

German-Russian Institute of Advanced Technologies  
TU-Ilmenau (Germany) and KNTRU-KAI (Kazan, Russia)

Computational-Graphic work of the subject  
«Computer systems»  
«Application for Keypad and LCD Security System.  
Analog Digital Conversion»

Student: Kukarcev K.S.

Teacher: \_\_\_\_\_

Kazan 2018

# Application for Keypad and LCD Security

## Variant №3

Password is 42547

EEPROM Memory address is 0x35

Delay is 780

Task is to implement access to LCD by password security with given parameters.

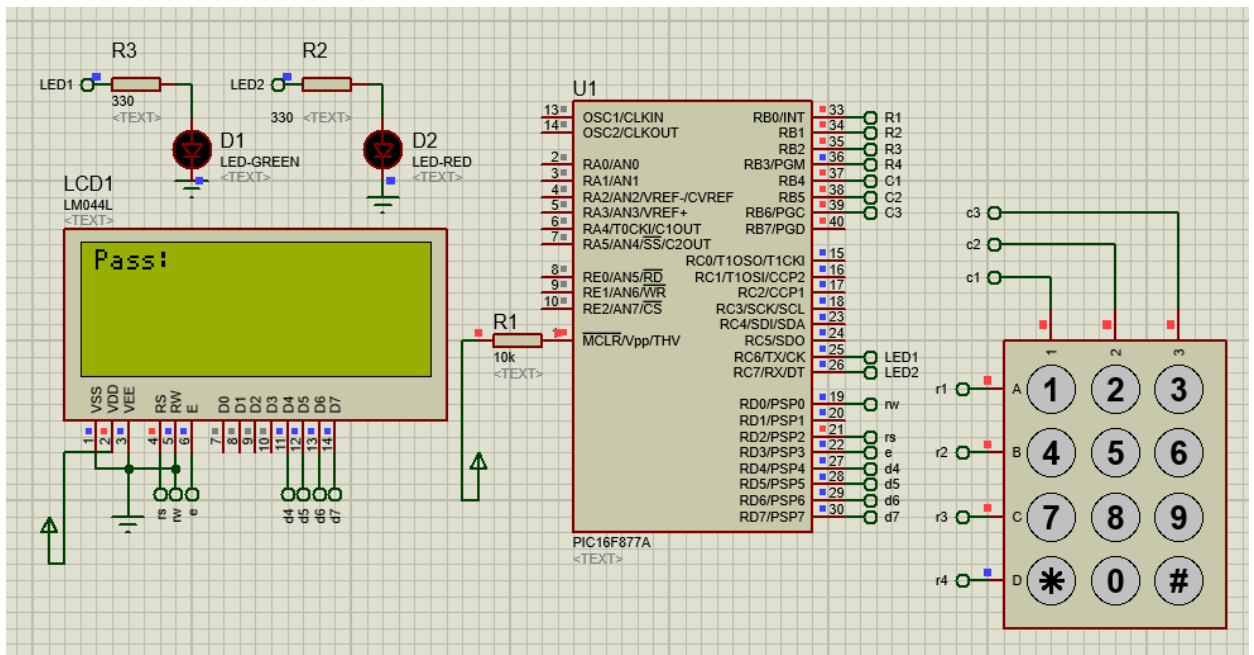


Image 1. The scheme of LCD password security circuit

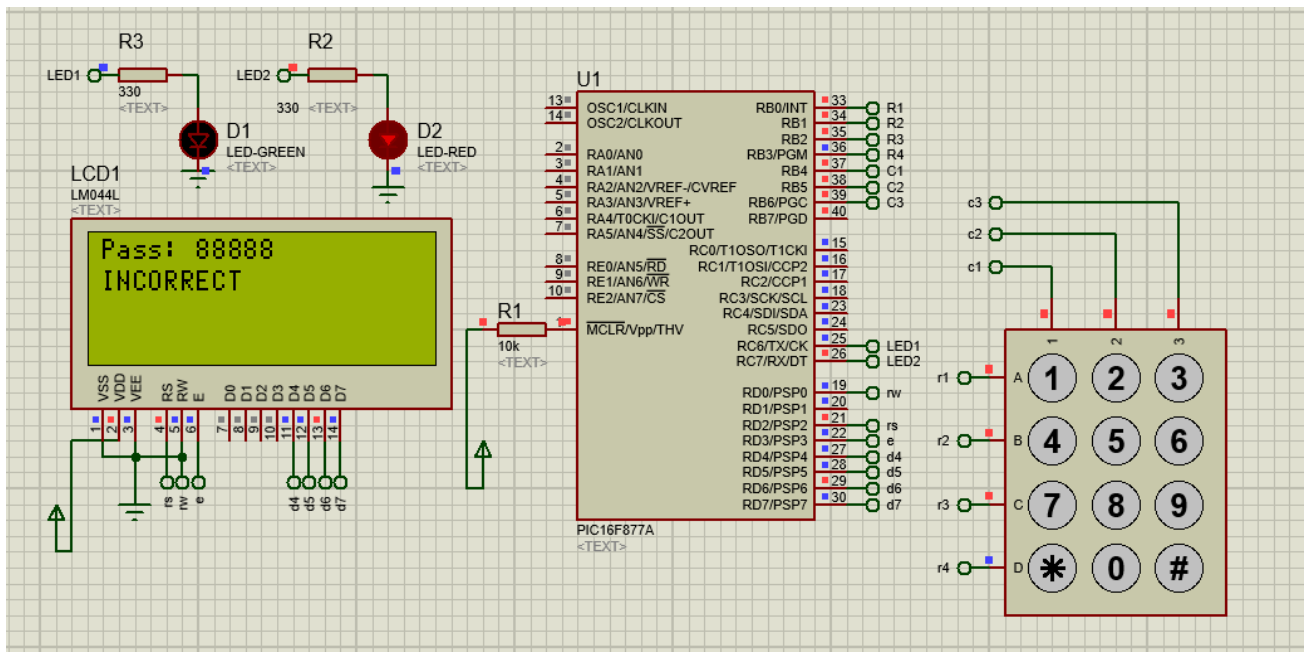


Image 2. Reaction for wrong password

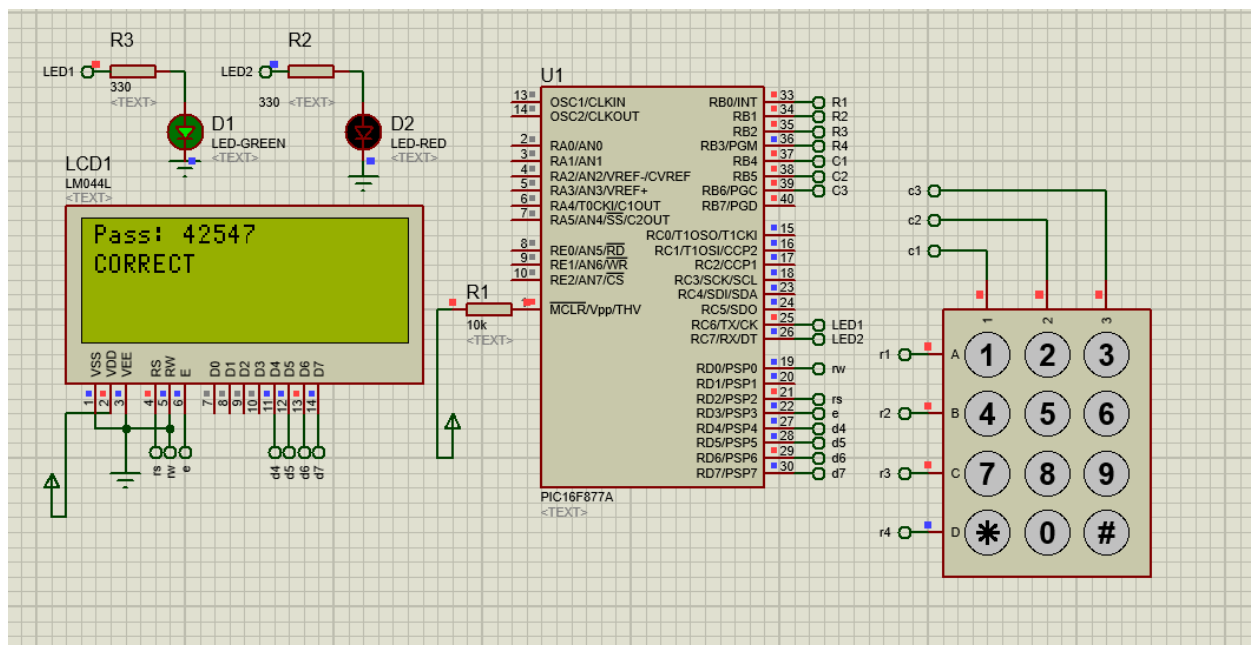


Image 3. Allowed access by provided right password "42547"

# Analog Digital Conversion

## Variant №3

Analog Value is 1.5V

Task is to simulate the program using the circuit which converts analog values into digital form.

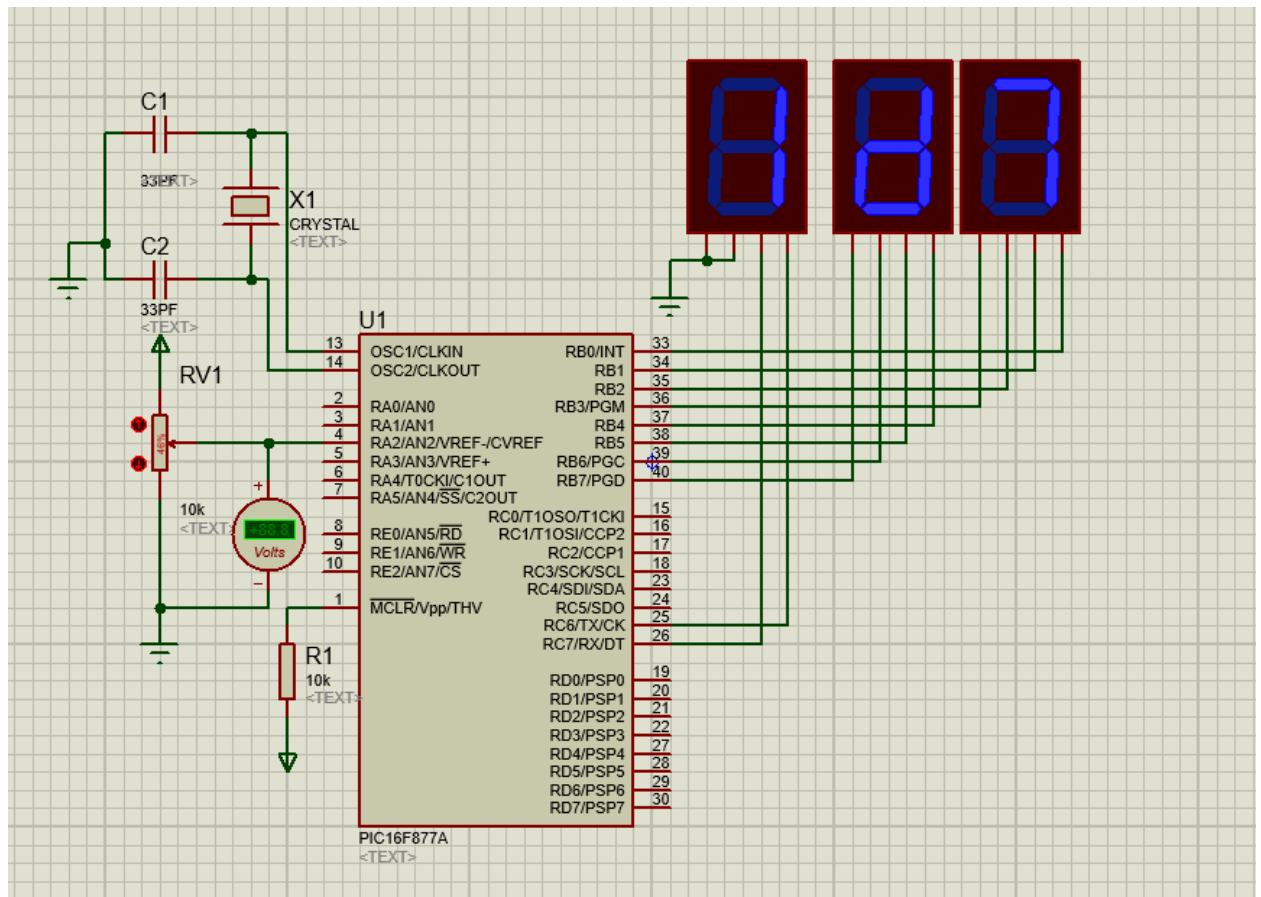


Image 4. A circuit with PIC16F877A

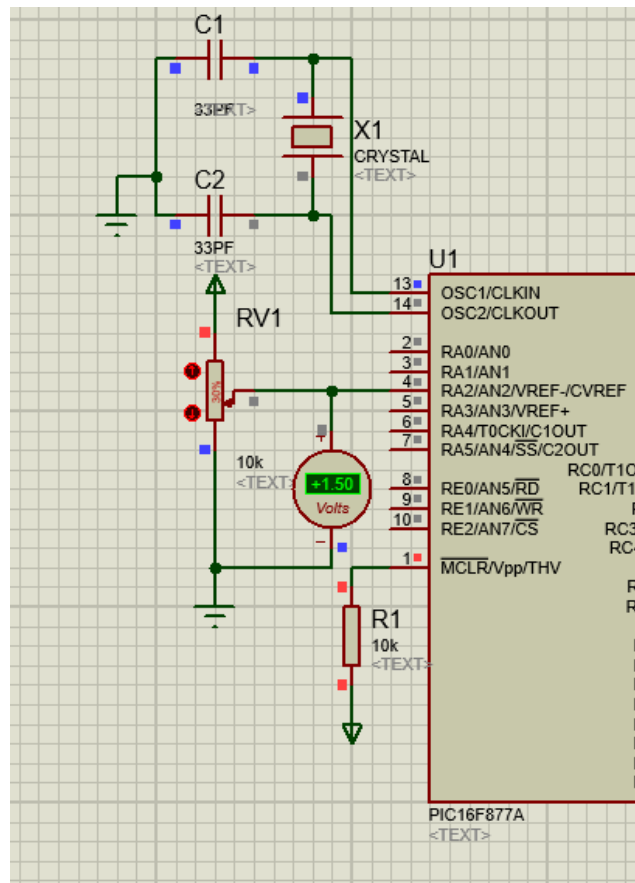


Image 5. Analog 1.5V is supplied to the circuit

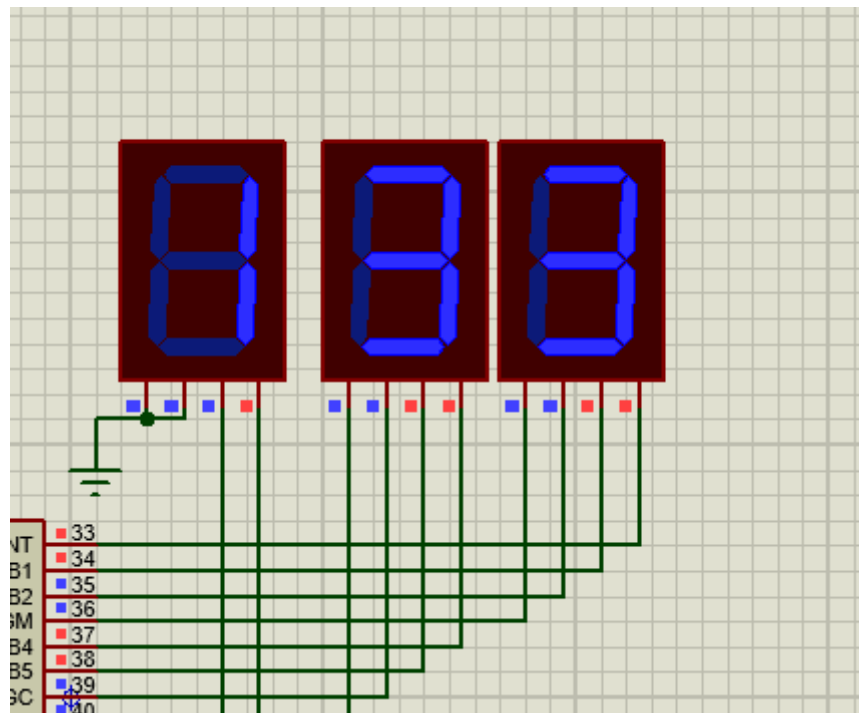


Image 6. Transformed analog signal to digital form

## *Appendix 1. ACD analog transformation main function code.*

```
#include <xc.h>
#include <pic16f877a.h>

#define _XTAL_FREQ 8000000

void ADC_Init()
{
    ADCON0 = 0b01000001; //ADC ON and Fosc/16 is selected
    ADCON1 = 0b11000000; // Internal reference voltage is selected
}

unsigned int ADC_Read(unsigned char channel)
{
    ADCON0 &= 0x11000101; //Clearing the Channel Selection Bits
    ADCON0 |= channel<<3; //Setting the required Bits
    __delay_ms(2); //Acquisition time to charge hold capacitor
    GO_nDONE = 1; //Initializes A/D Conversion
    while(GO_nDONE); //Wait for A/D Conversion to complete
    return ((ADRESH<<8)+ADRESL); //Returns Result
}

void main()
{
    int a;
    TRISA = 0xFF;           //Analog pins as Input
    TRISB = 0x00;           //Port B as Output
    TRISC = 0x00;           //Port C as Output
    ADC_Init();             //Initialize ADC
    do
    {
        a = ADC_Read(2);    //Read Analog Channel 0
        PORTB = a;          //Write bits to PORTB
        PORTC = a>>2;       //Write higher 2 bits to PORTC
    } while(1);
}
```