**ЛАБОРАТОРНА РОБОТА №1**

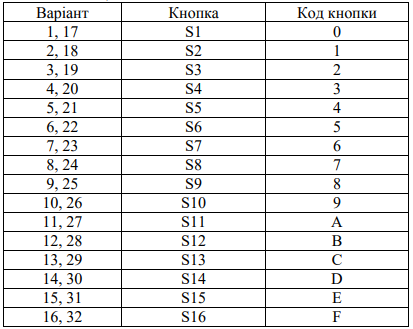
**ОРГАНІЗАЦІЯ ВВЕДЕННЯ ІНФОРМАЦІЇ ЗА ДОПОМОГОЮ КЛАВІАТУРИ**

***Мета заняття:*** ознайомитися з принципами побудови клавіатури. Отримати практичні навички по підключенню клавіатури до мікроконтролерів та організації її опитування

**Хід роботи:**

**Завдання 1:** Реалізувати опитування однієї кнопки клавіатури (у відповідності варіанту див. таблицю 1.1). При натисканні виконати зсув інформації на семисегментному індикаторі та виведення коду натиснутої кнопки у шістнадцятковій системі в останньому розряді індикатору.

Таблиця 1.1



**Завдання 2**: Реалізувати опитування усіх кнопок клавіатури. При натисканні виконати зсув інформації на семисегментному індикаторі та виведення коду натиснутої кнопки у шістнадцятковій системі в останньому розряді індикатору (див. таблицю 1.1).

int PinOut[4] {2, 3, 4, 5}; *// пины выходы*

int PinIn[4] {9, 8, 7, 6}; *// пины входа*

int val = 0;

const char value[4][4]

{ {'1', '4', '7', '\*'},

{'2', '5', '8', '0' },

{'3', '6', '9', '#'},

{'A', 'B', 'C', 'D'}

};

*// двойной массив, обозначающий кнопку*

int b = 0; *// переменная, куда кладется число из массива(номер кнопки)*

void setup()

{

pinMode (2, **OUTPUT**); *// инициализируем порты на выход (подают нули на столбцы)*

pinMode (3, **OUTPUT**);

pinMode (4, **OUTPUT**);

pinMode (5, **OUTPUT**);

pinMode (6, **INPUT**); *// инициализируем порты на вход с подтяжкой к плюсу (принимают нули на строках)*

digitalWrite(6, **HIGH**);

pinMode (7, **INPUT**);

digitalWrite(7, **HIGH**);

pinMode (8, **INPUT**);

digitalWrite(8, **HIGH**);

pinMode (9, **INPUT**);

digitalWrite(9, **HIGH**);

Serial.begin(9600); *// открываем Serial порт*

}

void matrix () *// создаем функцию для чтения кнопок*

{

**for** (int i = 1; i <= 4; i++) *// цикл, передающий 0 по всем столбцам*

{

digitalWrite(PinOut[i - 1], **LOW**); *// если i меньше 4 , то отправляем 0 на ножку*

**for** (int j = 1; j <= 4; j++) *// цикл, принимающих 0 по строкам*

{

**if** (digitalRead(PinIn[j - 1]) == **LOW**) *// если один из указанных портов входа равен 0, то..*

{

Serial.println( value[i - 1][j - 1]); *// то b равно значению из двойного массива*

delay(175);

}

}

digitalWrite(PinOut[i - 1], **HIGH**); *// подаём обратно высокий уровень*

}

}

void loop()

{

matrix(); *// используем функцию опроса матричной клавиатуры*

*}*

**Завдання 3**: Виконати завдання 2, виконуючі роботу з портами за допомогою регістрів МК.

#define LCHCLK 10 *//линия синхронизации*

#define STFCLK 11 *//линия тактирования*

#define SDI 12 *//линия передачи данных*

const int PinOut[4] {5, 4, 3, 2}; *// пины выходы*

const int PinIn[4] {6, 7, 8, 9}; *// пины входа*

int col=0,row=0,tmp=3;

int old[4][2]={{0,0},{0,0},{0,0},{0,0}};

bool check=true;

const byte values[4][4]

{{0xC0,0xF9,0xA4,0xB0},

{0x99,0x92,0x82,0xF8},

{0x80,0x90,0x88,0x83},

{0xC6,0xA1,0x86,0x8E},

};

const byte SEGMENT\_SELECT[] = {0xF1,0xF2,0xF4,0xF8};

void setup ()

{

pinMode(LCHCLK,**OUTPUT**);

pinMode(STFCLK,**OUTPUT**);

pinMode(SDI,**OUTPUT**);

pinMode (2, **OUTPUT**); *// инициализируем порты на выход (подают нули на столбцы)*

pinMode (3, **OUTPUT**);

pinMode (4, **OUTPUT**);

pinMode (5, **OUTPUT**);

pinMode (6, **INPUT**); *// инициализируем порты на вход с подтяжкой к плюсу (принимают нули на строках)*

digitalWrite(6, **HIGH**);

pinMode (7, **INPUT**);

digitalWrite(7, **HIGH**);

pinMode (8, **INPUT**);

digitalWrite(8, **HIGH**);

pinMode (9, **INPUT**);

digitalWrite(9, **HIGH**);

Serial.begin(9600);

}

void loop(){

**for**(int i=3; i>=0; i--){

matrix();

WriteNumberToSegment(i,old[i][0],old[i][1]);

}

}

*/\* Write a decimal number between 0 and 9 to one of the 4 digits of the display \*/*

void WriteNumberToSegment(byte Segment, byte Value1, byte Value2)

{

digitalWrite(LCHCLK,**LOW**);

shiftOut(SDI, STFCLK, MSBFIRST, values[Value1][Value2]);

shiftOut(SDI, STFCLK, MSBFIRST, SEGMENT\_SELECT[Segment] );

digitalWrite(LCHCLK,**HIGH**);

}

void Write(){

}

int matrix() *// создаем функцию для чтения кнопок*

{

**for** (int i = 0; i <= 3; i++) *// цикл, передающий 0 по всем столбцам*

{

digitalWrite(PinOut[i], **LOW**); *// если i меньше 4 , то отправляем 0 на ножку*

**for** (int j = 0; j <= 3; j++) *// цикл, принимающих 0 по строкам*

{

**if** (digitalRead(PinIn[j]) == **LOW**) *// если один из указанных портов входа равен 0, то..*

{

old[0][0]=old[1][0];

old[0][1]=old[1][1];

old[1][0]=old[2][0];

old[1][1]=old[2][1];

old[2][0]=old[3][0];

old[2][1]=old[3][1];

old[3][0]=col = i;

old[3][1]=row = j;

delay(175);

}

}

digitalWrite(PinOut[i], **HIGH**); *// подаём обратно высокий уровень*

}

}