



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 Информатика и вычислительная техника

О Т Ч Е Т

по лабораторной работе № 3

Название: Арифметическая обработка данных.

Дисциплина: Микропроцессорные системы.

Студент

ИУ6-62Б

(Группа)

(Подпись, дата)

С.В. Астахов, Д.И. Вариханов

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

(И.О. Фамилия)

Москва, 2022

Вариант 1.

Цели работы:

- изучение способов представления числовых данных в микроконтроллерах;
- изучение двоичных арифметических операций;
- программирование арифметических процедур.

Ход работы.

Задание 1.

Изучить программу для исследования арифметических операций в стартовом наборе STK500, приведенную ниже.

Алгоритм программы приведен на рисунках 1 и 2.

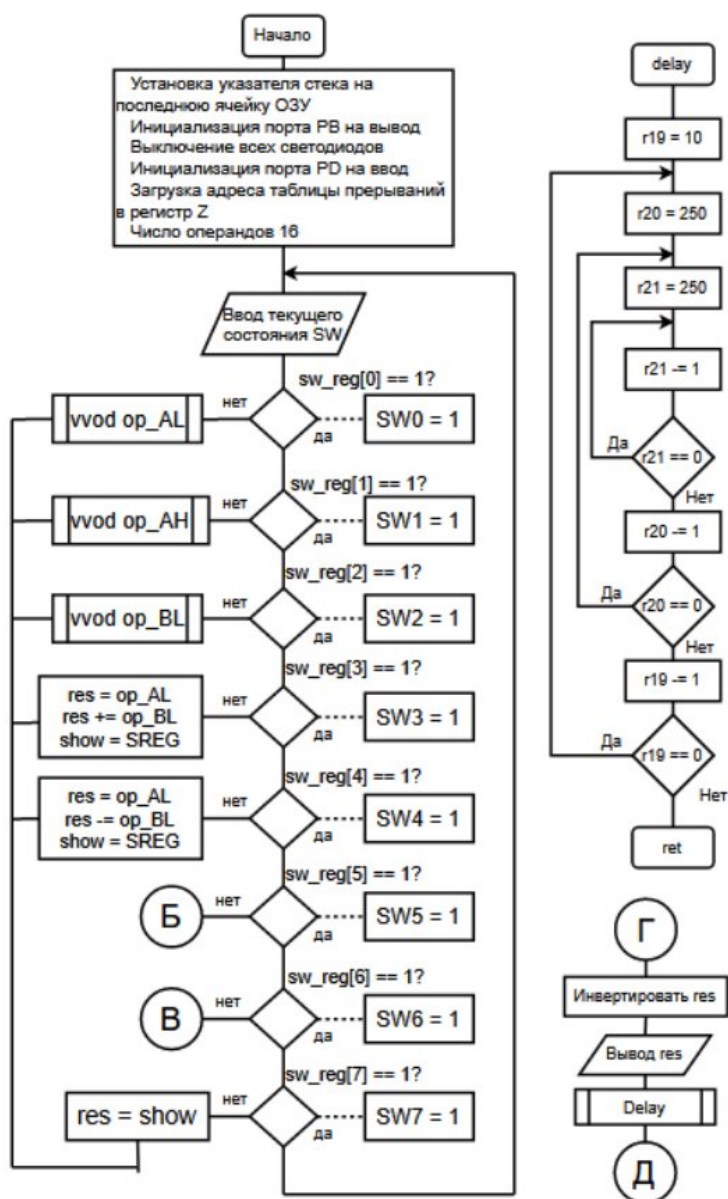


Рисунок 1 – схема алгоритма

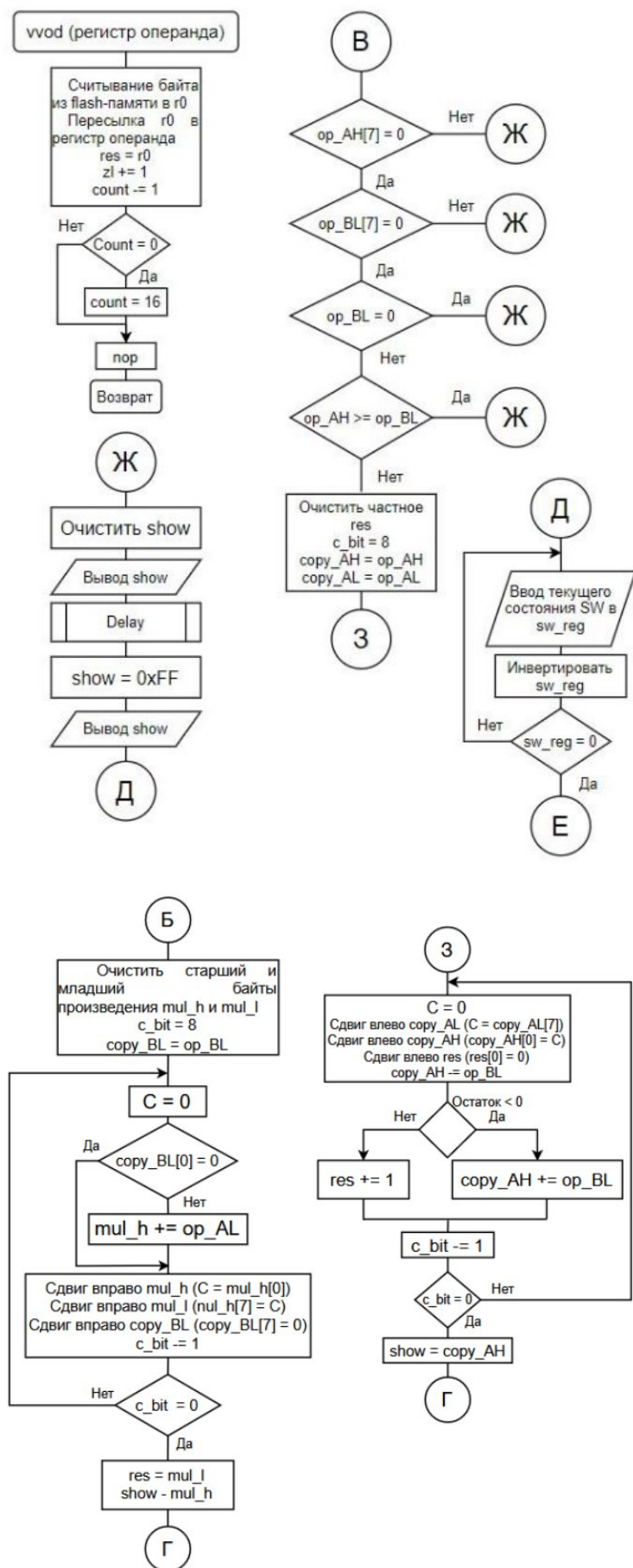


Рисунок 2 – схема алгоритма

Исходный код:

```
;*****
;*****
;Программа тестирования в STK500 двоичных арифметических
операций
; сложения, вычитания, умножения, деления
;Порт PD – порт управления для выбора операндов и операций
;Порт PB – порт индикации исходных операндов и результатов
операции
;Соединения шлейфами: порт PB-LED, порт PD-SW
;*****
;*****
.include "m8515def.inc" ;файл определений для ATmega8515
;назначение входов порта PD
.equ SW_op_AL = 0 ;кнопка выбора операнда op_AL
.equ SW_op_AH = 1 ;кнопка выбора операнда op_AH
.equ SW_op_BL = 2 ;кнопка выбора операнда op_BL
.equ SW_ADD = 3 ;кнопка операции сложения res=op_AL+op_BL
.equ SW_SUB = 4 ;кнопка операции вычитания res=op_AL-op_BL
.equ SW_MUL = 5 ;кнопка операции умножения shov.res=op_AL x
op_BL
.equ SW_DIV = 6 ;кнопка операции деления
res=op_AH.op_AL/op_BL
.equ SW_SHOW = 7 ;кнопка для просмотра признаков сложения-
вычитания,
;старшего байта произведения или остатка при делении
.def op_AL = r16 ;1-й операнд AL
.def op_AH = r17 ;старший байт делимого AH
.def op_BL = r18 ;2-й операнд BL
.def res = r1 ;результат операции (сумма, разность,
; младший байт произведения или частное)
.def show = r31 ;регистр признаков сложения-вычитания,
; старшего байта произведения или остатка при делении
.def mul_l = r21 ;младший байт произведения
.def mul_h = r22 ;старший байт произведения

.def copy_AH = r23 ;копия старшего байта делимого
.def copy_AL = r24 ;копия младшего байта делимого
.def copy_BL = r25 ;копия множителя
.def temp = r26 ;временный регистр
.def sw_reg = r27 ;регистр состояния кнопок
.def count = r28 ;число операндов в таблице операндов
.def c_bit = r29 ;счетчик циклов умножения (деления)
.macro vvod ;ввод операнда
lpm ;считывание байта из flash-памяти в r0
mov @0,r0 ; и пересылка в регистр операнда
mov res, r0
adiw zl, 1 ;увеличение указателя адреса на 1
dec count
brne exit
ldi ZL,low(tabl_op*2) ;перезагрузка начала таблицы операндов
ldi ZH,high(tabl_op*2) ; в регистр Z
ldi count, 10 ;число заданных операндов в таблице 10
exit: nop
.endmacro
.org $000
;Инициализация стека, портов, адресного регистра Z
ldi temp,low(RAMEND) ;установка
out SPL,temp ; указателя стека
```

```

ldi temp,high(RAMEND) ; на последнюю
out SPH,temp ; ячейку ОЗУ
ser temp ;настройка
out DDRB,temp ; порта PB
out PORTB,temp ; на вывод
clr temp ;настройка
out DDRD,temp ; порта PD
ser temp ; на
out PORTD,temp ; ввод
ldi ZL,low(tabl_op*2) ;загрузка адреса таблицы операндов
ldi ZH,high(tabl_op*2) ; в регистр Z
ldi count,10 ;число операндов 10
;Опрос кнопок и выполнение заданных действий
LOOP: in sw_reg,PIND
sbrs sw_reg,0
rjmp f_op_AL
sbrs sw_reg,1

rjmp f_op_AH
sbrs sw_reg,2
rjmp f_op_BL
sbrs sw_reg,3
rjmp add_bin
sbrs sw_reg,4
rjmp sub_bin
sbrs sw_reg,5
rjmp mul_bin
sbrs sw_reg,6
rjmp div_bin
sbrs sw_reg,7
rjmp loop
mov res,show
rjmp outled
;Выборка 1-го операнда из таблицы операндов
f_op_AL: vvod op_AL
rjmp outled
;Выборка старшего байта 1-го операнда (при делении)
f_op_AH: vvod op_AH
rjmp outled
;Выборка 2-го операнда
f_op_BL: vvod op_BL
rjmp outled
;Сложение 8-разрядных операндов
add_bin: mov res,op_AL
add res,op_BL
in show,SREG ;выборка из регистра SREG
rjmp outled
;Вычитание 8-разрядных операндов
sub_bin: mov res,op_AL
sub res,op_BL
in show,SREG ;выборка из регистра SREG
rjmp outled
;Умножение 8-разрядных операндов
mul_bin: mul op_AL,op_BL
mov show,r1 ;копируем старший и
mov res,r0 ; младший байт произведения
rjmp outled
;Деление 16-разрядного числа на 8-разрядное
div_bin: sbrc op_AH,7 ;ошибки исходных данных

```

```

rjmp error
sbrc op_BL,7
rjmp error
tst op_BL ;ошибка при делении на 0
breq error
cp op_AH,op_BL ;ошибка при переполнении
brge error
clr res ;обнуляем частное
ldi c_bit,8 ; число итераций
mov copy_AH,op_AH
mov copy_AL,op_AL
L4: clc
rol copy_AL ;сдвиг
rol copy_AH ; делимого
lsl res ;сдвиг частного влево
sub copy_AH,op_BL ;вычитание делителя
brcs recov ;если остаток < 0,переход
inc res ; иначе добавить 1 в частное
rjmp L5
recov: add copy_AH,op_BL ;восстановление остатка
L5: dec c_bit
brne L4
mov show,copy_AH ;пересылка остатка
rjmp outled
error: clr temp ;сигнал об ошибке деления
out PORTB, temp
rcall delay
ser temp
out PORTB, temp
rjmp wait
outled: com res
out portb,res
rcall delay
wait: in sw_reg,PIND ;ждать, пока кнопка не отпущена
com sw_reg
brne wait
rjmp loop
; Задержка
DELAY: ldi r19,10
m1: ldi r20,1;250

m3: ldi r21,250
m2: dec r21
brne m2
dec r20
brne m3
dec r19
brne m1
ret
; Таблица операндов в шестнадцатеричном представлении
tabl_op: .db 0x9C, 0xF0, 0xF6, 0x2A, 0x6F, 0x5F, 0x8A, 0x5F,
0xF6, 0xB1

```

Задание 2.

Выполнить ряд примеров на сложение и вычитание, выбирая операнды слагаемых AL и BL нажатием кнопок SW0 и SW2. Объяснить результаты операций при нажатиях

кнопки SW3 (сложение) и SW4 (вычитание), рассматривая операнды как беззнаковые числа, затем как числа со знаком.

Результаты наблюдений показаны в таблице 1.

Таблица 1 - результаты операций сложения и вычитания

Число A_2/A_{10}	Число B_2/B_{10}	$A+B/A-B$	Признаки: H-S-V-N-Z-C
1001 1100 / 156 Беззнаковое	1111 0000 / 240 Беззнаковое	1000 1100 (140) 1010 1100 (172)	0 - - - 0 1 0 - - - 0 1
1111 0110 / 246 Беззнаковое	0010 1010 / 42 Беззнаковое	0010 0000 (32) 1100 1100 (204)	1 - - - 0 1 1 - - - 0 0
1000 1010 / -118 Со знаком	0101 1111 / 95 Со знаком	1110 1001 (-23) 0010 1011 (43)	110100 111000
1111 0110 / 10 Со знаком	1011 0001 / -79 Со знаком	1010 0111 (-89) 0100 0101 (69)	010101 000000

Задание 3.

Выполнить ряд примеров умножения 8-разрядных двоичных чисел.

Схема алгоритма умножения показана на рисунке 2.



Рисунок 2 – схема алгоритма умножения

Результаты работы программы показаны в таблице 2.

Таблица 2 - результаты умножения

Число A_2/A_{10}	Число B_2/B_{10}	Результат
1111 0000 / 240	1111 0110 / 246	1110 0110 1010 0000 (59040)
1001 1100 / 156	0010 1010 / 42	0001 1001 1001 1000 (6552)
0110 1111 / 111	1000 1010 / 138	0011 1011 1101 0110 (15318)

Задание 4.

Выполнить деление беззнаковых чисел, 16-разрядного делимого на 8-разрядный делитель, с восстановлением остатка при условиях, что делитель не равен 0 и его значение не вызовет переполнения, а также делимое и делитель заданы с нулевыми значениями старших разрядов.

Выполнить 2-3 примера на деление двоичных чисел, самостоятельно подобрав делимое и делитель.

Схема алгоритма деления представлена на рисунке 3.

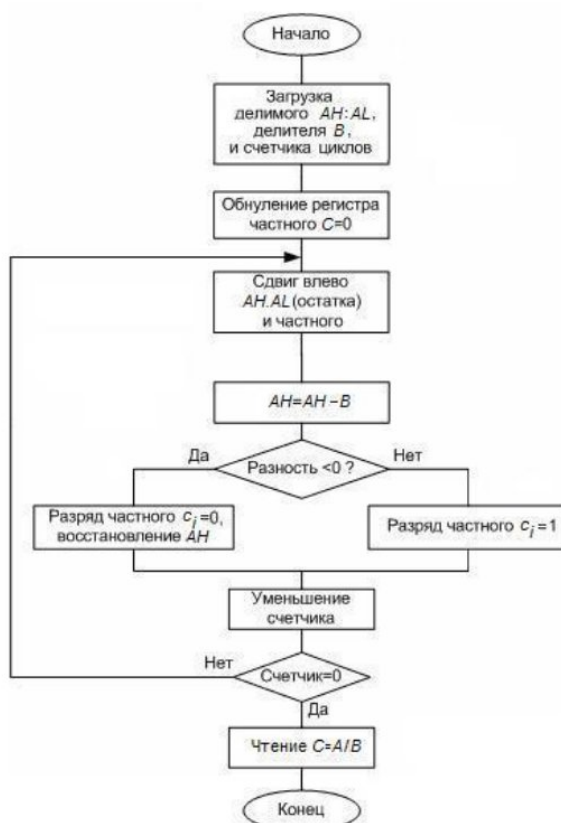


Рисунок 3 – схема алгоритма деления

Результаты деления представлены в таблице 3.

Таблица 3 – результаты деления

$(AH:AL)_2/(AH:AL)_{10}$	BL_2/BL_{10}	Частное	Остаток	Проверка
0111 1111 1000 0000 / 32640	1000 0000 / 128	1111 1111 / 255	0000 0000 / 0	32640 = 255*128 + 0
0010 1010 0110 1111 / 10863	0101 1111 / 95	0111 0010 / 114	0010 0001 / 33	10863 = 114*95 + 33
0001 0101 0000 0000 / 5376	0110 0100 / 100	0011 0101 / 53	0100 1100 / 76	5376 = 53*100 + 76

Работа алгоритма деления для делимого 5376_{10} и делителя 100_{10} :

$$5376_{10} = 0001\ 0101\ 0000\ 0000_2$$

$$100_{10} = 0110\ 0100_2$$

$$-100_{10} = 1001\ 1100_2 \text{ (дополнительный код)}$$

C – частное

$$\begin{array}{r}
 0001\ 0101\ 0000\ 0000 \\
 + \quad 1001\ 1100 \\
 \hline
 1011\ 0001 - \text{пробное вычитание В, так как разность меньше 0, то} \\
 \text{переполнения нет} \\
 \\
 0010\ 1010\ 0000\ 000x - \text{сдвиг А влево} \\
 + \quad 1001\ 1100 \\
 \hline
 1100\ 0110 - \text{вычитание В, остаток меньше 0, значит разряд частного (с7) = 0} \\
 \\
 0101\ 0100\ 0000\ 00xx \\
 + \quad 1001\ 1100 \\
 \hline
 1111\ 0000 - \text{вычитание В, остаток меньше 0, значит разряд частного (с6) = 0} \\
 \\
 1010\ 1000\ 0000\ 0xxx \\
 + \quad 1001\ 1100 \\
 \hline
 1\ 0100\ 0100 - \text{вычитание В, остаток больше 0, значит разряд частного (с5) = 1} \\
 \\
 1000\ 1000\ 0000\ xxxx \\
 + \quad 1001\ 1100 \\
 \hline
 1\ 0010\ 0100 - \text{вычитание В, остаток больше 0, значит разряд частного (с4) = 1} \\
 \\
 0100\ 1000\ 000x\ xxxx \\
 + \quad 1001\ 1100 \\
 \hline
 1110\ 0100 - \text{вычитание В, остаток меньше 0, значит разряд частного (с3) = 0} \\
 \\
 1001\ 0000\ 00xx\ xxxx \\
 + \quad 1001\ 1100 \\
 \hline
 1\ 0010\ 1100 - \text{вычитание В, остаток больше 0, значит разряд частного (с2) = 1} \\
 \\
 0101\ 1000\ 0xxx\ xxxx \\
 + \quad 1001\ 1100 \\
 \hline
 1111\ 0100 - \text{вычитание В, остаток меньше 0, значит разряд частного (с1) = 0} \\
 \\
 1011\ 0000\ xxxx\ xxxx \\
 + \quad 1001\ 1100 \\
 \hline
 \end{array}$$

1 0100 1100 - вычитание В, остаток больше 0, значит разряд частного (с0) = 1

$$C = 0011\ 0101_2 = 53_{10}$$

$$\text{Остаток } 0100\ 1100_2 = 76_{10}$$

Вывод: в ходе данной лабораторной работы были изучены способы представления числовых данных в микроконтроллерах, флаги арифметических операций, а также алгоритмы умножения и деления, ограничения для входных данных алгоритма деления.