**Министерство образования и науки Российской Федерации федеральное государственное бюджетное образовательное учреждение высшего образования «Казанский национальный исследовательский технический университет им. А.Н. Туполева-КАИ»**

Отделение СПО института компьютерных технологий и защиты информации

(09.02.03 — «ПРОГРАММИРОВАНИЕ В КОМПЬЮТЕРНЫХ СИСТЕМАХ»)

Отчет

По дисциплине “Микропроцессорные системы”

Вариант 2

Выполнил: студент группы 4441

Павлов И.С.

Проверил: преподаватель

Ширшова Д. В.

Дата\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Лабораторная работа №1.

1. Задание:
   1. Нахождение максимума и минимума массива чисел, вводимых с клавиатуры. Вывод максимума и минимума осуществляется в разные порты.
   2. Обнуление заданного элемента массива. Номер элемента вводится с клавиатуры в порт.
2. Решение:
3. Блок-схема

Задание 1:

Начало

DDRB=0x00;

DDRC=0xFF;

DDRD=0xFF;

int mass[10];

int i, j, min, max;

for(i=0;i<10;i++)

+

mass[i] = PINB;

max=mass[0];

min=mass[0];

for(j=0;j<10;j++)

+

if(mass[j]<min)

+

min=mass[j];

if(mass[j]>max)

+

max=mass[j];

PORTA = max;

PORTB = min;

Конец

Задание 2:

Начало

DDRA=0xFF;

DDRB=0x00;

DDRC=0x00;

int arr[10];

int i=0,num;

for(i=0;i<10;i++)

+

arr[i] = PINC;

num=PINB;

arr[num]=0;

for(i=0;i<10;i++)

+

PORTA=arr[i];

Конец

1. Листинг программы с комментариями

Задание 1:

#include <mega16.h>

void main(void)

{

// Declare your local variables here

int mass[10];

int min, max;

int i, j;S

// Input/Output Ports initialization

// Port A initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTA=0x00;

DDRA=0x00;

// Port B initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTB=0x00;

DDRB=0x00;

// Port C initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTC=0x00;

DDRC=0xFF;

// Port D initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTD=0x00;

DDRD=0xFF;

// Timer/Counter 0 initialization

// Clock source: System Clock

// Clock value: Timer 0 Stopped

// Mode: Normal top=FFh

// OC0 output: Disconnected

TCCR0=0x00;

TCNT0=0x00;

OCR0=0x00;

// Timer/Counter 1 initialization

// Clock source: System Clock

// Clock value: Timer 1 Stopped

// Mode: Normal top=FFFFh

// OC1A output: Discon.

// OC1B output: Discon.

// Noise Canceler: Off

// Input Capture on Falling Edge

// Timer 1 Overflow Interrupt: Off

// Input Capture Interrupt: Off

// Compare A Match Interrupt: Off

// Compare B Match Interrupt: Off

TCCR1A=0x00;

TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

// Timer/Counter 2 initialization

// Clock source: System Clock

// Clock value: Timer 2 Stopped

// Mode: Normal top=FFh

// OC2 output: Disconnected

ASSR=0x00;

TCCR2=0x00;

TCNT2=0x00;

OCR2=0x00;

// External Interrupt(s) initialization

// INT0: Off

// INT1: Off

// INT2: Off

MCUCR=0x00;

MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization

TIMSK=0x00;

// Analog Comparator initialization

// Analog Comparator: Off

// Analog Comparator Input Capture by Timer/Counter 1: Off

ACSR=0x80;

SFIOR=0x00;

while (1)

{

// Place your code here

for(i = 0; i < 10; i++){ // цикл for

mass[i] = PINB; // ввод элементов массива

}

min = mass[0]; // задаю минимальные и максимальные начальные значения для сравнения

max = mass[0];

for(j = 0; j < 10; j++){

if(min > mass[j]){ // проверяю меньше ли следующий элемент массива

min = mass[j]; // если меньше, то перезаписываю min

}

if(max < mass[j]){ проверяю больше ли следующий элемент

max = mass[j]; // если больше, то перезаписываю min

}

}

PORTC = min; // вывожу минимум

PORTD = max; // вывожу максимум

};

}

Задание 2:

#include <mega16.h>

void main(void)

{

int num, i;

int mass[10];

// Input/Output Ports initialization

// Port A initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTA=0x00;

DDRA=0x00;

// Port B initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTB=0x00;

DDRB=0x00;

// Port C initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTC=0x00;

DDRC=0x00;

// Port D initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTD=0x00;

DDRD=0xFF;

// Timer/Counter 0 initialization

// Clock source: System Clock

// Clock value: Timer 0 Stopped

// Mode: Normal top=FFh

// OC0 output: Disconnected

TCCR0=0x00;

TCNT0=0x00;

OCR0=0x00;

// Timer/Counter 1 initialization

// Clock source: System Clock

// Clock value: Timer 1 Stopped

// Mode: Normal top=FFFFh

// OC1A output: Discon.

// OC1B output: Discon.

// Noise Canceler: Off

// Input Capture on Falling Edge

// Timer 1 Overflow Interrupt: Off

// Input Capture Interrupt: Off

// Compare A Match Interrupt: Off

// Compare B Match Interrupt: Off

TCCR1A=0x00;

TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

// Timer/Counter 2 initialization

// Clock source: System Clock

// Clock value: Timer 2 Stopped

// Mode: Normal top=FFh

// OC2 output: Disconnected

ASSR=0x00;

TCCR2=0x00;

TCNT2=0x00;

OCR2=0x00;

// External Interrupt(s) initialization

// INT0: Off

// INT1: Off

// INT2: Off

MCUCR=0x00;

MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization

TIMSK=0x00;

// Analog Comparator initialization

// Analog Comparator: Off

// Analog Comparator Input Capture by Timer/Counter 1: Off

ACSR=0x80;

SFIOR=0x00;

while (1)

      {

      for(i = 0; i < 10; i++)

      {

        mass[i] = PINB;

      }

      num = PINC;

      mass[num] = 0;

      for(i = 0; i < 10; i++){

        PORTD = mass[i];

      }

      };

}

1. Результаты выполнения программы

Задание 1:

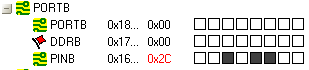


Рис. 1.1. 1. Ввод нулевого элемента массива

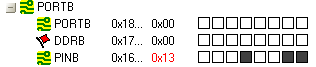


Рис. 1.1.2 Ввод первого элемента массива

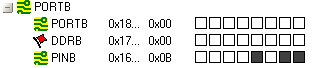


Рис. 1. 1. 3. Ввод второго элемента массива

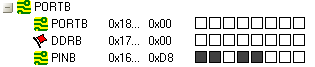


Рис. 1.1.4. Ввод третьего элемента массива

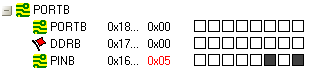


Рис. 1.1.5. Ввод четвертого элемента

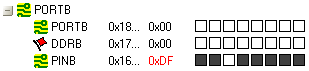


Рис. 1.1.6 Ввод пятого элемента массива

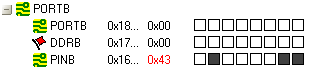


Рис. 1.1.7. Ввод шестого элемента массива

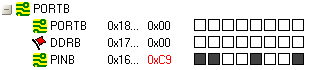


Рис. 1.1.8. Ввод седьмого элемента массива

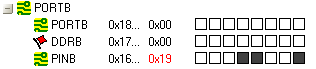


Рис. 1.1.9 Ввод восьмого элемента массива

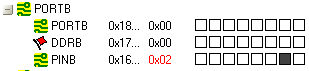


Рис. 1.1.10. Ввод девятого элемента массива

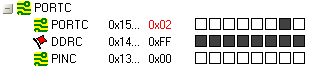


Рис. 1.1.11. Вывод минимального элемента массива

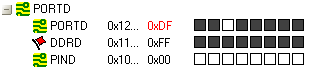


Рис. 1.1.12. Вывод максимального элемента массива

Задание 2:

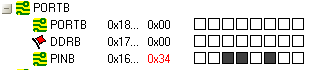


Рис. 1.2.1. Ввод нулевого элемента массива

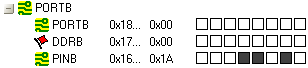


Рис. 1.2.2. Ввод первого элемента массива

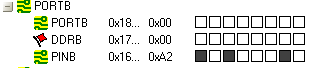


Рис. 1.2.3. Ввод второго элемента массива

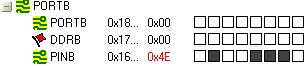


Рис. 1.2.4. Ввод третьего элемента массива

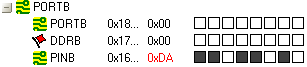


Рис. 1.2.5. Ввод четвертого элемента массива

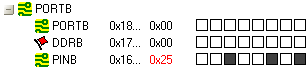


Рис. 1.2.6. Ввод пятого элемента массива

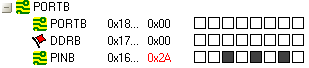


Рис. 1.2.7. Ввод шестого элемента массива

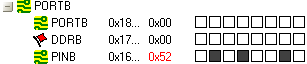


Рис. 1.2.8. Ввод седьмого элемента массива

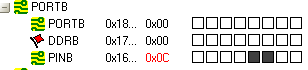


Рис. 1.2.9. Ввод восьмого элемента массива

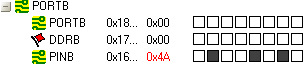


Рис. 1.2.10 Ввод девятого элемента массива

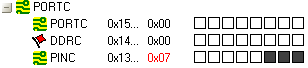


Рис. 1.2.11 Ввод номера элемента для удаления

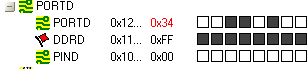


Рис. 1.2.12 Нулевой элемент преобразованного массива

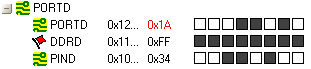


Рис. 1.2.13 Первый элемент преобразованного массива

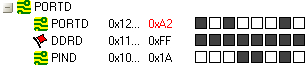


Рис. 1.2.14 Второй элемент преобразованного массива

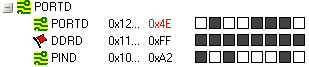


Рис. 1.2.15 Третий элемент преобразованного массива

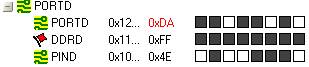


Рис. 1.2.16 Четвертый элемент преобразованного массива

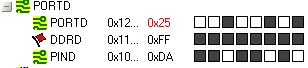


Рис. 1.2.17 Пятый элемент преобразованного массива

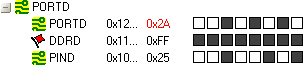


Рис. 1.2.18 Шестой элемент преобразованного массива

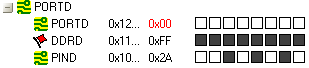


Рис. 1.2.19 Седьмой элемент преобразованного массива

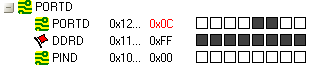


Рис. 1.2.20 Восьмой элемент преобразованного массива

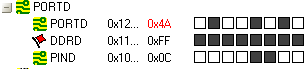
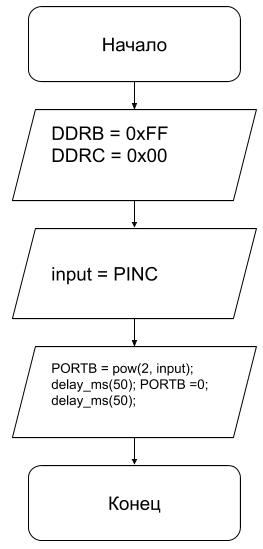


Рис. 1.2.21 Девятый элемент преобразованного массива

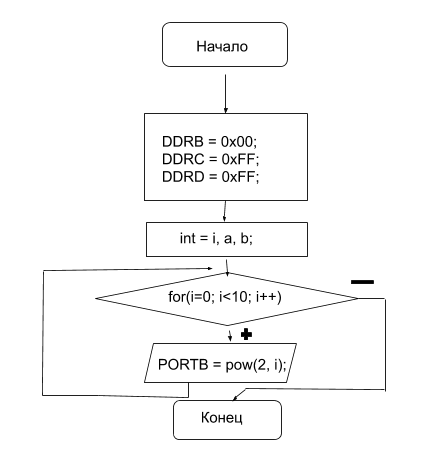
Лабораторная работа №2

1. Задание:
   1. Мигание светодиода. Номер вывода, к которому подключен светодиод вводится с порта С.
   2. Программа «Бегущий огонёк»
2. Решение:
3. Блок-схема

Задание 1:



Задание 2:



1. Листинг программы с комментариями

Задание 1:

#include <mega16.h>

#include <delay.h>

#include <math.h>

// Declare your global variables here

void main(void)

{

// Declare your local variables here

int input, outLED;

// Input/Output Ports initialization

// Port A initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTA=0x00;

DDRA=0x00;

// Port B initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTB=0x00;

DDRB=0xFF;

// Port C initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTC=0x00;

DDRC=0x00;

// Port D initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTD=0x00;

DDRD=0x00;

// Timer/Counter 0 initialization

// Clock source: System Clock

// Clock value: Timer 0 Stopped

// Mode: Normal top=FFh

// OC0 output: Disconnected

TCCR0=0x00;

TCNT0=0x00;

OCR0=0x00;

// Timer/Counter 1 initialization

// Clock source: System Clock

// Clock value: Timer 1 Stopped

// Mode: Normal top=FFFFh

// OC1A output: Discon.

// OC1B output: Discon.

// Noise Canceler: Off

// Input Capture on Falling Edge

// Timer 1 Overflow Interrupt: Off

// Input Capture Interrupt: Off

// Compare A Match Interrupt: Off

// Compare B Match Interrupt: Off

TCCR1A=0x00;

TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

// Timer/Counter 2 initialization

// Clock source: System Clock

// Clock value: Timer 2 Stopped

// Mode: Normal top=FFh

// OC2 output: Disconnected

ASSR=0x00;

TCCR2=0x00;

TCNT2=0x00;

OCR2=0x00;

// External Interrupt(s) initialization

// INT0: Off

// INT1: Off

// INT2: Off

MCUCR=0x00;

MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization

TIMSK=0x00;

// Analog Comparator initialization

// Analog Comparator: Off

// Analog Comparator Input Capture by Timer/Counter 1: Off

ACSR=0x80;

SFIOR=0x00;

input = PINC;

while (1)

      {

      PORTB = pow(2, input);

      delay\_ms(50);

      PORTB = 0;

      delay\_ms(50);

      };

}

Задание 2:

#include <mega16.h>

#include <math.h>

*// Declare your global variables here*

void main(void)

{

*// Declare your local variables here*

int i;

*// Input/Output Ports initialization*

*// Port A initialization*

*// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In*

*// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T*

PORTA=0x00;

DDRA=0x00;

*// Port B initialization*

*// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out*

*// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0*

PORTB=0x00;

DDRB=0xFF;

*// Port C initialization*

*// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In*

*// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T*

PORTC=0x00;

DDRC=0x00;

*// Port D initialization*

*// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In*

*// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T*

PORTD=0x00;

DDRD=0x00;

*// Timer/Counter 0 initialization*

*// Clock source: System Clock*

*// Clock value: Timer 0 Stopped*

*// Mode: Normal top=FFh*

*// OC0 output: Disconnected*

TCCR0=0x00;

TCNT0=0x00;

OCR0=0x00;

*// Timer/Counter 1 initialization*

*// Clock source: System Clock*

*// Clock value: Timer 1 Stopped*

*// Mode: Normal top=FFFFh*

*// OC1A output: Discon.*

*// OC1B output: Discon.*

*// Noise Canceler: Off*

*// Input Capture on Falling Edge*

*// Timer 1 Overflow Interrupt: Off*

*// Input Capture Interrupt: Off*

*// Compare A Match Interrupt: Off*

*// Compare B Match Interrupt: Off*

TCCR1A=0x00;

TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

*// Timer/Counter 2 initialization*

*// Clock source: System Clock*

*// Clock value: Timer 2 Stopped*

*// Mode: Normal top=FFh*

*// OC2 output: Disconnected*

ASSR=0x00;

TCCR2=0x00;

TCNT2=0x00;

OCR2=0x00;

*// External Interrupt(s) initialization*

*// INT0: Off*

*// INT1: Off*

*// INT2: Off*

MCUCR=0x00;

MCUCSR=0x00;

*// Timer(s)/Counter(s) Interrupt(s) initialization*

TIMSK=0x00;

*// Analog Comparator initialization*

*// Analog Comparator: Off*

*// Analog Comparator Input Capture by Timer/Counter 1: Off*

ACSR=0x80;

SFIOR=0x00;

while (1)

      {

*// Place your code here*

      for(i = 0; i < 10; i++){ *// цикл for*

        PORTB = pow(2, i); *//  вывожу в порт В  2 в степени i. Получается 1, 2, 4, 8, 16*

      }

      };

}

c)Результаты выполнения программы

Задание 1:

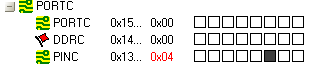


Рис. 2.1.1. Ввод номер вывода порта светодиода

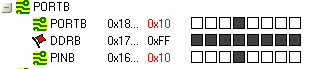


Рис. 2.1.2. Загорелся порт 4

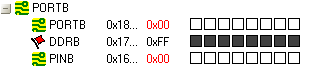


Рис. 2.1.3. Порт 4 погас.

Задание 2:

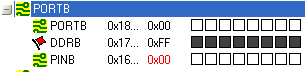


Рис. 2.2.1. Исходное состояние.

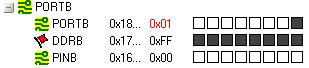


Рис. 2.2.2. Первый проход цикла.

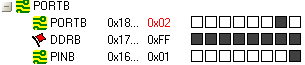


Рис. 2.2.3. Второй проход цикла

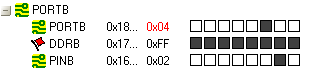


Рис. 2.2.4. Третий проход цикла

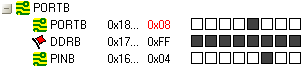
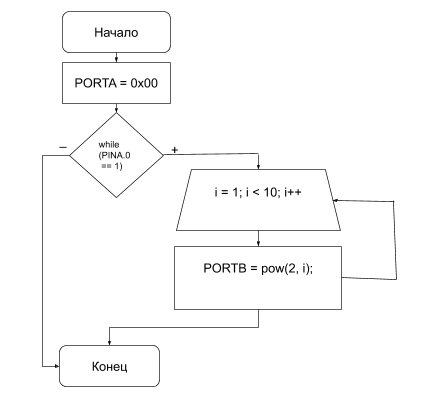


Рис. 2.2.5. Четвертый проход цикла.

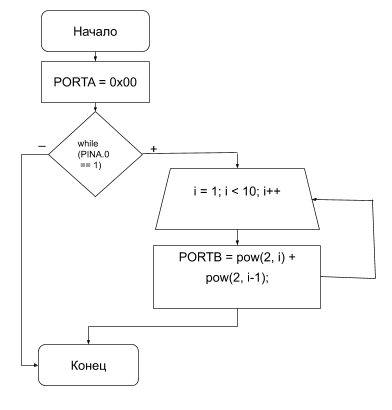
Лабораторная работа №3

1. Задание:
   1. Программа «Бегущий огонёк» стартует по нажатию PINX 0
   2. Программа «Бегущие глаза» стартует по нажатию PINX 0
2. Решение:
3. Блок-схема

Задание 1:



Задание 2:



1. Листинг кода с комментариями

Задание 1:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

This program was produced by the

CodeWizardAVR V1.25.9 Standard

Automatic Program Generator

� Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.

http://www.hpinfotech.com

Project :

Version :

Date    : 29.10.2019

Author  : F4CG

Company : F4CG

Comments:

Chip type           : ATmega16

Program type        : Application

Clock frequency     : 4,000000 MHz

Memory model        : Small

External SRAM size  : 0

Data Stack size     : 256

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <mega16.h>

#include <math.h>

// Declare your global variables here

void main(void)

{

// Declare your local variables here

int i;

// Input/Output Ports initialization

// Port A initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTA=0x00;

DDRA=0x00;

// Port B initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTB=0x00;

DDRB=0xFF;

// Port C initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTC=0x00;

DDRC=0x00;

// Port D initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTD=0x00;

DDRD=0x00;

// Timer/Counter 0 initialization

// Clock source: System Clock

// Clock value: Timer 0 Stopped

// Mode: Normal top=FFh

// OC0 output: Disconnected

TCCR0=0x00;

TCNT0=0x00;

OCR0=0x00;

// Timer/Counter 1 initialization

// Clock source: System Clock

// Clock value: Timer 1 Stopped

// Mode: Normal top=FFFFh

// OC1A output: Discon.

// OC1B output: Discon.

// Noise Canceler: Off

// Input Capture on Falling Edge

// Timer 1 Overflow Interrupt: Off

// Input Capture Interrupt: Off

// Compare A Match Interrupt: Off

// Compare B Match Interrupt: Off

TCCR1A=0x00;

TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

// Timer/Counter 2 initialization

// Clock source: System Clock

// Clock value: Timer 2 Stopped

// Mode: Normal top=FFh

// OC2 output: Disconnected

ASSR=0x00;

TCCR2=0x00;

TCNT2=0x00;

OCR2=0x00;

// External Interrupt(s) initialization

// INT0: Off

// INT1: Off

// INT2: Off

MCUCR=0x00;

MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization

TIMSK=0x00;

// Analog Comparator initialization

// Analog Comparator: Off

// Analog Comparator Input Capture by Timer/Counter 1: Off

ACSR=0x80;

SFIOR=0x00;

while (1)

      {

      // Place your code here

while(PORTA.0 == 1){

        for(i = 0; i < 10; i++){

          PORTB = pow(2, i);

        }

      }

      };

}

Задание 2:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

This program was produced by the

CodeWizardAVR V1.25.9 Standard

Automatic Program Generator

� Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.

http://www.hpinfotech.com

Project :

Version :

Date    : 29.10.2019

Author  : F4CG

Company : F4CG

Comments:

Chip type           : ATmega16

Program type        : Application

Clock frequency     : 4,000000 MHz

Memory model        : Small

External SRAM size  : 0

Data Stack size     : 256

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <mega16.h>

#include <math.h>

// Declare your global variables here

void main(void)

{

// Declare your local variables here

int i;

// Input/Output Ports initialization

// Port A initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTA=0x00;

DDRA=0x00;

// Port B initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTB=0x00;

DDRB=0xFF;

// Port C initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTC=0x00;

DDRC=0x00;

// Port D initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTD=0x00;

DDRD=0x00;

// Timer/Counter 0 initialization

// Clock source: System Clock

// Clock value: Timer 0 Stopped

// Mode: Normal top=FFh

// OC0 output: Disconnected

TCCR0=0x00;

TCNT0=0x00;

OCR0=0x00;

// Timer/Counter 1 initialization

// Clock source: System Clock

// Clock value: Timer 1 Stopped

// Mode: Normal top=FFFFh

// OC1A output: Discon.

// OC1B output: Discon.

// Noise Canceler: Off

// Input Capture on Falling Edge

// Timer 1 Overflow Interrupt: Off

// Input Capture Interrupt: Off

// Compare A Match Interrupt: Off

// Compare B Match Interrupt: Off

TCCR1A=0x00;

TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

// Timer/Counter 2 initialization

// Clock source: System Clock

// Clock value: Timer 2 Stopped

// Mode: Normal top=FFh

// OC2 output: Disconnected

ASSR=0x00;

TCCR2=0x00;

TCNT2=0x00;

OCR2=0x00;

// External Interrupt(s) initialization

// INT0: Off

// INT1: Off

// INT2: Off

MCUCR=0x00;

MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization

TIMSK=0x00;

// Analog Comparator initialization

// Analog Comparator: Off

// Analog Comparator Input Capture by Timer/Counter 1: Off

ACSR=0x80;

SFIOR=0x00;

while (1){

      // Place your code here

        while(PINA.0 == 1){

          for(i = 1; i < 10; i++){

            PORTB = pow(2, i) + pow(2, i-1);

          }

        }

      };

}

1. Результат выполнения программы

Задание 1:

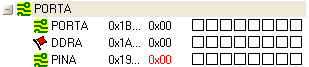


Рис. 3.1.1. На PINA.0 – ноль

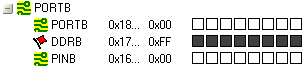


Рис. 3.1.2. На PORTB нет вывода.

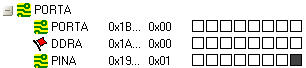


Рис. 3.1.3 Установил наPINA.0 – единицу

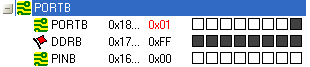


Рис. 3.1.4. Бегущий огонёк побежал

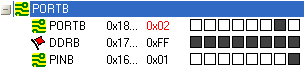


Рис. 3.1.5. Вторая итерация бегущего огонька.

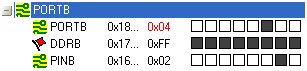


Рис. 3.1.6. Третья итерация бегущего огонька.

Задание 2:

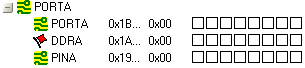


Рис. 3.2.1. На PINA сигнала нет

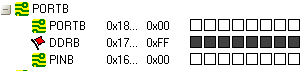


Рис. 3.2.2. Соответственно на PORTB не выводятся «Бегущие глазки»

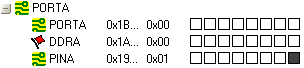


Рис. 3.2.3. На PINA подал 1

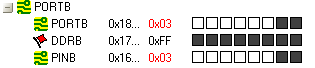


Рис. 3.2.4. Включились бегущие глазки.

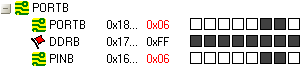


Рис. 3.2.5. Второе положение «глазок»

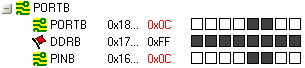


Рис. 3.2.6. Второе положение «глазок»

Лабораторная работа 4

1. Задание

Задание 1:

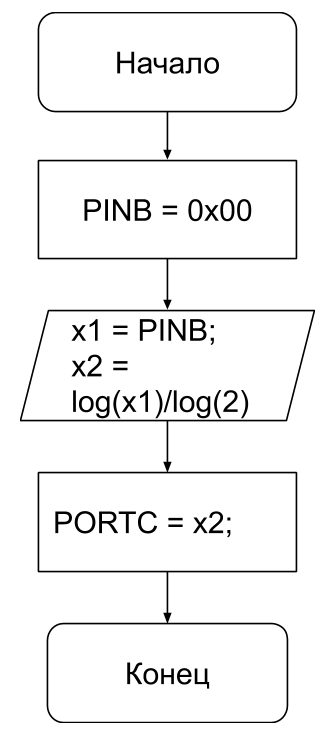
Номер нажатого вывода порта выводится в другой порт

Задание 2:

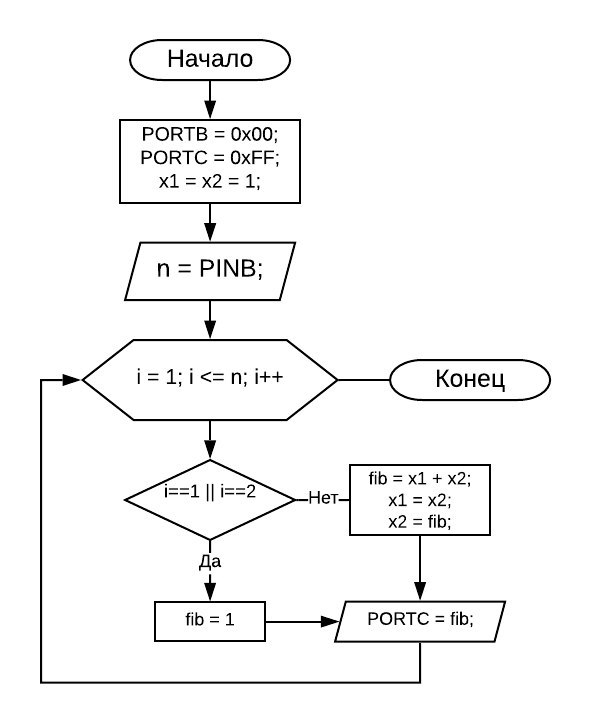
Генерация последовательности Фибоначчи

1. Решение
   1. Блок схемы

Задание 1:



Задание 2:



* 1. Листинг программы

Задание 1:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

This program was produced by the

CodeWizardAVR V1.25.9 Standard

Automatic Program Generator

� Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.

http://www.hpinfotech.com

Project :

Version :

Date    : 30.10.2019

Author  : F4CG

Company : F4CG

Comments:

Chip type           : ATmega16

Program type        : Application

Clock frequency     : 4,000000 MHz

Memory model        : Small

External SRAM size  : 0

Data Stack size     : 256

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <mega16.h>

#include <math.h>

// Declare your global variables here

void main(void)

{

// Declare your local variables here

int x1, x2;

// Input/Output Ports initialization

// Port A initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTA=0x00;

DDRA=0x00;

// Port B initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTB=0x00;

DDRB=0x00;

// Port C initialization

// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out

// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

PORTC=0x00;

DDRC=0xFF;

// Port D initialization

// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In

// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T

PORTD=0x00;

DDRD=0x00;

// Timer/Counter 0 initialization

// Clock source: System Clock

// Clock value: Timer 0 Stopped

// Mode: Normal top=FFh

// OC0 output: Disconnected

TCCR0=0x00;

TCNT0=0x00;

OCR0=0x00;

// Timer/Counter 1 initialization

// Clock source: System Clock

// Clock value: Timer 1 Stopped

// Mode: Normal top=FFFFh

// OC1A output: Discon.

// OC1B output: Discon.

// Noise Canceler: Off

// Input Capture on Falling Edge

// Timer 1 Overflow Interrupt: Off

// Input Capture Interrupt: Off

// Compare A Match Interrupt: Off

// Compare B Match Interrupt: Off

TCCR1A=0x00;

TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

// Timer/Counter 2 initialization

// Clock source: System Clock

// Clock value: Timer 2 Stopped

// Mode: Normal top=FFh

// OC2 output: Disconnected

ASSR=0x00;

TCCR2=0x00;

TCNT2=0x00;

OCR2=0x00;

// External Interrupt(s) initialization

// INT0: Off

// INT1: Off

// INT2: Off

MCUCR=0x00;

MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization

TIMSK=0x00;

// Analog Comparator initialization

// Analog Comparator: Off

// Analog Comparator Input Capture by Timer/Counter 1: Off

ACSR=0x80;

SFIOR=0x00;

while (1)

      {

      // Place your code here

      x1 = PINB;

      x2 = log(x1)/log(2);

      PORTC = x2;

      };

}

Задание 2:

*/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**

*This program was produced by the*

*CodeWizardAVR V1.25.9 Standard*

*Automatic Program Generator*

*� Copyright 1998-2008 Pavel Haiduc, HP InfoTech s.r.l.*

*http://www.hpinfotech.com*

*Project :*

*Version :*

*Date    : 06.11.2019*

*Author  : F4CG*

*Company : F4CG*

*Comments:*

*Chip type           : ATmega16*

*Program type        : Application*

*Clock frequency     : 4,000000 MHz*

*Memory model        : Small*

*External SRAM size  : 0*

*Data Stack size     : 256*

*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/*

#include <mega16.h>

*// Declare your global variables here*

void main(void)

{

*// Declare your local variables here*

int n, i, x1, x2, fib;

*// Input/Output Ports initialization*

*// Port A initialization*

*// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In*

*// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T*

PORTA=0x00;

DDRA=0x00;

*// Port B initialization*

*// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In*

*// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T*

PORTB=0x00;

DDRB=0x00;

*// Port C initialization*

*// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out Func0=Out*

*// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0*

PORTC=0x00;

DDRC=0xFF;

*// Port D initialization*

*// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In*

*// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T*

PORTD=0x00;

DDRD=0x00;

*// Timer/Counter 0 initialization*

*// Clock source: System Clock*

*// Clock value: Timer 0 Stopped*

*// Mode: Normal top=FFh*

*// OC0 output: Disconnected*

TCCR0=0x00;

TCNT0=0x00;

OCR0=0x00;

*// Timer/Counter 1 initialization*

*// Clock source: System Clock*

*// Clock value: Timer 1 Stopped*

*// Mode: Normal top=FFFFh*

*// OC1A output: Discon.*

*// OC1B output: Discon.*

*// Noise Canceler: Off*

*// Input Capture on Falling Edge*

*// Timer 1 Overflow Interrupt: Off*

*// Input Capture Interrupt: Off*

*// Compare A Match Interrupt: Off*

*// Compare B Match Interrupt: Off*

TCCR1A=0x00;

TCCR1B=0x00;

TCNT1H=0x00;

TCNT1L=0x00;

ICR1H=0x00;

ICR1L=0x00;

OCR1AH=0x00;

OCR1AL=0x00;

OCR1BH=0x00;

OCR1BL=0x00;

*// Timer/Counter 2 initialization*

*// Clock source: System Clock*

*// Clock value: Timer 2 Stopped*

*// Mode: Normal top=FFh*

*// OC2 output: Disconnected*

ASSR=0x00;

TCCR2=0x00;

TCNT2=0x00;

OCR2=0x00;

*// External Interrupt(s) initialization*

*// INT0: Off*

*// INT1: Off*

*// INT2: Off*

MCUCR=0x00;

MCUCSR=0x00;

*// Timer(s)/Counter(s) Interrupt(s) initialization*

TIMSK=0x00;

*// Analog Comparator initialization*

*// Analog Comparator: Off*

*// Analog Comparator Input Capture by Timer/Counter 1: Off*

ACSR=0x80;

SFIOR=0x00;

while (1)

      {

*// Place your code here*

      x1 = 1;

      x2 = 1;

      n = PINB;

      for (i = 1; i <= n; i++)

      {

        if (i==1 || i==2) fib = 1;

        else

        {

          fib = x1 + x2;

          x1 = x2;

          x2 = fib;

        }

        PORTC = fib;

      }

      };

}

1. Результат выполнения программы

Задание 1:

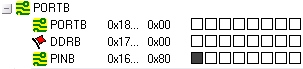


Рис. 4.1.1 Нажал седьмой выход порта



Рис. 4.1.2. Получил цифру семь в PORTC.

Задание 2:

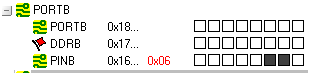


Рис. 4.2.1. Ввел количество требуемых чисел Фибоначчи

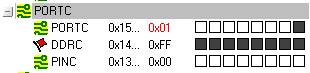


Рис. 4.2.2. Первое число Фибоначчи

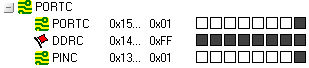


Рис. 4.2.3. Второе число Фибоначчи

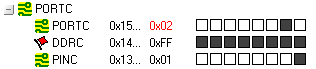


Рис. 4.2.4. Третье число Фибоначчи

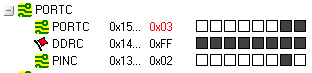


Рис. 4.2.5. Четвертое число Фибоначчи

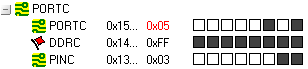


Рис. 4.2.6. Пятое число Фибоначчи

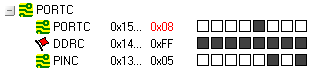


Рис. 4.2.7. Шестое число Фибоначчи