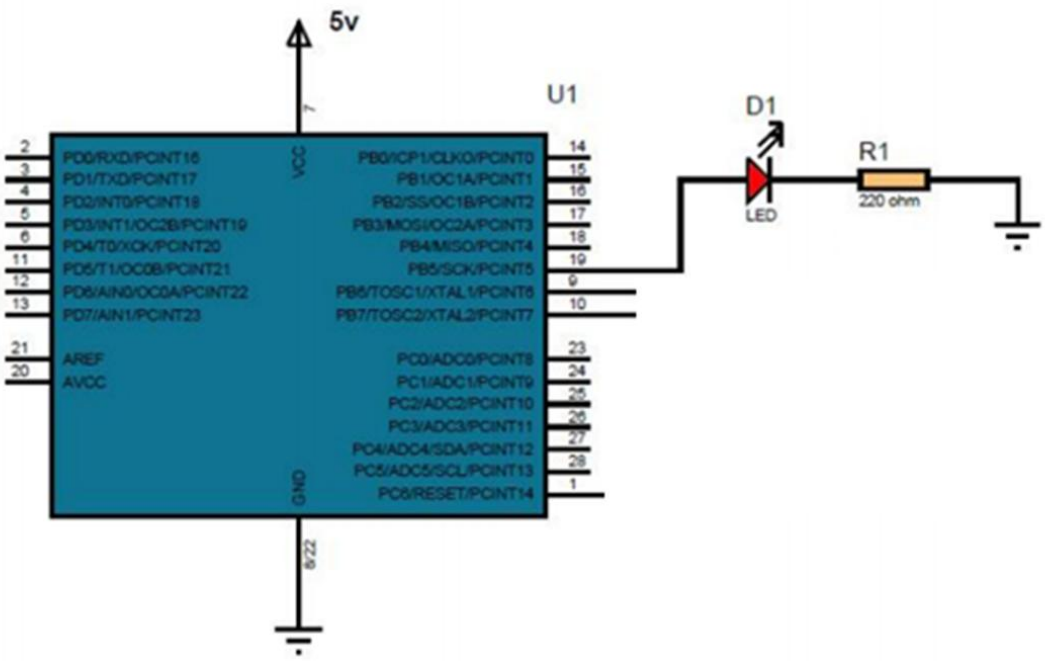
Project (01)

Blinking LED

```
void setup() {

  pinMode(13, OUTPUT);          <<---------- الرجل رقم 13 كمخرج
}

void loop() {                   <<---------- عمل حلقة غير منتهية

  digitalWrite(13, HIGH);       <<---------- إخراج تيار من الرجل رقم 13
                                              (سيضيء LED )
  delay(1000);                  <<---------- تأخير لمدة ثانية

  digitalWrite(13, LOW);        <<---------- جعل الرجل 13 تساوي 0 فولت
                                              (سينطفيء LED )
  delay(1000);                  <<---------- تأخير لمدة ثانية

}                               <<----------
```

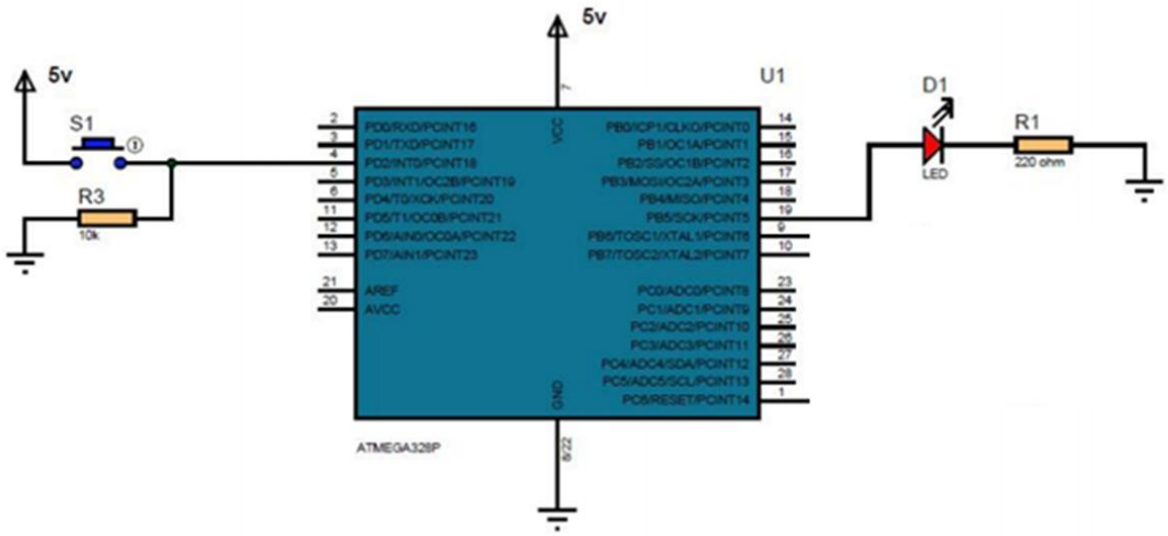


Project (02)

Blinking LED using button (IF conditions)

الأوامر

```
int LED=13;                                   <<---------- ترميز الرجل رقم 13 بـ كلمة LED
int BUTTON=2;                                 <<------ ترميز الرجل رقم 2 بـ كلمة BUTTON

void setup() {

  pinMode(LED, OUTPUT);                       <<---------- الرجل رقم 13 كمخرج
  pinMode(BUTTON, INPUT);                     <<---------- الرجل رقم 2 كمدخل
}

void loop() {                                 <<---------- عمل حلقة غير منتهية

 if (digitalRead(BUTTON)==HIGH) {             <<-- الجملة الشرطية في حالة الضغط على المفتاح
  digitalWrite(LED, HIGH);                    <<---------- سيضيء LED
  delay(1000);                                <<---------- تأخير لمدة ثانية
  digitalWrite(LED, LOW);                     <<---------- سينطفيء LED
  delay(1000);                                <<---------- تأخير لمدة ثانية
}                                             <<---------- نهاية الجميلة الشرطية

}                                             <<---------- العودة إلى كلمة loop
```



Project (03)

Blinking LED using button (if else condition)

الأوامر

```
int LED=13;
int BUTTON=2;

void setup() {

  pinMode(LED, OUTPUT);
  pinMode(BUTTON, INPUT);
}

void loop() {

 if (digitalRead(BUTTON)==HIGH) {
  digitalWrite(LED, HIGH); }
  else {
  digitalWrite(LED, LOW);  }

}
```

ترميز الرجل رقم 13 بـ كلمة LED <<----------

ترميز الرجل رقم 2 بـ كلمة BUTTON <<-----

الرجل رقم 13 كمخرج <<----------

الرجل رقم 2 كمدخل <<----------

عمل حلقة غير منتهية <<----------

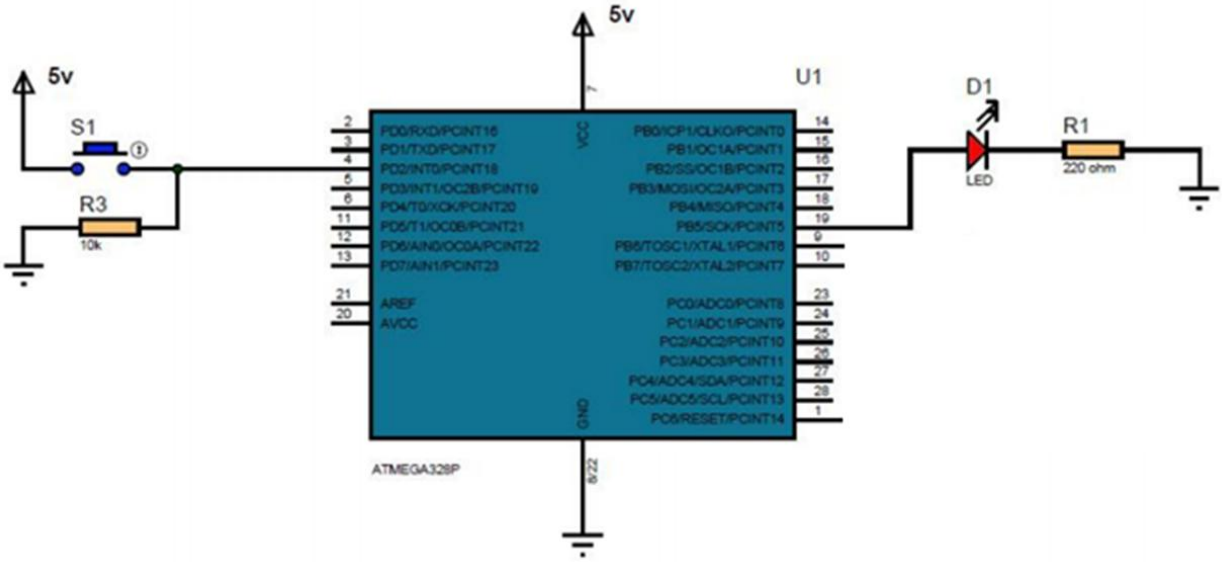
الجملة الشرطية في حالة الضغط على المفتاح <<--

سيضيء LED <<----------

ما عدا (إذا كان الشرط غير متحقق) <<------

سينطفيء LED <<----------

العودة إلى كلمة loop <<----------

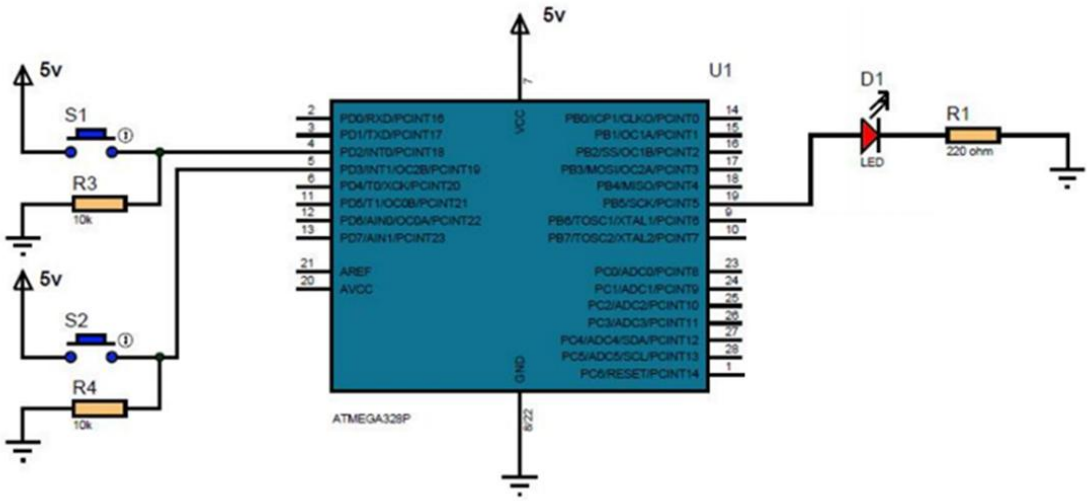


Project (04)

2 buttons (one turns on LED, the other turns it off)

الأوامر

```
int LED=13;                                   <<---------- ترميز الرجل 13 بـ كلمة LED
int BUTTON_ON=2;                              <<----- ترميز الرجل 2 بـ كلمة BUTTON_ON
int BUTTON_OFF=3;                             <<---- ترميز الرجل 3 بـ كلمة BUTTON_OFF

void setup() {

  pinMode(LED, OUTPUT);                       <<---------- الرجل رقم 13 كمخرج
  pinMode(BUTTON_ON, INPUT);                  <<---------- الرجل رقم 2 كمدخل
  pinMode(BUTTON_OFF, INPUT);  }              <<---------- الرجل رقم 3 كمدخل

void loop() {                                 <<---------- عمل حلقة غير منتهية

 if (digitalRead(BUTTON_ON)==HIGH) {          <<---------- الجملة الشرطية الأولى
  digitalWrite(LED, HIGH); }                  <<---------- سيضيء LED

 if (digitalRead(BUTTON_OFF)==HIGH) {         <<---------- الجملة الشرطية الثانية
  digitalWrite(LED, LOW); }                   <<---------- سينطفيء LED

}                                             <<----------
```



Project (05)

Alarm using button & buzzer

الأوامر

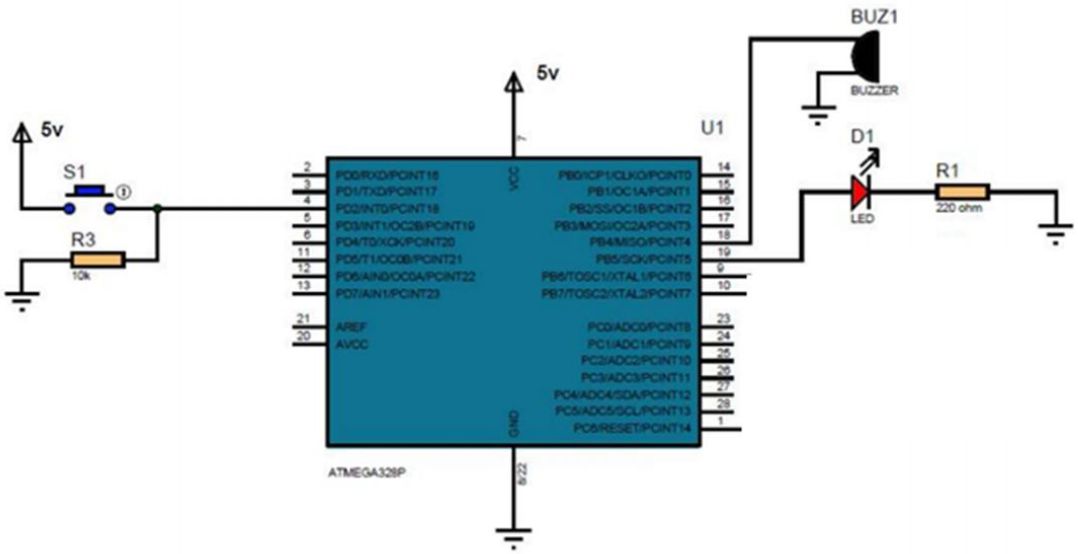
```
int LED =13;
int BUZZER =4;
int BUTTON =2;

void setup() {
  pinMode(LED, OUTPUT);
  pinMode(BUZZER, OUTPUT);
  pinMode(BUTTON, INPUT);
}

void loop() {

 if (digitalRead(BUTTON)==HIGH) {
  digitalWrite(LED, HIGH);
  digitalWrite(BUZZER, LOW);
  delay(500);
  digitalWrite(LED, LOW);
  digitalWrite(BUZZER, HIGH);
  delay(500); }
}
```

- ترميز الرجل رقم 13 بـ كلمة LED ----------<<
- ترميز الرجل رقم 4 بـ كلمة BUZZER------<<
- ترميز الرجل رقم 2 بـ كلمة BUTTON -----<<

- الرجل رقم 13 كمخرج ----------<<
- الرجل رقم 12 كمخرج ----------<<
- الرجل رقم 2 كمدخل ----------<<

- عمل حلقة غير منتهية ----------<<

- الجملة الشرطية في حالة الضغط على المفتاح <<-
- سيضيء LED ----------<<
- لن يعمل الطنان ----------<<
- سيصطر الطنان صوتاً ----------<<
- سينطفيء LED ----------<<
- سيصطر الطنان صوتاً ----------<<
- تأخير لمدة نصف ثانية ----------<<
- العودة إلى كلمة loop ----------<<



Project (06)

Traffic light project

الأوامر

```
int RED_LED = 2;
int YALLOW_LED =  8;
int GREEN_LED = 13;

void setup() {
  pinMode(RED_LED,OUTPUT);
  pinMode(YALLOW_LED,OUTPUT);
  pinMode(GREEN_LED,OUTPUT);
}
void loop() {

   digitalWrite(RED_LED, HIGH);
   digitalWrite(YALLOW_LED, LOW);
   digitalWrite(GREEN_LED, LOW);
   delay(2000);

   digitalWrite(RED_LED, LOW);
   digitalWrite(YALLOW_LED, HIGH);
   digitalWrite(GREEN_LED, LOW);
   delay(1000);
```

ترميز الرجل 2 بـ كلمة RED_LED <<---------

ترميز الرجل 8 بـ كلمة YALLOW_LED <<----

ترميز الرجل 13 بـ كلمة GREEN_LED <<-----

الرجل رقم 2 كمخرج <<----------

الرجل رقم 8 كمخرج <<----------

الرجل رقم 13 كمخرج <<----------

عمل حلقة غير منتهية <<----------

يضيء LED الأحمر <<----------

ينطفيء LED الأصفر <<----------

ينطفيء LED الأخضر <<----------

تأخير لمدة ثانيتين <<----------

ينطفيء LED الأحمر <<----------

يضيء LED الأصفر <<----------

ينطفيء LED الأخضر <<----------

تأخير لمدة ثانية <<----------

```
   digitalWrite(RED_LED, LOW);
   digitalWrite(YALLOW_LED, LOW);
   digitalWrite(GREEN_LED, HIGH);
   delay(2000);

   digitalWrite(RED_LED, LOW);
   digitalWrite(YALLOW_LED, HIGH);
   digitalWrite(GREEN_LED, LOW);
   delay(1000);

}
```

ينطفيء LED الأحمر <<----------

ينطفيء LED الأصفر <<----------

يضيء LED الأخضر <<----------

تأخير لمدة ثانيتين <<----------

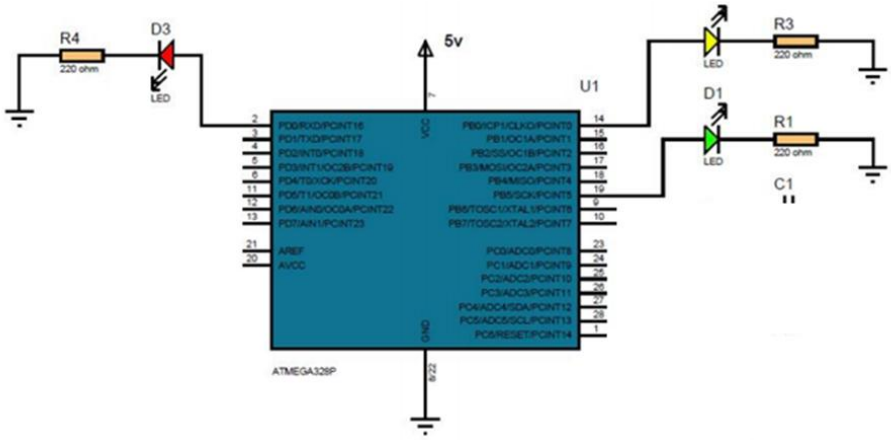
ينطفيء LED الأحمر <<----------

يضيء LED الأصفر <<----------

ينطفيء LED الأخضر <<----------

تأخير لمدة ثانية <<----------

<<----------

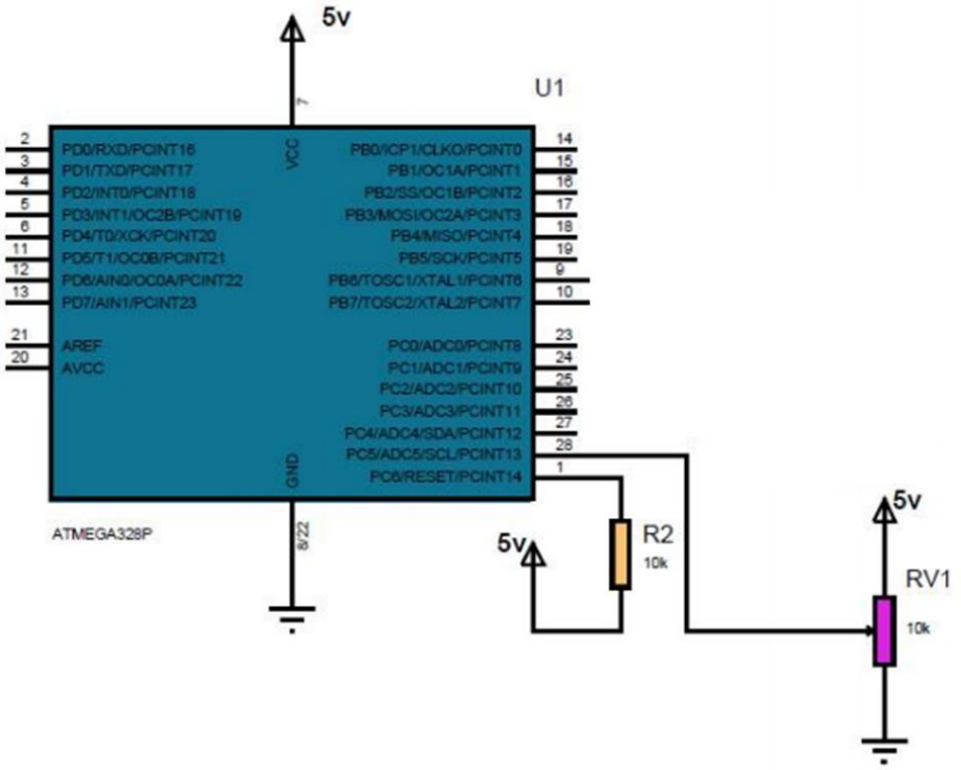


Project (07)

Reading Analog values from sensor & display it on serial monitor (variable resistance)

الأوامر

```
int sensorValue;                          <<------- sensorValue تعريف متغير باسم
int sensorPin=A5;                         <<------- sensorPin ترميز الرجل A5 بـ كلمة

void setup()
{
  Serial.begin(9600);                     <<-------- تفعيل شاشة المراقب التسلسلي ---
}

void loop () {                            <<---------- عمل حلقة غير منتهية
                                          قراءة القيمة التناظرية لقيمة المقاومة
  sensorValue=analogRead(sensorPin);      <<---------- المتغيرة وتخزينها في متغير
  Serial.println(sensorValue);            <<-- عرض المتغير على شاشة المراقب التسلسلي
  delay(500);                             <<---------- تأخير لمدة نصف ثانية

}                                         <<----------
```

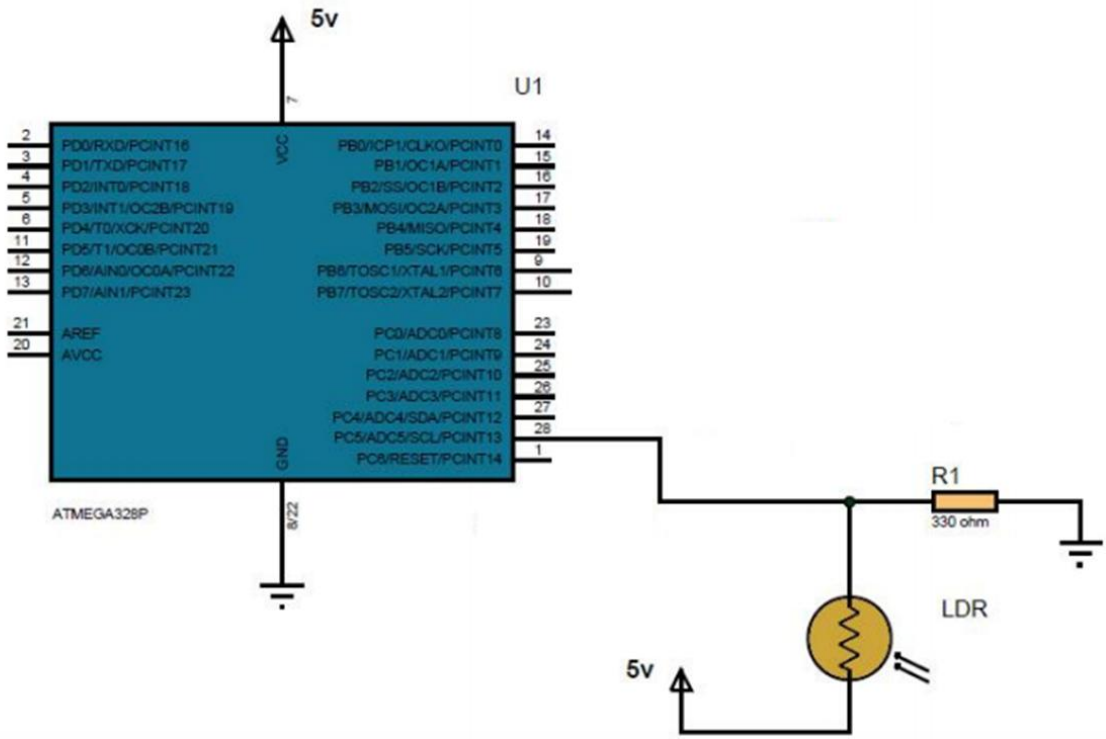


Project (08)

LDR application

الأوامر

```
int LDR;                                  <<---------- LDR تعريف متغير باسم
int sensorPin=A5;                         <<------- sensorPin ترميز الرجل A5 بـ كلمة

void setup()
{
  Serial.begin(9600);                     <<-------- تفعيل شاشة المراقب التسلسلي ---
}

void loop () {

                                          قراءة القيمة التناظرية لقيمة المقاومة
  LDR=analogRead(sensorPin);              الضوئية وتخزينها في متغير <<----------
  Serial.println(LDR);                    عرض المتغير على شاشة المراقب التسلسلي <<--
  delay(500);                             تأخير لمدة نصف ثانية <<----------

}                                         <<----------
```

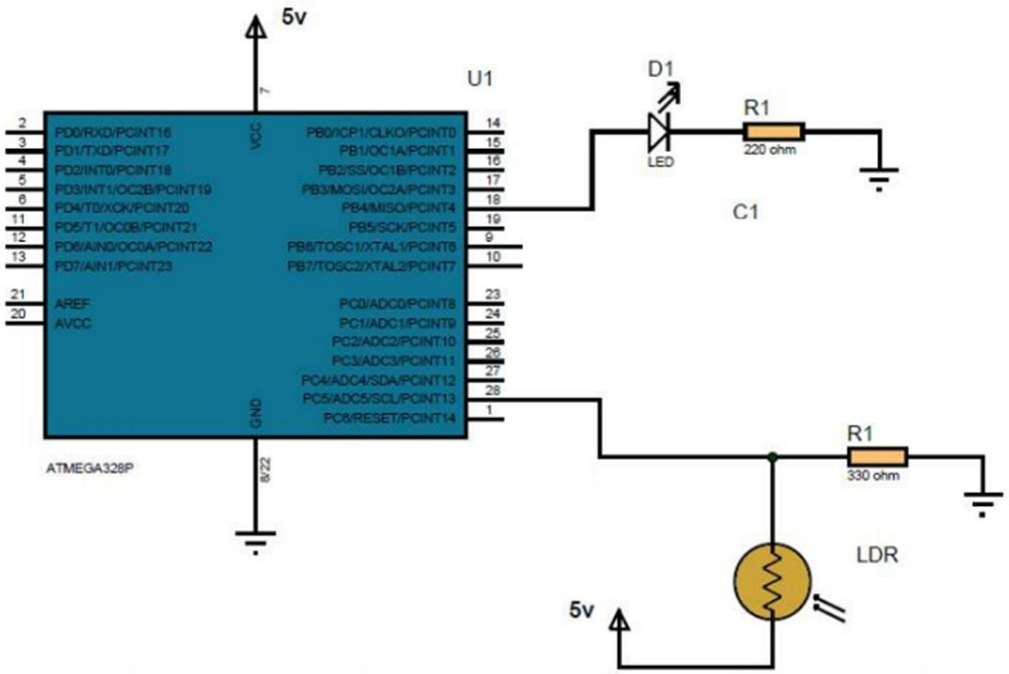


Project (09)

Automatic Street Lighting

الأوامر

```
int WHITE_LED = 12;
int LDR;                                  <<---------- LDR تعريف متغير باسم
int sensorPin=A5;                         <<------- sensorPin ترميز الرجل A5 بـ كلمة

void setup() {
  pinMode(WHITE_LED,OUTPUT);
  Serial.begin(9600);                     <<-------- تفعيل شاشة المراقب التسلسلي ---
}
void loop() {
  LDR=analogRead(sensorPin);
  Serial.println(LDR);
  delay(100);

  if (LDR<140) {                          <<-- 140 الجملة الشرطية إذا كانت القيمة أصغر من
    digitalWrite(WHITE_LED,HIGH); }       <<---------- تضيء الإنارة في حالة الإظلام
  else {
    digitalWrite(WHITE_LED,LOW); }        <<--------- وتنطفيء في حالة وجود الضوء

}
```



Project (10)

Servo motor application (project 1)

الأوامر

```
#include <Servo.h>
Servo myservo;

void setup()
{
  myservo.attach(3);
}

void loop()
{

 myservo.write(0);
 delay(2000);
 myservo.write(90);
 delay(2000);
 myservo.write(180);
 delay(2000);

}
```

- إضافة مكتبة Servo إلى البرنامج ---------<<
- ربط مكتبة Servo بكلمة myservo ------<<
- تحديد الرجل رقم 3 كمخرج لإشارة المحرك <<
- أمر أن يتحرك المحرك للزاوية 0 ----------<<
- أمر أن يتحرك المحرك للزاوية 90 ----------<<
- أمر أن يتحرك المحرك للزاوية 180 ---------<<

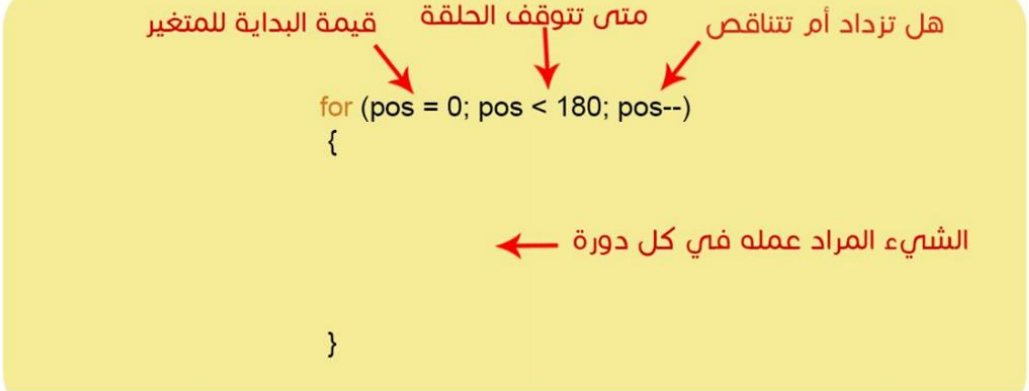
Project (11)

Servo motor application (project 2)

الأوامر

```
#include <Servo.h>                          <<--------- إضافة مكتبة Servo إلى البرنامج
Servo myservo;                              <<------ ربط مكتبة Servo بكلمة myservo
int pos = 0;                                <<---------- تعريف متغير باسم pos

void setup() {
  myservo.attach(3);                        << تحديد الرجل رقم 3 كمخرج للإشارة المحرك
 Serial.begin(9600);
}

void loop() {                               حلقة تتكرر 180 مرة
                                            يبدأ المتغير من 0 إلى 179 وفي كل مرة
  for (pos = 0; pos < 180; pos++)           <<---------- يزداد بواحد
  {
    myservo.write(pos);                     <<- تتحرك زاوية المحرك بنفس قيمة المتغير
    Serial.print("position: ");
    Serial.print(pos);                      <<---- كتابة المتغير على المراقب التسلسلي
    Serial.println(" degree");
    delay(15);
  }
```

```
                                            حلقة تتكرر 180 مرة يبدأ المتغير
 for (pos = 0; pos < 180; pos--)            من 179 إلى 0 وفي كل مرة يتناقص بواحد <<
 {
   myservo.write(pos);                      <<- تتحرك زاوية المحرك بنفس قيمة المتغير
   Serial.print("position: ");
   Serial.print(pos);                       <<---- كتابة المتغير على المراقب التسلسلي
   Serial.println(" degree");
   delay(15);
 }

}
```



Project (12)

LED Blinking using PWM

```
int LED = 11;                                   <<------- يتطلب اختيار رجل بها خاصية PWM
int fade = 0;                                   <<---------- تعريف متغير باسم fade

void setup()  {
}

void loop() {                                   حلقة تتكرر 255 مرة
                                                يبدأ المتغير من 0 إلى 255 وفي كل مرة
 for (fade = 0; fade <= 255; fade +=3) {        <<---------- يزداد بثلاثة
   analogWrite (LED, fade);                     <<- تعتمد قوة إضاءة LED على قيمة fade
   delay(20);
 }

                                                حلقة تتكرر 255 مرة
                                                يبدأ المتغير من 255 إلى 0 وفي كل مرة
 for (fade = 255; fade >= 0; fade −=3) {        <<---------- يتناقص بثلاثة
   analogWrite (LED, fade);                     <<- تعتمد قوة إضاءة LED على قيمة fade
   delay(20);
 }

}
```

Project (13)

Ultrasonic sensor application

```
#include "Ultrasonic.h"                         << إضافة مكتبة ULTRASONIC إلى البرنامج
                                                تحديد أرجل ULTRASONIC
Ultrasonic ultrasonic(8,9);                     <<---------- (Trig,Echo)
int cm;                                         <<---------- تعريف متغير باسم CMM
int BUZZER =13;

void setup() {
  pinMode(BUZZER , INPUT); }
void loop() {
   cm= ultrasonic.Ranging(cm);                  <<- قراءة قيمة المسافة وتخزينها في المتغير CMM
                                                الجملة الشرطية في حالة إقتراب جسم من
   if (cm>200) {                                <<---------- مجسة ULTRASONIC
   digitalWrite (BUZZER,HIGH);
   delay(100);
   digitalWrite (BUZZER,LOW);
   delay(100);
   }
   else  {
   digitalWrite (BUZZER,LOW);
   }
}
```

Project (14)

Turn on LED using Keypad 4*4

```
#include <Keypad.h>                                   <<------- إضافة مكتبة Keypad إلى البرنامج

const byte ROWS = 4;                                  <<------------------------------ أربعة صفوف
const byte COLS = 4;                                  <<------------------------------ أربعة أعمدة

char hexaKeys[ROWS][COLS] = {                         <<--------- تعريف رموز المفاتيح في اللوحة
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};
byte rowPins[ROWS] = {3, 4, 5, 6};                    << أرقام الأرجل التي يتم توصيلها مع الصفوف
byte colPins[COLS] = {7, 8, 9, 10};                   <<  أرقام الأرجل التي يتم توصيلها مع الأعمدة

Keypad customKeypad = Keypad( makeKeymap(hexaKeys),
 rowPins, colPins, ROWS, COLS);

int LED=13;                                           <<----------------- ترميز مجموعة من الأرجل
char customKey;                                       <<---------- تعريف متغير باسم customKey
void setup()  {
  pinMode(LED,OUTPUT);
  Serial.begin(9600);
}

void loop()
{
                                                      قراءة رمز المفتاح الذي تم ضغطه وتخزين
  customKey = customKeypad.getKey();                  <<---------------------------- الرمز في متغير
                                                      الجملة الشرطة المرتبطة بعرض الرقم في
  if (customKey != NO_KEY)                            <<---------------- حالة الضغط على أي مفتاح
  {
    Serial.println(customKey);                        <<---- كتابة المتغير على المراقب التسلسلي

    switch (customKey)
    {
      case '1':                                       << الجملة الشرطية في حالة الضغط على الرقم 1
       digitalWrite(LED,HIGH);                        <<------------------------------ تشغيل LED
       break;

      case '2':                                       << الجملة الشرطية في حالة الضغط على الرقم 2
       digitalWrite(LED,LOW);                         <<------------------------------- إطفاء LED
       break;

      default: ;
    }
  }
}
```

Project (15)

Display Hello on LCD (16*2)

```
#include<LiquidCrystal.h>                    <<---------- إضافة مكتبة LCD إلى البرنامج
LiquidCrystal lcd(8, 9, 4, 5, 6, 7);        <<---- تهيئة المكتبة مع المنافذ المستخدمة
                                                    (RS,E,D4,D5,D6,D7)
void setup() {
  lcd.begin(16,2);                          <<-- تهيئة شاشة LCD وتحديد ابعادها (الاعمدة والصفوف)
  lcd.print("Hello");                       <<--------- طباعة النص على شاشة العرض
 }

void loop() {

  lcd.display();                            <<---------- عرض النص على الشاشة
  delay(1000);                              <<---------- تأخير لمدة ثانية
  lcd.noDisplay();                          <<---------- إخفاء النص من على الشاشة
  delay(1000);                              <<---------- تأخير لمدة ثانية

 }
```