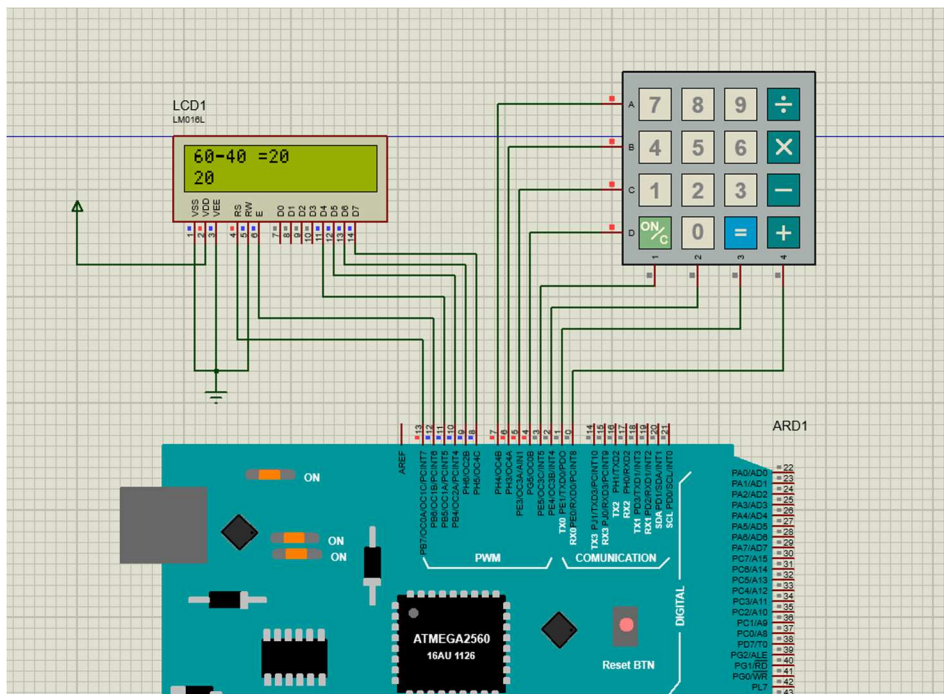
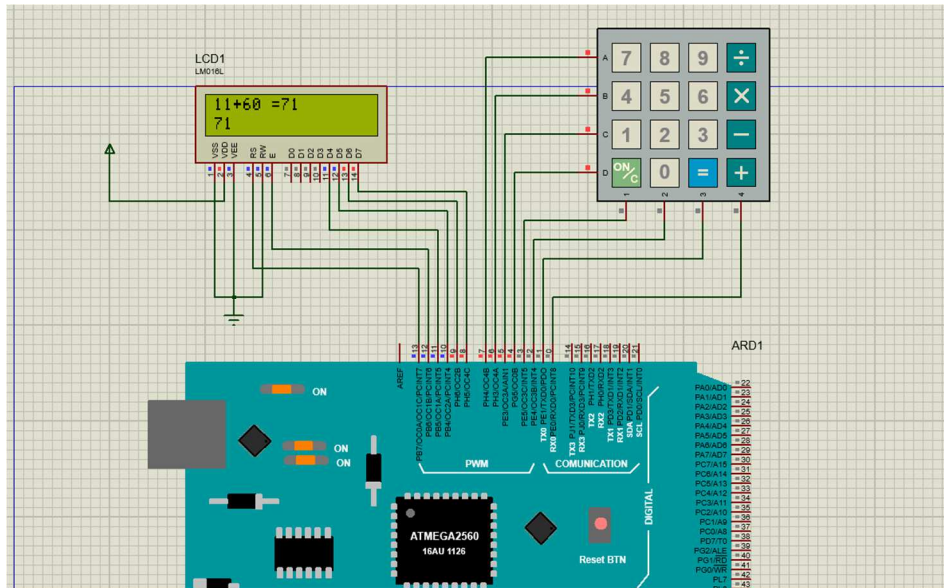
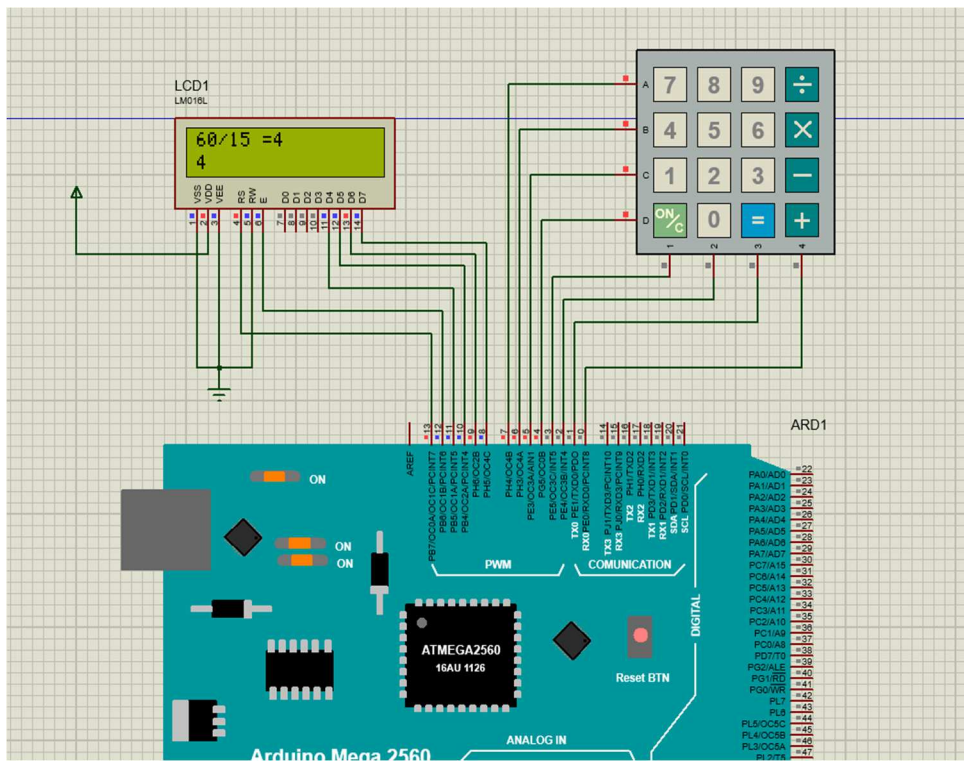
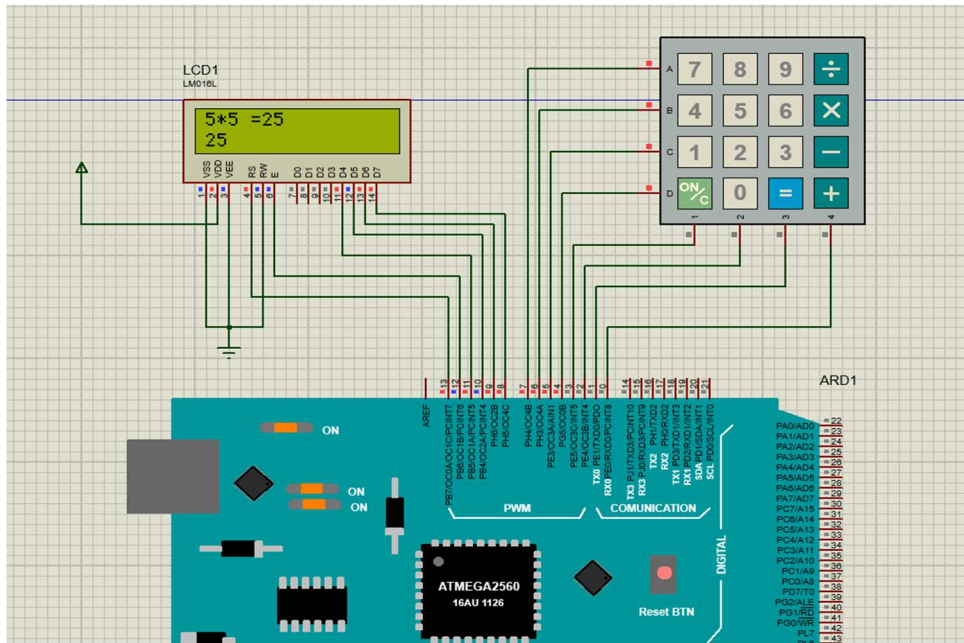


Mohammad kazemi 9731050

Roham zende del 9731088





```

sketch_oct4
#include <LiquidCrystal.h>
#include <Keypad.h>

// Define the Keypad
char keys[4][4] = {
  {'7','8','9','D'},
  {'4','5','6','C'},
  {'1','2','3','B'},
  {'*','0','#','A'}
};

byte rowPins[4] = { 7, 6, 5, 4 };
byte colPins[4] = { 3, 2, 1, 0 };

Keypad kpd = Keypad( makeKeymap(keys), rowPins, colPins, 4, 4 ); // Create the Keypad

const int rs = 13, en = 12, d4 = 11, d5 = 10, d6 = 9, d7 = 8;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);

long Num1, Num2, Number;
char key, action;
boolean result = false;

void setup() {
  lcd.begin(16, 2);
  lcd.setCursor(0, 1);
}

```

در ابتدا کلید های کی پد رو تو یک آرایه ی چهار در چهار تعریف میکنیم.

سپس بین های سطر وستون ها را مشخص کرده و یک شی کی پد با اطلاعات داده شده می سازیم.

یک شی ال سی دی می سازیم وبعد متغیر هایی رو برای استفاده تعریف می کنیم.

```

sketch_oct4

void loop() {

  key = kpd.getKey(); //storing pressed key value in a char

  if (key!=NO_KEY)
    DetectButtons();

  if (result==true)
    CalculateResult();

  DisplayResult();

}

```

در اینجا هم ابتدا کلیدی که گرفته می شود از کی پد در متغیر کی ذخیره می شود.

و سه تابع فراخوانی می کنیم که در ادامه توضیح میدیم.

sketch_oct4

```
void DetectButtons()
{
    lcd.clear();

    if(key >= '0' && key <= '9')
    {
        Serial.print ("Button ");
        Serial.println (key - '0');

        Number = (Number*10) + (key - '0');
    }

    if (key=='*') //If cancel Button is pressed
    {
        Serial.println ("Button Cancel");
        Number=Num1=Num2=0;
        action = 0x00;
        result=false;
    }

    if (key == '#')
    {
        Serial.println ("Button Equal");
        Num2=Number;
        result = true;
    }
}
```

اگر کلید های 0 تا 9 فشرده شود ان ها را در متغیر نامبر ثبت می کند(حال اگر نامبر خالی نباشد نامبر در ده ضرب می شود و با کلید فشرده شده جمع می شود.

اگر کلید * فشرده شود همه متغیر ها عددی 0 میشوند و نتیجه هم برابر false می شود.

اگر کلید # فشرده شود نتیجه چاپ میشود.

```

    if (key == 'A' || key == 'B' || key == 'C' || key == 'D') //Detecting Buttons on Column 4
    {
        Num1 = Number;
        Num2 = 0 ;
        Number =0;
        result = false;

        if (key == 'A')
        {Serial.println ("Addition"); action = '+';}
        if (key == 'B')
        {Serial.println ("Subtraction"); action = '-'; }
        if (key == 'C')
        {Serial.println ("Multiplication"); action = '*';}
        if (key == 'D')
        {Serial.println ("Devesion"); action = '/';}

        delay(100);
    }
}

```

در اینجا هم متناسب با کلید ی که انتخاب شده یکی از چهار عمل اصلی را انتخاب میکند.

```

sketch_oct4
}

void CalculateResult()
{
    if (action=='+')
        Number = Num1+Num2;

    if (action=='-')
        Number = Num1-Num2;

    if (action=='*')
        Number = Num1*Num2;

    if (action=='/')
        Number = Num1/Num2;
}

void DisplayResult()
{
    lcd.setCursor(0, 0); // set the cursor to column 0, line 0
    lcd.print(Num1); lcd.print(action); lcd.print(Num2);

    if (result==true)
    {lcd.print(" ="); lcd.print(Number);} //Display the result

    lcd.setCursor(0, 1); // set the cursor to column 0, line 1
    lcd.print(Number); //Display the result
}

```

در اینجا هم یکی از 4 عمل اصلی رو انجام میدهد و نتیجه ها رو درون ال سی دی نشان می دهد.

توابع مربوط به serial و keypad

`keypad(makekeymap(userkeymap), row[], col[], rows, cols)`

یک شیء keypad با پارامترهای داده شده می سازد که بتوانیم از توابع آن استفاده کنیم.

`char waitForKey()`

این تابع منتظری ماند تا شخص دکمه ای را فشار دهد.

`char getKey()`

این تابع دکمه ای که فشرده شده را برمی گرداند.
کلیدی

`KeyState getState()`

وضعیت کنونی هر کلیدی که بفراهم را برمی گرداند.

وضعیت ← IDLE, PRESSED, RELEASED, HOLD

`Serial.begin(speed)` ⇒ برای تنظیم کردن پورت سریال

`Serial.end()` ⇒ ارتباط سریال را قطع می کند و اجازه می دهد بین هایی که بر روی حالت ارتباط سریال قرار داشتند به صورت ورودی خروجی دیجیتال مورد استفاده قرار بگیرند.

`Serial.print()` val این تابع دیتا را به پورت سریال ارسال می کند.

`Serial.println()` این تابع همانند تابع print عمل می کند با این تفاوت که به خط بعد می رود (بعد از ارسال اطلاعات)

مشخصات و شکل LCD LM016L

شماره پایه	سمبول	نحوه اتصال پایه
1	Vss	اتصال به زمین
2	Vdd	اتصال به +5V
3	VEE یا Vo	تنظیم کنتراست LCD
4	RS	کنترل رجیستر
5	RW	انتخاب مد خواندن یا نوشتن
6	E	فعال سازی LCD
7 - 14	D0 - D7	گذرگاه 8 تایی اطلاعات و دستورالعمل
15، 16	کاتد LED، آند LED	آند و کاتد LED پس زمینه



مشخصات و شکل keypad 4*4

