

Unidad 6. SISTEMAS DOMÓTICOS BASADOS EN ELECTRÓNICA. ARDUINO

ACTIVIDADES DE COMPROBACIÓN

6.1. d)

6.2. d)

6.3. a)

6.4. c)

6.5. b)

6.6. a)

6.7. a)

6.8. d)

6.9. a)

6.10. b)

ACTIVIDADES DE APLICACIÓN

6.11.

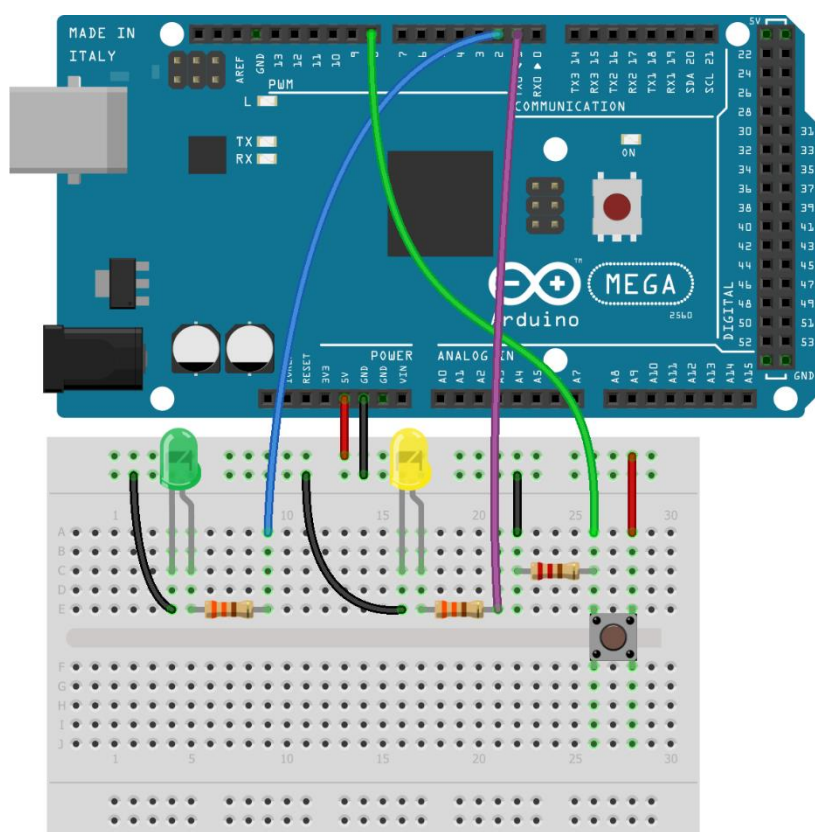
El código del programa Arduino Mega es:

```
int LED1 = 2;
int LED2 = 3;
int boton = 8;

void setup()
{
  pinMode( LED1, OUTPUT) ;
  pinMode( LED2, OUTPUT) ;
  pinMode( boton, INPUT) ;
}

void loop()
{
  int valor = digitalRead(boton) ;
  digitalWrite( LED1, valor) ;
  digitalWrite( LED2, valor) ;
}
```

El montaje es:



6.12.

```
int boton = 7;

void setup()
{
  Serial.begin (9600) ;
  pinMode( boton, INPUT) ;
}

void loop()
{
  int valor = digitalRead(boton) ;

  if (valor == 1)
  {
    Serial.println (123+87);
  }
  else Serial.println(0);
  delay(500);
}
```

Se ha configurado la comunicación a una velocidad de 9600 bits por segundo.

Se ha optado por incluir una secuencia de control (*if*) que evalúa cuándo se ha pulsado. En este caso muestra el valor de la suma (210) y en caso contrario, es decir, cuando no se acciona el pulsador, que muestre el valor 0. Además, se ha puesto un retardo (*delay*).

6.13.

```
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  int lectura = analogRead (A3);
  Serial.println (lectura);
  delay (200);
}
```

6.14.

```
int LED1 = 2;
int LED2 = 3;
int boton = 8;
int zumbador = 9;
void setup()
{
  pinMode( LED1, OUTPUT) ;
  pinMode( LED2, OUTPUT) ;
  pinMode( zumbador, OUTPUT) ;
  pinMode( boton, INPUT) ;
}

void loop()
{
  int valor = digitalRead(boton) ;
  if (valor==1)
  {
    digitalWrite( LED1, valor) ;
    digitalWrite( LED2, valor) ;
    beep(250);
  }
}

void beep(unsigned char pausa)
{
  analogWrite(zumbador, 20);
  delay(pausa);
  analogWrite(zumbador, 0);
  delay(pausa);
}
```

6.15.

```
int LED = 2;
int boton = 8;
int operacion;

void setup()
{
  pinMode( LED, OUTPUT);
  pinMode( boton, INPUT);
  operacion = 0;
}

void loop()
{
  int valor = digitalRead(boton) ;
  delay (100);

  if (valor==1) // se pulsa el boton
  {
    operacion=operacion+1;
    if (operacion==3) operacion=0;

    if (operacion==0) analogWrite(LED, 0); // Se apaga
    if (operacion==1) analogWrite(LED, 128); // Se pone al 50%
    if (operacion==2) analogWrite(LED, 255); // Se pone al 100%
  }
}
```

En la variable *operación* se guarda el número de veces que se ha pulsado. Si se pulsa una vez, se pone la intensidad a un 50 % (valor de 128); si se pulsa dos veces se pone al 100 % (valor de 255) y, si se pulsa tres veces, se pone el contador a cero y se apaga. Esta variable se crea y se inicializa con el valor cero.

Para incrementar el número de veces que se pulsa (*operación*) se utiliza el código:

```
operación=operación+1;
```

Aunque es más común que se indique de la siguiente manera y significa lo mismo:

```
operación++;
```

Se ha decidido poner un retardo, indicado como:

```
delay(100);
```

6.16.

```
#include <Wire.h>
#include <LCD.h>
#include <LiquidCrystal_I2C.h>

// Esta dirección se debe averiguar.
#define I2C_ADDR 0x20

LiquidCrystal_I2C lcd(I2C_ADDR,2, 1, 0, 4, 5, 6, 7);

void setup()
{
  lcd.begin (16,2);
  lcd.setBacklightPin(3,POSITIVE);
  lcd.setBacklight(HIGH);

  lcd.home ();
  lcd.print("Nombre");
  lcd.setCursor ( 0, 1 );
  lcd.print("Apellidos");
}
void loop()
{
}
```

Se considera que la pantalla LCD está en la posición 0x20.

Con la instrucción:

 lcd.home ();

se sitúa el cursor en la primera posición de la primera línea, escribiendo el nombre (aquí debe poner cada uno su propio nombre).

Posteriormente, se sitúa el cursor en la primera posición de la segunda línea (hay que tener en cuenta que empieza a contar desde 0), escribiendo el apellido (que también hay que sustituir por el de cada uno).

6.17.

```

#include <Wire.h>
#include <LCD.h>
#include <LiquidCrystal_I2C.h>

// Esta dirección se debe averiguar.
#define I2C_ADDR 0x20

LiquidCrystal_I2C lcd(I2C_ADDR,2, 1, 0, 4, 5, 6, 7);

int LED = 4;
int boton = 5;

void setup()
{
    pinMode( LED, OUTPUT) ;
    pinMode( boton, INPUT) ;

    lcd.begin (16,2);
    lcd.setBacklightPin(3,POSITIVE);
    lcd.setBacklight(HIGH);
}

void loop()
{
    int valor = digitalRead(boton) ;
    delay (500);

    lcd.clear();
    lcd.home ();
    lcd.print("Situacion LED");
    if (valor==1) // se pulsa el boton
    {
        digitalWrite( LED, 1) ;
        lcd.setCursor ( 0, 1 );
        lcd.print("Encendido");
    }
    else // en caso contrario
    {
        digitalWrite( LED, 0) ;
        lcd.setCursor ( 0, 1 );
        lcd.print("Apagado");
    }
}

```

En la función *loop* se lee el estado del pulsador denominado *botón*. Se hace una pausa entre lecturas (no hace falta poner esta línea de código).

Con el código siguiente, se borra lo que hay escrito y el cursor se sitúa en la posición *home* (primera línea y primer carácter) y se escribe “Situacion LED”.

```

    lcd.clear();
    lcd.home ();
    lcd.print("Situacion LED");

```

Si se acciona el pulsador, se enciende el led y se coloca al principio de la segunda línea el mensaje de encendido.

En caso contrario (*else*), cuando no se acciona el pulsador, se apaga el led y se coloca el mensaje de apagado.

6.18.

```
// Para la pantalla LCD
#include <Wire.h>
#include <LCD.h>
#include <LiquidCrystal_I2C.h>

// Para el sensor
#include "DHT.h"
#define DHTpin 5 // Puerto 5
#define DHTtype DHT22 // o DHT11
DHT dht(DHTpin, DHTtype);

int led_t = 8;
int led_h = 9;

#define I2C_ADDR 0x20 // Direccion de la pantalla LCD
LiquidCrystal_I2C lcd(I2C_ADDR,2, 1, 0, 4, 5, 6, 7); // Configuracion LCD

void setup()
{
    dht.begin(); //Se inicia el sensor

    // Se inicializa la pantalla LCD
    lcd.begin (16,2);
    lcd.setBacklightPin(3,POSITIVE);
    lcd.setBacklight(HIGH);

    pinMode(led_t, OUTPUT);
    pinMode(led_h, OUTPUT);
}

void loop()
{
    // Se definen las variables de humedad y temperatura. Son de tipo decimal (float)
    float h = dht.readHumidity();
    float t = dht.readTemperature();

    lcd.clear();
    lcd.home();
    lcd.print("Humedad ");
    lcd.print(h);
    lcd.setCursor (0,1);
    lcd.print("Temp. ");
    lcd.print(t);

    delay(1000); // Pausa de 1 segundo

    if (t>25.0) // Temperatura mayor de 25 grados
    {
```

```
    digitalWrite(led_t,1); // Enciende el led
  }
  else
  {
    digitalWrite(led_t,0); // Apaga el led
  }

  if (h>60.0) // Humedad mayor de 60
  {
    digitalWrite(led_h,1); // Enciende el led
  }
  else
  {
    digitalWrite(led_h,0); // Apaga el led
  }
}
```