**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра САУ**

**Лабораторная работа №5**

**по дисциплине «МПУ»**

**Вариант 7**

|  |  |  |
| --- | --- | --- |
| Студенты гр. 6 |  |  |
| Преподаватель |  |  |

Санкт-Петербург

2018

**Цель работы**. Программирование процедур вывода символьной инфор-мации на жидкокристаллический индикатор с использованием параллельного порта МК.

**Задание:** задано дискретное дифференциальное уравнение где *u*(*k*) и *y*(*k*) – значения входного воздействия и управляющего сигнала в дискретные моменты времени *t* = *kT*, где *T* – период прерывания (шаг расчета).

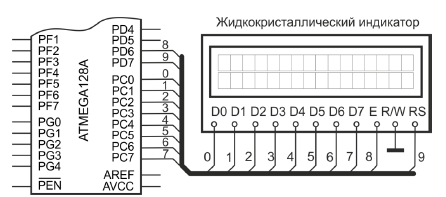
В данной лабораторной работе необходимо реализовать это соотношение в виде программы для расчета управляющего сигнала *y*(*k*) на пяти шагах при начальном условии *y*(0) = 0 и известных значениях *u*(*k*) на каждом шаге. Для индикации полученных значений *y*(*k*) необходимо на каждом шаге выводить их на индикатор. Формат вывода: Результат: Y(k) = X, где X – выводимое число, а k – номер шага.

D:\education\5 сем\Микропроцессоры\7_VAR\3.png

*Таблица 1. Данные задания.*

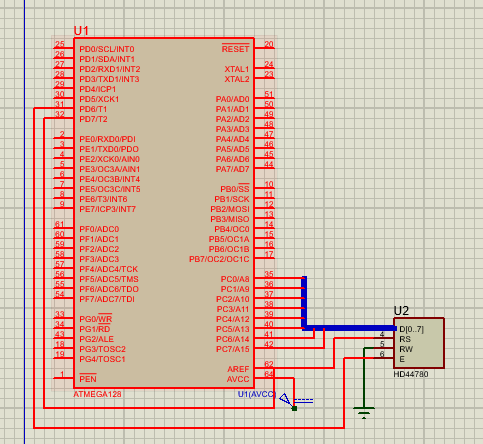
**

*Рис.1 Отображение символов из DDRAM в окне индикатора.*

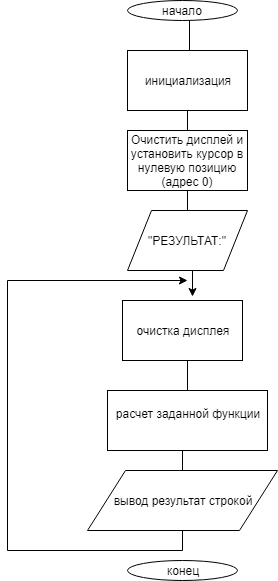
**

*Рис.2 Схема подключения ЖКИ и МК*

*Proteus*

**

*Блок-схема*

**

**Код программы.**

/\*

Variant 7

lab 5

\*/

#include <io.h>

#include <mega128a.h>

#include <delay.h>

#define CMD 1

#define DATA 0

#define DISPLAY 1

#define CURSOR 0

#define RHIGHT 1

#define LEFT 0

#define ON 1

#define OFF 0

#define EIGHT 1

#define FOUR 0

#define ONE 0

#define TWO 1

#define EIGHT 1

#define FOUR 0

#define LOWERCASE 0

#define UPPERCASE 1

#define RS 7 // выбор регистра

#define E 6 // строб передачи

//прототипы функции для работы с дисплеем

void LCD\_message(char cmd,int type );

void clear\_display() ;

void default\_display() ;

void shift\_direction(int ID,int S);

void switch\_display(int B,int D,int C);

void shift(int choose,int direction);

void display\_setting(int data\_bus\_width,int line\_number,int font);

void LCD\_init(void) ;

void LCD\_string(unsigned char \*str);

void LCD\_int(int value);

void LCD\_data\_string(int data, unsigned char i) ;

unsigned char Code(unsigned char symb);

unsigned char Digit (unsigned int d, unsigned char m);

void main(void)

{

unsigned char T=1;

int i, y, u[5]={8, 28, 48, 68, 88};

DDRC = 0xFF; // все разряды PORTC на выход

DDRD|= ((1<<E)|(1<<RS));// разряды PORTD на выход

LCD\_init(); //инициализация

clear\_display(); //Очистить дисплей и установить курсор в нулевую позицию (адрес 0)

LCD\_string("РЕЗУЛЬТАТ:");

while (1)

{

clear\_display() ;

y=0;

for (i=0; i<5; i++)

{

y = y + T \* (-5\*y+u[i]/4-1); //расчет заданной функции

LCD\_data\_string(y, i); //вывод результат строкой i - аргумент функции y - значение

delay\_ms(1000);

}

}

}

//Вывод числового значения (max 5 десятичных разрядов и больше нуля)

void LCD\_int(int value)

{

int i ;

unsigned char flag\_first\_num = 0 ;

unsigned char number;

if (value<0) LCD\_message(0b10010110,DATA);//выводим минус

for(i=1; i<=5; i++)

{

number = Digit(value,i);

if(number != 0) //появление перового символа

{

flag\_first\_num = 1;

}

if(flag\_first\_num == 1)

{

LCD\_message(number+'0',DATA);//выводим цифру

}

}

}

void LCD\_message(char message,int type)

{

//[]--------------------------------------------------[]

//| Назначение: запись кодов в регистр команд ЖКИ |

//| Входные параметры: message - сообщение |

//| Входные параметры: type - тип сообщения (код или данные) |

//[]--------------------------------------------------[]

if(type) // 1

PORTD &= ~(1<<RS); // выбор регистра команд RS=0

else

PORTD |= (1<<RS); // выбор регистра данных RS=1

PORTC=message; // записать команду в порт PORTC

PORTD|= (1<<E); // \ сформироватьна

delay\_us(5); // |выводе E строб 1-0

PORTD&= ~(1<<E); // / передачи команды

delay\_ms(100); // задержка для завершения записи

}

void LCD\_init(void)

{

//[]--------------------------------------------------[]

//| Назначение: инициализация ЖКИ |

//[]--------------------------------------------------[]

delay\_ms (100); // задержка для установления

// напряжения питания

LCD\_message(0x30,CMD); // \ вывод

LCD\_message(0x30,CMD); //| трех

LCD\_message(0x30,CMD); // / команд 0x30

LCD\_message(0x38,CMD); //8 разр.шина, 2 строки, 5 ? 7 точек

LCD\_message(0x0E,CMD); // включить ЖКИ и курсор, без мерцания

LCD\_message(0x06,CMD); //инкремент курсора, без сдвига экрана

LCD\_message(0x01,CMD); // очистить экран, курсор в начало

}

void LCD\_data\_string(int data, unsigned char i)

{

unsigned char STR[5] ={'Y','(', ')', '=', '\0'};

unsigned char k=0;

while (k<5)

{

if (k==2) LCD\_message(i+'0',DATA); // вывод аргумента функции

if (k==4)

{

LCD\_int(data);

break;

}

LCD\_message(STR[k],DATA); //оформление

k++;

delay\_ms(50);

}

}

//вывод строковой информации на дисплей

void LCD\_string(unsigned char\* str)

{

while(\*str != '\0')

{

LCD\_message(Code(\*str++),DATA);

//\*str извлечение элемента по адресу в указателе str

//str++ увеличение значение указателя на 1 (т.е. переход к следующему элементу массива)

}

}

unsigned char Digit (unsigned int d, unsigned char m)

{

//[]-----------------------------------------------------[]

//| Назначение: выделение цифр из разрядов пятиразрядного |

//| десятичного положительного числа |

//| Входные параметры: |

//| d - целое десятичное положительное число |

//| m - номер разряда (от 1 до 5, слева направо) |

//| Функция возвращает значение цифры в разряде m числа d |

//[]-----------------------------------------------------[]

unsigned char i = 5, a;

while(i)

{

a = d%10; //если d < 0 то a<0

if(i-- == m) break;

d /= 10;

}

return(a);

}

unsigned char Code(unsigned char symb)

{

//[]------------------------------------------------[]

//| Назначение: перекодировка символов кириллицы |

//| Входные параметры: symb – символ ASCII |

//| Функция возвращает код отображения символа |

//[]------------------------------------------------[]

unsigned char TabCon[] =

{0x41,0xA0,0x42,0xA1,0xE0,0x45,0xA3,0xA4,0xA5,0xA6,0x4B, 0xA7,0x4D,0x48,0x4F,0xA8,0x50,0x43,0x54,0xA9,0xAA,0x58,

0xE1,0xAB,0xAC,0xE2,0xAD,0xAE,0x62,0xAF,0xB0,0xB1,0x61,

0xB2,0xB3,0xB4,0xE3,0x65,0xB6,0xB7,0xB8,0xB9,0xBA,0xBB,

0xBC,0xBD,0x6F,0xBE,0x70,0x63,0xBF,0x79,0x5C,0x78,0xE5,

0xC0,0xC1,0xE6,0xC2,0xC3,0xC4,0xC5,0xC6,0xC7};//коды символов в кириллице

return (symb >= 192 ? TabCon[symb-192]: symb);

}

//очистить дисплей и установить курсор в нулевую позицию(адрес 0 )

void clear\_display()

{

LCD\_message(1,CMD);

};

//установить курсор в нулевую позицию, дисплей относительно буфера DDRAM в начальную позицию

void default\_display()

{

LCD\_message(2,CMD);

};

//установить направление сдвига курсора при записи кода в DDRAM

// и разрешить(запретить) сдвиг окна вместе с курсором

// ID = RIGHT

// ID = LEFT

// S = ON разрешить сдвиг окна вместе с курсором

// S = OFF запретить сдвиг окна вместе с курсором

void shift\_direction(int ID,int S)

{

LCD\_message((1<<2)|(ID<<1)|S,CMD);

}

//Включить(выключить) индикатор,зажечь(погасить) курсор.Сделать курсор мигающим

// B = ON курсор мигает

// B = OFF курсор не мигает

// D = ON включить индикатор

// D = OFF выключить индикатор

// C = ON зажечь курсор

// C = OFF погасить курсор

void switch\_display(int B,int D,int C)

{

LCD\_message((1<<3)|(D<<2)|(C<<1)|B,CMD);

}

//сдвиг курсора или дисплея вправо или влево

// choose = DISPLAY

// choose = CURSOR

// direction = RIGHT

// direction = LEFT

void shift(int choose,int direction)

{

LCD\_message((1<<4)|(choose<<3)|(direction<<2),CMD);

}

//установить разрядность шины данных, количество строк, шрифт

//data\_bus\_width = EIGHT 8 бит

//data\_bus\_width = FOUR 4 бит

//line\_number = ONE 1 строка

//line\_number = TWO 2 строки

//font = LOWERCASE строчные

////font = UPPERCASE заглавные

void display\_setting(int data\_bus\_width,int line\_number,int font)

{

LCD\_message((1<<5)|(data\_bus\_width<<4)|(line\_number<<3)|(font<<2),CMD);

}

Вывод: Программирование процедур вывода символьной инфор-мации на жидкокристаллический индикатор с использованием параллельного порта МК.