str\_a+i)-0x30;

char\_b = \*(str\_b+i)-0x30;

if(char\_a+char\_b+buff>=0x0A){

\*(pointer\_c+i+1) = char\_a+char\_b+buff+0x26;

buff=0x1;

}else{

\*(pointer\_c+i+1) = char\_a+char\_b+buff+0x30;

}

}

if(i==0 & buff==0x01)

\*(pointer\_c+i)=0x31;

}

while(1);

}

*Листинг A51 кода*

$TITLE(LAB1\_1)

XSEG AT 70

RES\_STR: ds 50

CSEG AT 0

JMP START

STR\_A: DB "25,82", 00h

STR\_B: DB "77,33", 00h

START:

CLR A

MOV R1, #0 //LEN OF STR\_A

MOV R0, #0 //ITERATOR

COUNT\_LENGTH\_STR:

MOV DPTR,#STR\_A

MOV A, r0

MOVC A,@A+DPTR

JZ STR\_IS\_COUNTED

INC R0

AJMP COUNT\_LENGTH\_STR

STR\_IS\_COUNTED:

MOV A, R0

MOV R2, A //cycle length

MOV R1, #RES\_STR

MOV A, R1

ADD A, R0

MOV R1, A

DEC R0

MOV R3,#0 //additional 1

MOV R4,#0 //buffer for str\_a char

START\_SUM:

MOV DPTR,#STR\_A

MOV A, r0

MOVC A,@A+DPTR

MOV R4, A

MOV B, #2CH

SUBB A, B

JNZ NOT\_A\_POINT

ADD A,B

MOVX @R1,A // write point

AJMP CYCLE\_SETTINGS

NOT\_A\_POINT:

MOV DPTR,#STR\_B

MOV A, r0

MOVC A,@A+DPTR

ADD A, R4

ADD A, R3

MOV R4, A //SUM IS SAVED

MOV B, #69H

SUBB A, B

JC LESS\_THEN\_10

MOV R3, #1

MOV B, #1H

SUBB A,B

ADD A,#30H

MOVX @R1,A //write more then 10

AJMP CYCLE\_SETTINGS

LESS\_THEN\_10:

MOV R3, #0

MOV A, R4

SUBB A,#30H

MOVX @R1,A //write less then 10

AJMP CYCLE\_SETTINGS

CYCLE\_SETTINGS:

DEC R0

DEC R2

DEC R1

MOV A, R2

JNZ START\_SUM

MOV A, R3

JZ END\_PROG

MOVX @R1,A //write 1

END\_PROG:

MOV r0, #RES\_STR

INC r0

INC r0

NOP

JMP $

END