

Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

**Лабораторна робота №3**

“Робота з РК дисплеєм”

з дисципліни “Архітектура Комп’ютерів – 3. Мікропроцесорні системи”

Виконав:

студент 3 курсу групи ІО-52

Бояршин Ігор

Номер заліковки: 5207

Перевірив:

Стешин В. В.

Київ 2018 р.

# ***Мета роботи:*** вивчення схеми відображення на основі РК дисплеїв.

**Завдання**

Номер залікової книжки: 520710 = 10100010**10111**2

**h3h2h1 = 111**

Відображати на дисплеї лічільник натиснень кнопки SW15 в шістнадцятковому форматі.

**h5h4 = 10**

Реалізувати введення uint\_16 з клавіатури. Відображати проміжне значення на дисплеї.

**Код програми**

#include ".\lab3.h"

#include <AT89LP51.h>

#include <stdlib.h>

#include <8051.h>

#include "ev8031.lib\bitdef.h"

#include "ev8031.lib\ev8031.c"

const unsigned char NO\_KEY\_CODE = 12;

const char HEX\_DIGITS[] = {

0x30,

0x31,

0x32,

0x33,

0x34,

0x35,

0x36,

0x37,

0x38,

0x39,

0x41,

0x42,

0x43,

0x44,

0x45,

0x46

};

//-----------------------------------------------------------------------------

void make\_delay16(uint16\_t);

void get\_key();

void display\_char(unsigned char code);

void display\_char(uint8\_t x, uint8\_t y, unsigned char code);

void display\_number(uint8\_t y, uint16\_t number);

uint8\_t get\_amount\_of\_digits(uint16\_t number);

void clear\_row(uint8\_t row);

//-----------------------------------------------------------------------------

void main(void)

{

unsigned char key;

uint16\_t press\_counter;

bool part\_1\_done;

bool part\_2\_done;

uint16\_t current\_number;

hd\_turn\_on();

hd\_init();

make\_delay16(160);

while(true) {

press\_counter = 0;

part\_1\_done = 0; // false

part\_2\_done = 0; // false

current\_number = 0;

clear\_row(3);

display\_number(3, press\_counter);

while (!part\_1\_done) {

key = get\_key();

if (key == NO\_KEY\_CODE) {

continue;

}

if (key == 15) {

press\_counter++;

display\_number(3, press\_counter);

} else if (key == 0) {

part\_1\_done = 1; // true

}

}

clear\_row(0);

display\_number(0, current\_number);

while (!part\_2\_done) {

display\_number(0, current\_number);

key = get\_key();

if (key == NO\_KEY\_CODE) {

continue;

}

switch (key) {

case 10:

part\_2\_done = 1; // true;

break;

default:

current\_number \*= 10;

current\_number += key;

display\_number(0, current\_number);

break;

}

}

}

}

//-----------------------------------------------------------------------------

void make\_delay16(uint16\_t millis) {

uint16\_t it;

for (it = 0; it < millis; it++) {}

}

unsigned char get\_key() {

const uint8\_t DELAY = 10;

uint8\_t i;

unsigned char new\_scan;

unsigned char previous\_scan;

previous\_scan = key\_scan();

if (previous\_scan == 12) { // no key

return NO\_KEY\_CODE;

}

// Make a delay of ~12ms

for (i = 0; i < DELAY; i++) {

display\_current(); // ~600 ops

}

new\_scan = key\_scan();

if (previous\_scan == new\_scan) {

do {

display\_current();

new\_scan = key\_scan(); // wait for user to release the key

} while (new\_scan != 12);

return previous\_scan;

} else {

return NO\_KEY\_CODE;

}

}

void display\_char(unsigned char code) {

make\_delay16(160);

LCD\_DATA = code;

}

void display\_char(uint8\_t x, uint8\_t y, unsigned char code) {

make\_delay16(160);

set\_cursor(x, y);

make\_delay16(160);

LCD\_DATA = code;

}

void display\_number(uint8\_t y, uint16\_t number) {

uint16\_t digit;

uint8\_t x;

x = 10 - get\_amount\_of\_digits(number);

make\_delay16(160);

set\_cursor(x, y);

while (number > 0) {

digit = number % 10;

number /= 10;

display\_char(HEX\_DIGITS[digit]);

}

}

uint8\_t get\_amount\_of\_digits(uint16\_t number) {

uint8\_t counter;

while (number > 0) {

number /= 10;

cunter++;

}

return counter;

}

void clear\_row(uint8\_t row) {

uint8\_t i;

make\_delay16(160);

set\_cursor(i, row);

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

display\_char(0x20); // empty char

}

}