

Curs d'Arduino CTM16-17

Mòdul 3

SENSORS I ACTUADORS AMB ARDUINO

1. *Sensor de llum o LDR (Light Dependent Resistor)*
2. *Theremin òptic amb una LDR*
3. *Sensor de temperatura i humitat DHT*
4. *Visualització de temperatura i humitat en una pantalla LCD 16x2*
5. *Connectar un display LCD a Arduino mitjançant un controlador I²C*

SENSORS I ACTUADORS AMB ARDUINO

1. Sensor de llum o LDR (Light Dependent Resistor)

Principi:

Les fotocèl·lules són sensors que permeten detectar la llum. Aquestes poden ser les resistències dependents de la llum (LDR) i els fotoresistors.



Una LDR és un resistor que canvia el seu valor de resistència en funció de la quantitat de llum que incideix sobre la cara ondulada.

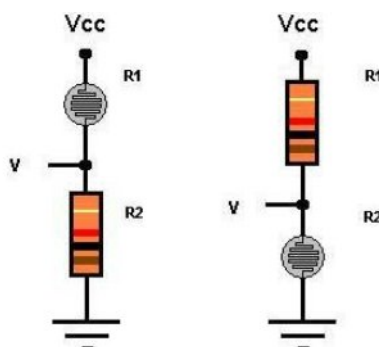
Cal tenir en compte que són components de molt baix cost, fàcil d'aconseguir en moltes mides i especificacions, però són molt inexactes. Cada LDR actuarà una mica diferent a una altra, fins i tot si són del mateix lot. Les variacions poden ser molt grans i per això no es aconsellable utilitzar-les en circuits que es requereix una certa precisió.

Mesura del voltatge obtingut en funció de la llum que incideix:

Un divisor de tensió està format per dues resistències en sèrie connectades entre una font de tensió i massa. Si R1 està connectada a l'alimentació (Vcc) i R2 es connecta a massa, la tensió en la unió entre les dues resistències és:

$$V = V_{CC} \cdot \frac{R_2}{R_1 + R_2}$$

Si R1 és la LDR, el voltatge V augmentarà amb l'augment d'intensitat de la llum. En canvi, si R2 és la LDR, la el voltatge V disminuirà en augmentar la intensitat de la llum.



Objectiu: Encendre/apagar gradualment un LED en funció de la llum captada per la LDR.

Components:

1	Arduino UNO o similar
1	protoboard
1	LED
1	$R = 330\ \Omega$ (o també $R = 220\ \Omega$)
1	LDR
1	$R = 10\ k\Omega$
	cables

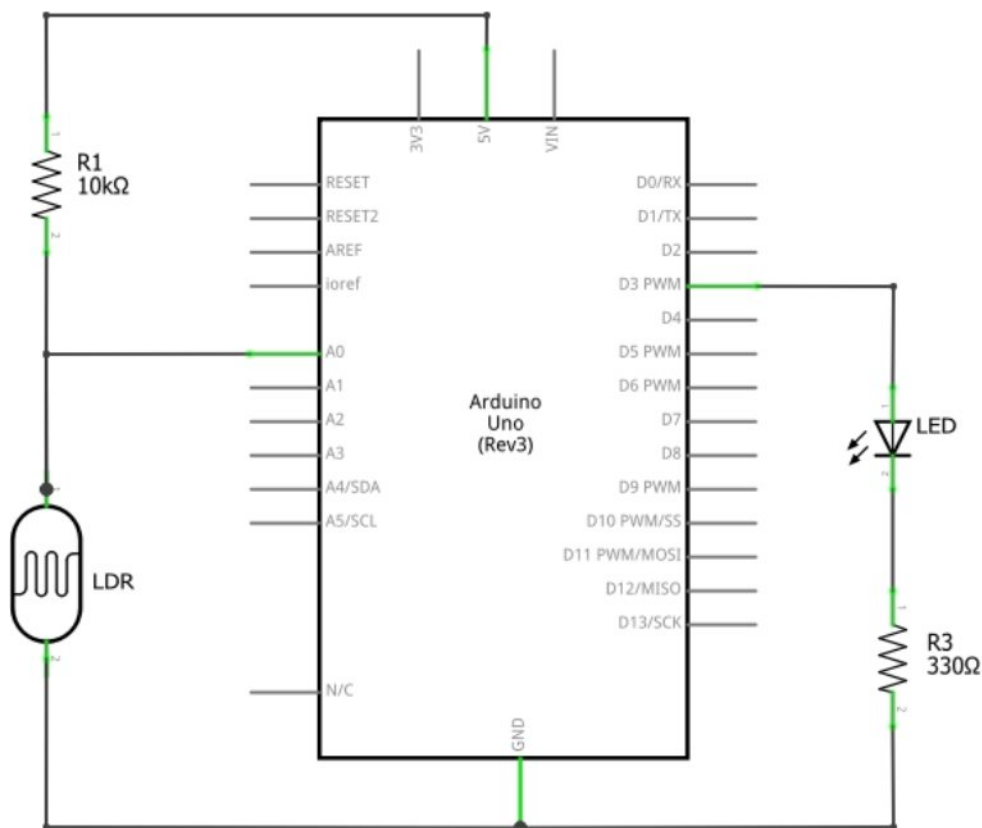
Circuit electrònic:

El LED el connectem a una sortida analògica PWM ja que regularem el seu voltatge d'alimentació.

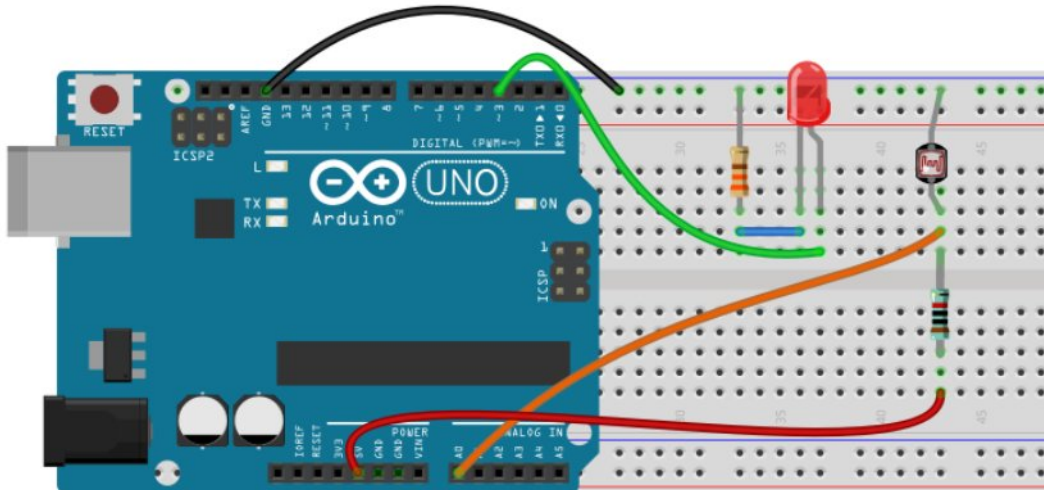
En canvi, la LDR la connectem en sèrie amb una resistència fixa de $10\ k\Omega$ (no és necessari que sigui aquest valor) entre 5V i 0V tenint en compte dues possibilitats:

- Si connectem la LDR a 0V, s'encendrà el LED al disminuir la intensitat lumínica incident sobre la LDR
- Si connectem la LDR a 5V, s'encendrà el LED a l'augmentar la intensitat lumínica incident sobre la LDR

Circuit connectant la LDR a 0V i la R_1 a 5V:



Muntatge: Connectant la LDR a 0V i la R₁ a 5V:



Codi:

```

/* IL·LUMINACIÓ D'UN LED SEGONS VALOR DE LDR

Encendre/apagar gradualment un LED en funció de la llum captada per la LDR
-----
*/

const int LED = 3;
int valorLDR = 0; // Valor llegit pel LDR.
int pwmLED = 0; // Valor de sortida PWM amb rang de 0 a 255 per il·luminació del LED

void setup()
{
    pinMode(LED, OUTPUT);
    Serial.begin(9600); // Inicialitza la comunicació amb el port sèrie a 9600 bauds
}

void loop()
{
    valorLDR = analogRead(A0); // Llegeix el valor analògic de la LDR

    pwmLED = map(valorLDR, 0, 1023, 0, 255); // Converteix el rang 0-1023 (valors reajustables) del LDR al rang
    // 0-255 per la sortida PWM del LED

    analogWrite(LED, pwmLED); // Il·lumina el LED amb el valor PWM de la sortida analògica

    Serial.print("LDR (0-1023) = ");
    Serial.print(valorLDR);
    Serial.print("\t Sortida PWM (0-255) = ");
    Serial.println(pwmLED);

    delay(20);
}

```

2. Theremin òptic amb una LDR

Principi:

Un bronzidor o buzzer en anglès, és un transductor electroacústic que produeix un so continu o intermitent d'un mateix to.

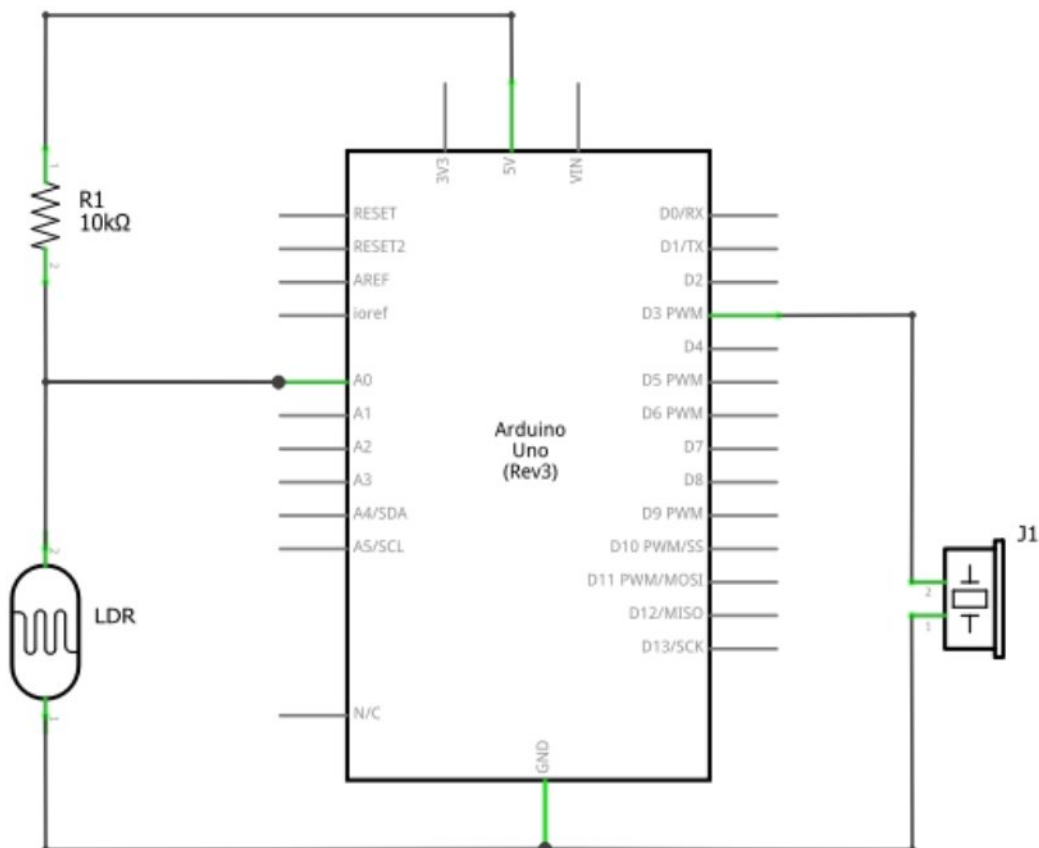
La seva construcció consta de dos elements, un electroimant i una làmina metàl·lica d'acer. Quan s'alimenta, el corrent passa per la bobina de l'electroimant i produeix un camp magnètic variable que fa vibrar la làmina d'acer sobre l'armadura produint un so.

Objectiu: Obtenir diferents tons de so d'un buzzer piezoelèctric en funció de la llum captada per la LDR.

Components:

1	Arduino UNO o similar
1	protoboard
1	LDR
1	R = 10 kΩ
1	buzzer piezoelèctric
	cables

Circuit electrònic:

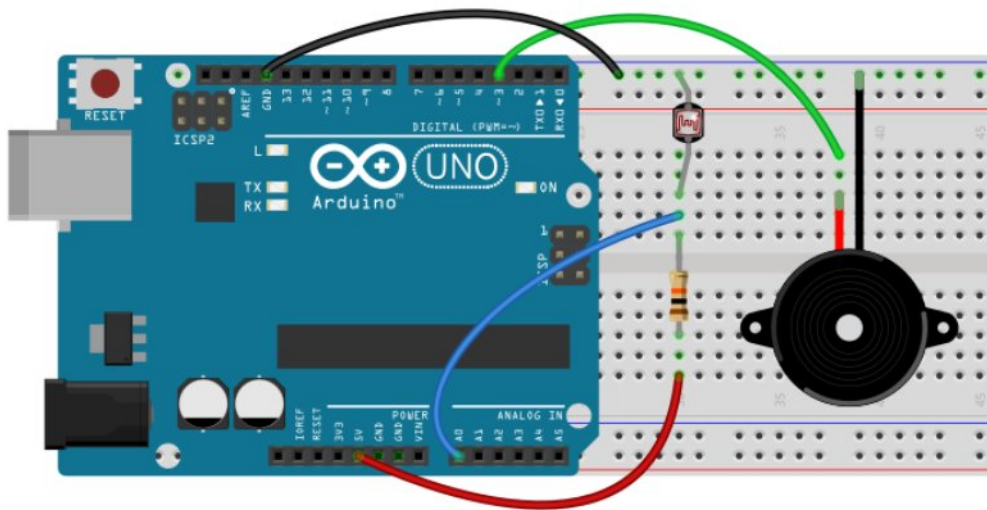


Es tracta d'un circuit similar al de la pràctica anterior, però substituïm el LED i la R_1 per un buzzer.



Connectem el born positiu del buzzer a un pin de sortida de l'Arduino i l'altre born a massa.

Muntatge:



Codi:

```
/* THEREMIN AMB BUZZER I LDR
```

Obtenir diferents tons de so d'un buzzer en funció de la llum captada per la LDR.

```
*/
```

```
const int Buzzer = 3;
```

```
int valorLDR = 0;
```

```
int freqBuzzer = 0;
```

```
// Valor llegit pel LDR.
```

```
// Variable que magatzema el valor de la freqüència del to del buzzer
```

```
void setup()
```

```
{
```

```
  pinMode (Buzzer , OUTPUT);
```

```
  Serial.begin (9600);
```

```
// Inicialitza la comunicació amb el port sèrie a 9600 bauds
```

```
}
```

```
void loop()
```

```
{
```

```
  valorLDR = analogRead(A0);
```

```
// Llegeix el valor analògic de la LDR
```

```
/* Converteix el rang 0-1023 (valors reajustables) del LDR a un rang d'una freqüència entre 20Hz i 5kHz pels tons del buzzer */
```

```

freqBuzzer = map (valorLDR, 0, 1023, 20, 5000); // Es pot experimentar amb aquests valors 20-5000 i canviar-los

tone (Buzzer,freqBuzzer); // El buzzer sona amb el to de la freqüència guardada

Serial.print("LDR (0-1023) = "); // Monitoritza els valors digitals del LDR i LED pel monitor sèrie
Serial.print(valorLDR);
Serial.print("\t To del so del buzzer = ");
Serial.println(freqBuzzer);

delay (20);
}

```

Referències:

- [tone](#)

3. Sensor de temperatura i humitat DHT

Principi:

Els sensors de la família DHT ens proporcionen lectures digitals de temperatura i humitat simultàniament. Els sensors DHT11 i DHT22 disposen d'un processador intern que realitza el procés de mesurament i fan servir el seu propi sistema de comunicació bidireccional mitjançant un únic conductor, emprant senyals temporitzades.

Els models DHT11 i el DHT22 tenen característiques tècniques similars, però el DHT22 és un model superior.



DHT11	DHT22
• Mesura de temperatura entre 0 a 50 °C, amb una precisió de 2 °C • Mesura d'humitat entre 20 i 80%, amb precisió del 5%. • Freqüència de mostreig d'1 mostres per segon • PVP = 2€ aprox.	• Mesura de temperatura entre -40 a 125 °C, amb una precisió de 0.5 °C • Mesura d'humitat entre 0 i 100%, amb precisió del 2-5%. • Freqüència de mostreig de 2 mostres per segon • PVP = 5€ aprox.

Objectiu: Mesurar temperatura i humitat mitjançant el sensor DHT11 i visualitzar els valors pel monitor sèrie.

Components:

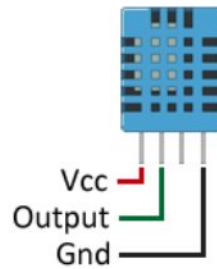
1	Arduino UNO o similar
1	protoboard

1 DHT11 cables

Circuit electrònic:

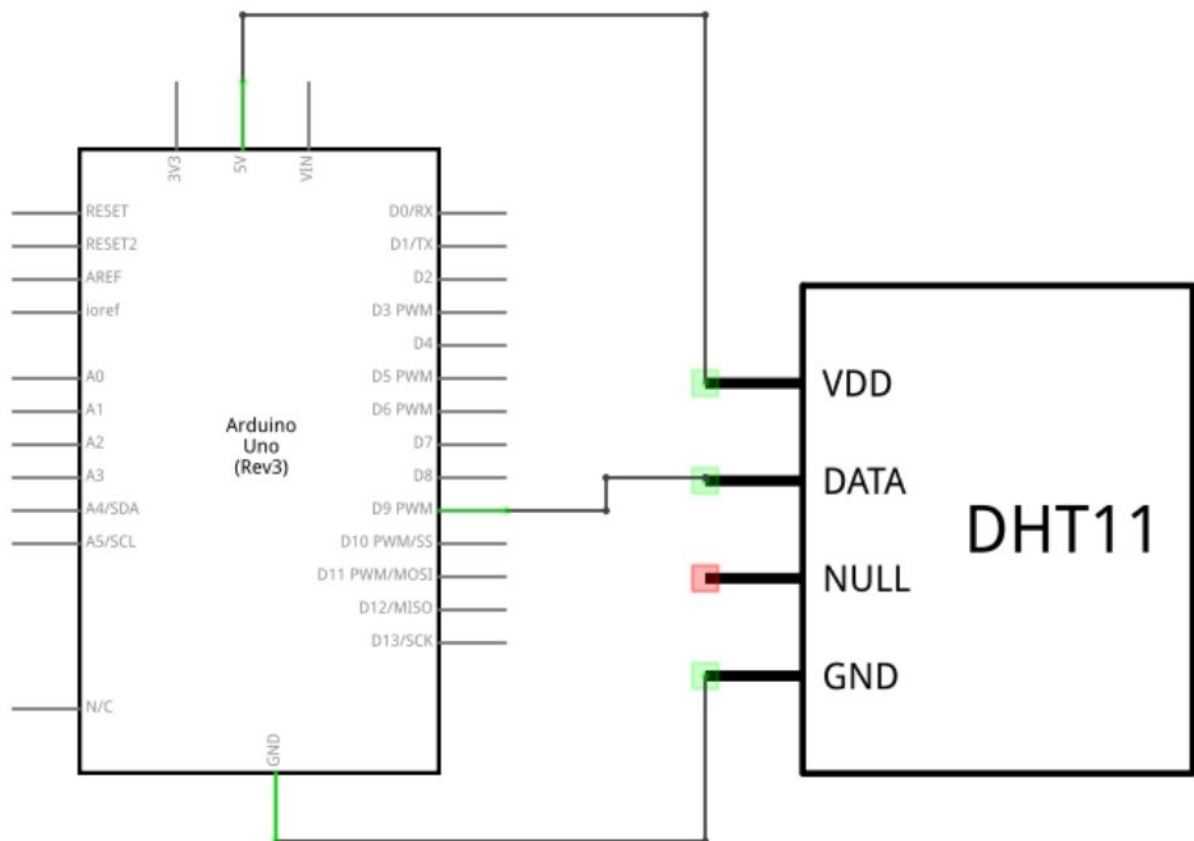
El LED el connectem a una sortida analògica PWM ja que regularem el seu voltatge d'alimentació i la seva il·luminació.

El DHT11 i DHT22 disposen de 4 patilles però només en farem servir 3, Vcc, Output i GND.

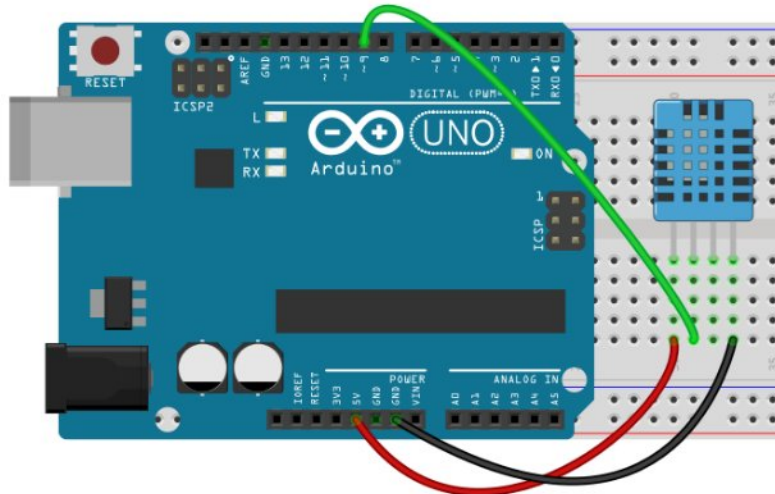


Per connectar el sensor alimentem des d'Arduino al sensor a través dels pins GND i Vcc del mateix. D'altra banda, connectem la sortida Output a una entrada digital d'Arduino.

L'esquema elèctric queda com la següent imatge



Muntatge:



Llibreria i codi:

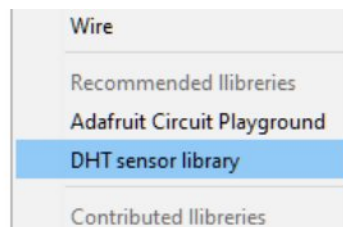
Primer de tot, descarreguem i instal·lem la llibreria DHT.h des d'aquest enllaç

[DHT-sensor-library-master](#)

Per instal·lar-la ho farem des del Menú de l'IDE d'Arduino:

[\[Menú\]](#) / [Sketch](#) / [Inclou Llibreria](#) / [Afegir Llibreria .Zip](#)

I seleccionem la llibreria descarregada. Després podrem comprovar que s'ha instal·lat correctament



Clickant sobre la llibreria instal·lada "DHT sensor library" s'afegeix l'ordre `#include <DHT.h>` a la primera fila de la pantalla de programació.

El codi complet és el següent:

```
/* TERMÒMETRE – HIGRÒMETRE DIGITAL

Mesurar temperatura i humitat mitjançant el sensor DHT11 i visualitzar els valors pel monitor sèrie.
-----
*/

/* VARIABLES I LLIBRERIES */

#include <DHT.h>                                     //Inclou la llibreria DHT.h per identificar el sensor DHT11

/*
La llibreria està formada per dos fitxers, DHT.h i DHT.cpp, que es troben dins del fitxer comprimit "DHT-sensor-
```

```

library-master.zip"
Es pot descarregar des de "https://github.com/adafruit/DHT-sensor-library" i s'instal·la dins la carpeta
C:\Users\'usuari'\Documents\Arduino\libraries\DHT-sensor-library-master
*/

/* DHTPIN, DHTTYPE, DHT11 i DHT estan definits a la llibreria DHT.h */

#define DHTPIN 9                                //El sensor DHT11 està connectat al pin digital 9 d'Arduino

#define DHTTYPE DHT11                          //Defineix el tipus de DHT = DHT11

DHT dht(DHTPIN, DHTTYPE);                     //Defineix la variable dht de la llibreria DHT.
                                              //El sensor és del tipus DHT11 i està connectat al pin 9

void setup()
{
    Serial.begin(9600);                        // Inicialitza la comunicació amb el port sèrie a 9600 bauds
    Serial.println("DHT11 test!");
    dht.begin();
}

void loop()
{
    delay(2000);                               //Espera 2 segons entre cada lectura

    float h = dht.readHumidity();              //La variable h guarda els valors llegits pel dht d'humitat
    float t = dht.readTemperature()*1.7;       //La variable t guarda els valors llegits pel dht de temperatura
                                              // 1.7 és un factor utilitzat per calibrar el valor d'humitat. Aquest valor
                                              // és experimental i depèn de cada sensor

    Serial.print("Humidity: ");                //Monitoritza els resultats mesurats pel monitor sèrie
    Serial.print(h);
    Serial.print(" %\t");
    Serial.print("Temperature: ");
    Serial.print(t);
    Serial.println(" *C ");
}

```

Referències:

- [#include](#)
- [#define](#)

4. Visualització de temperatura i humitat en una pantalla LCD 16x2

Principi:

Un display LCD (Liquid Crystal Display) és un display alfanumèric de matriu de punts que serveix per mostrar missatges a través de caràcters com lletres, números i símbols. La LCD ve equipada amb un microcontrolador que s'encarrega de generar els caràcters, polaritzar la pantalla, desplaçar el cursor.

Existeixen LCDs de diferent mida. Nosaltres farem servir un display 16 × 2 (16 columnes i 2 files).



Aquestes pantalles consten de 16 pins. D'esquerra a dreta i la seva funció és la següent:

Pin 1 - V_{SS} o GND

Pin 2 - V_{DD} o alimentació (+ 5V)

Pin 3 - Voltatge de contrast. Es connecta a un potenciòmetre.

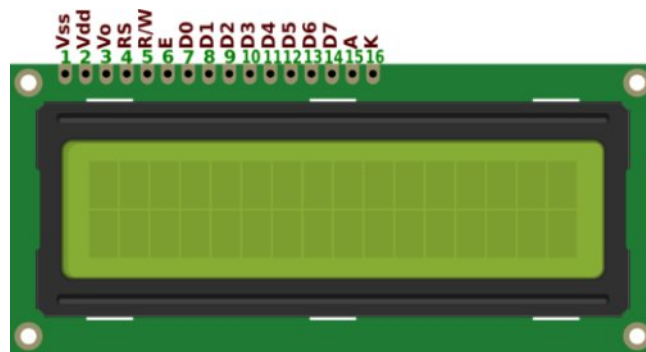
Pin 4 - RS (Register Select) Selecció de registre. Controla la memòria del display,

Pin 5 - R/W (Read/Write). Selecciona mode escriptura o lectura depenent de l'estat (HIGH o LOW),

Pin 6 - Enable. Habilita o deshabilita l'escriptura del display.

Pin 7 fins Pin 14 – Bus de dades de 8 bits per on s'envia o rep informació.

Pin 15-16 – Ànode i Càtode. Habilita la il·luminació de fons aplicant + 5v.



Objectiu: Mesurar temperatura i humitat mitjançant el sensor DHT11 i visualitzar la temperatura en graus Celcius, Fahrenheit i Kelvins i la humitat relativa pel monitor sèrie i en un display LCD 16x2.

Components:

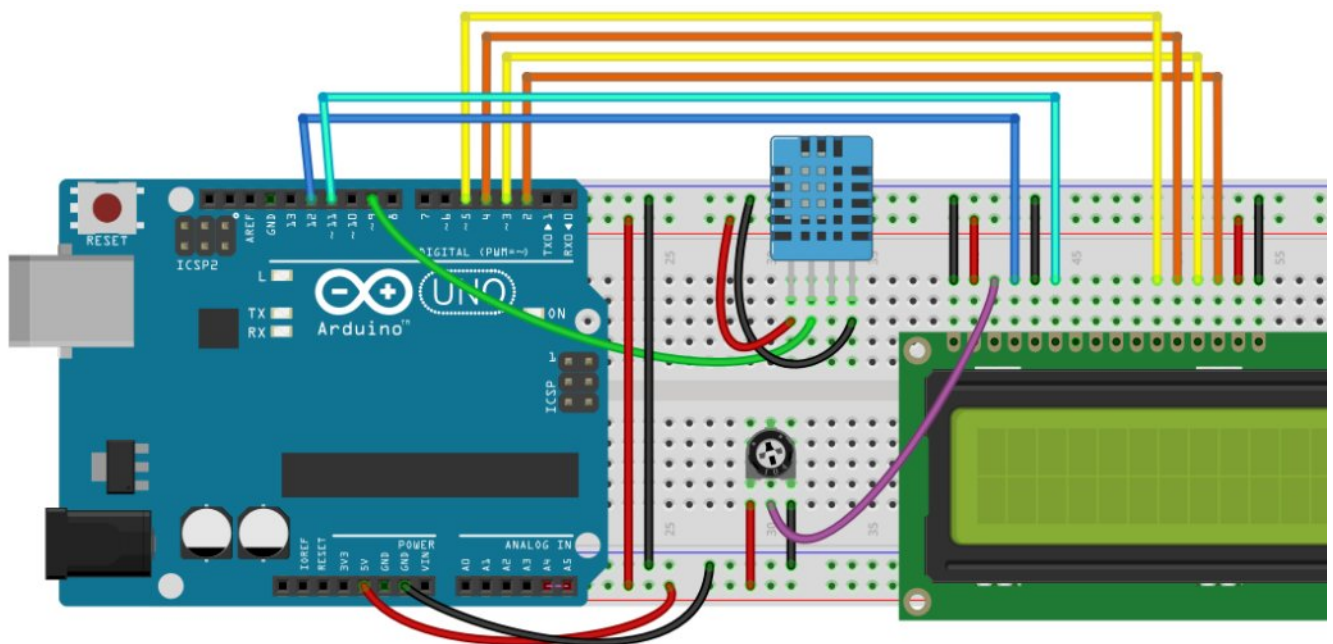
1	Arduino UNO o similar
1	protoboard
1	DHT11
1	LCD 16x2
1	Resistència variable, 10k
	cables

Circuit electrònic:

- Connectem els 16 pins de la LCD tal com indica la taula següent.
- La resistència variable la connectem al pin 3 per regular la intensitat de llum del display.
- Els pins 7, 8, 9 i 10 no es connecten.

Pin No.	Name	Description
1.	Vss	Ground pin
2.	Vcc	+5v power supply
3.	Vee	Contrast setting
4.	RS (Register select)	RS = 0; Command register is selected RS = 1; Data register is selected
5.	R/W (Read/Write)	R/W = 0; for writing R/W = 1; for reading
6.	E (Enable)	When enable pin goes from high to low the LCD takes command or data
7.	D0	Data pin0
8.	D1	Data pin1
9.	D2	Data pin2
10.	D3	Data pin3
11.	D4	Data pin4
12.	D5	Data pin5
13.	D6	Data pin6
14.	D7	Data pin7
15.	LED+	Backlight +5v
16.	LED-	Backlight Ground

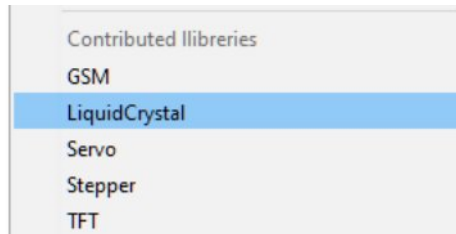
Muntatge:



Llibreria i codi:

L'IDE d'Arduino ja porta incorporada la llibreria LiquidCrystal.h i podem comprovar-ho des del Menú:

[\[Menú\] / Sketch / Inclou Llibreria](#)



Clickant sobre la llibreria “LiquidCrystal” s’afegeix l’ordre `#include <LiquidCrystal.h>` a la primera fila de la pantalla de programació.

El codi complet és el següent:

```
/* TERMÒMETRE – HIGRÒMETRE DIGITAL i LCD

Mesurar temperatura i humitat mitjançant el sensor DHT11 i visualitzar la temperatura en graus Celcius,
Fahrenheit i Kelvins i la humitat relativa pel monitor sèrie i en un display LCD 16x2.
-----
*/

/* VARIABLES I LLIBRERIES PEL SENSOR DHT11 */

#include <DHT.h>                                //Inclou la llibreria DHT.h per identificar el sensor DHT11
#define DHTPIN 9                                //El sensor DHT11 està connectat al pin digital 9 d'Arduino
#define DHTTYPE DHT11                          //Defineix el tipus de DHT = DHT 11
DHT dht(DHTPIN, DHTTYPE);                      //Defineix la variable dht de la llibreria DHT

/* VARIABLES I LLIBRERIES PEL LCD Display */

#include <LiquidCrystal.h>                      //Inclou la llibreria LiquidCrystal.h per identificar el display LCD 16x2
/* Les connexions del LCD display:
 * LCD RS pin to digital pin 12
 * LCD Enable pin to digital pin 11
 * LCD D4 pin to digital pin 5
 * LCD D5 pin to digital pin 4
 * LCD D6 pin to digital pin 3
 * LCD D7 pin to digital pin 2
 * LCD R/W pin to ground (perquè és de lectura)
 * LCD VO pin to Resistència variable (regula la brillantor del LCD)
 */

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);          //Inicialitza la llibreria amb els números dels pins de l'Arduino

void setup()
{
  Serial.begin(9600);                          // Inicialitza la comunicació amb el port sèrie a 9600 bauds
  Serial.println("DHT11 test!");
  dht.begin();

  lcd.begin(16, 2);                            //Configura el nombre de columnes i files del display
}
```

```

void loop()
{

    float h = dht.readHumidity();           //La variable h guarda els valors llegits pel dht d'humitat
    float t = dht.readTemperature();        //La variable t guarda els valors llegits pel dht de temperatura

    Serial.print("Humitat: ");              //Imprimeix els resultats mesurats pel monitor sèrie
    Serial.print(h);
    Serial.print(" %\t");
    Serial.print("Temperatura: ");
    Serial.print(t);
    Serial.println(" *C ");

    //-----

    /* Mostra "Humitat: " i la lectura h al LCD durant 3 segons */
    lcd.print("Humitat: ");
    lcd.setCursor(9, 1);                   //(nota: (9, 1) posa el cursor a la columna 9 i a la fila inferior
    lcd.print(h*1.7);                      // 1.7 és el factor que s'utilitza per calibrar la humitat. Aquest valor és
                                           // experimental i depèn de cada sensor

    lcd.print(" %");
    delay(3000);
    lcd.clear();

    /* Mostra "Temperatura: " i la lectura t en Celcius, Kelvins i Fahrenheit al LCD durant 9 segons */

    lcd.print("Temperatura: ");            //Temperatura en graus Celcius
    lcd.setCursor(11, 1);
    int tC = t;
    lcd.print(tC);
    lcd.print(" oC");
    delay(3000);
    lcd.clear();

    lcd.print("Temperatura: ");            //Temperatura en Kelvins
    lcd.setCursor(11, 1);
    int tk = t + 273;
    lcd.print(tk);
    lcd.print(" K");
    delay(3000);
    lcd.clear();

    lcd.print("Temperatura: ");            //Temperatura en graus Fahrenheit
    lcd.setCursor(11, 1);
    int tf = 9/5*t + 32;
    lcd.print(tf);
    lcd.print(" oF");
    delay(3000);
    lcd.clear();

}

```

Referències:

- [LiquidCrystal](#)

5. Connectar un display LCD a Arduino mitjançant un controlador I²C

Principi:

El controlador de LCD I²C és un dispositiu que ens permet controlar una pantalla a través del bus I²C (Inter Integrated Circuits bus, o IIC), usant únicament dos cables en lloc de la quantitat utilitzada en la pràctica anterior. També incorpora un potenciòmetre per regular el backlight del LCD.

Algunes especificacions del protocol I²C que ens interessin en el nostre controlador són:

- Protocol de dos fils de control, un per transmetre les dades, SDA i un altre, el rellotge asíncron que indica quan llegir les dades SCL.
- Cada dispositiu connectat al bus I²C té la seva adreça exclusiva de 7 bits, (així que, en teoria, podem connectar $2^7 = 128$ dispositius).

Tots els components units al bus I²C es connecten en paral·lel a les dues línies del Bus, SDA i SCL. Només pot haver-hi un màster, en aquest cas, el nostre Arduino i els altres es configuren com a Slave. Per fer-ho, l'Arduino utilitza dos dels pins analògics, A4 i A5, per a les funcions SDA (Dades) i SCL (Clock) respectivament.

Objectiu: Mesurar temperatura i humitat mitjançant el sensor DHT11 i visualitzar la temperatura en graus Celcius, Fahrenheit i Kelvins i la humitat relativa en un display LCD 16x2 utilitzant un controlador LCD I2C. Bàsicament, incorporarem el controlador LCD I2C a la pràctica anterior per simplificar el circuit.

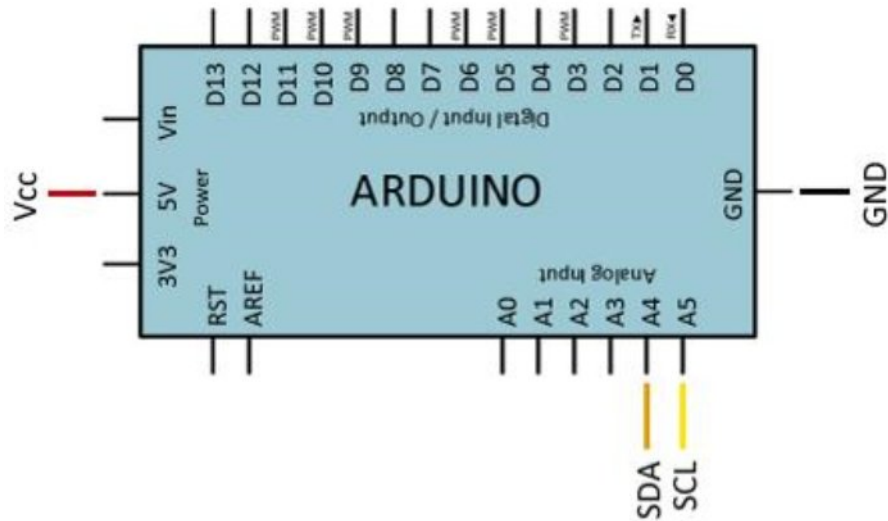
Components:

1	Arduino UNO o similar
1	protoboard
1	DHT11
1	LCD 16x2
1	controlador LCD I2C
	cables

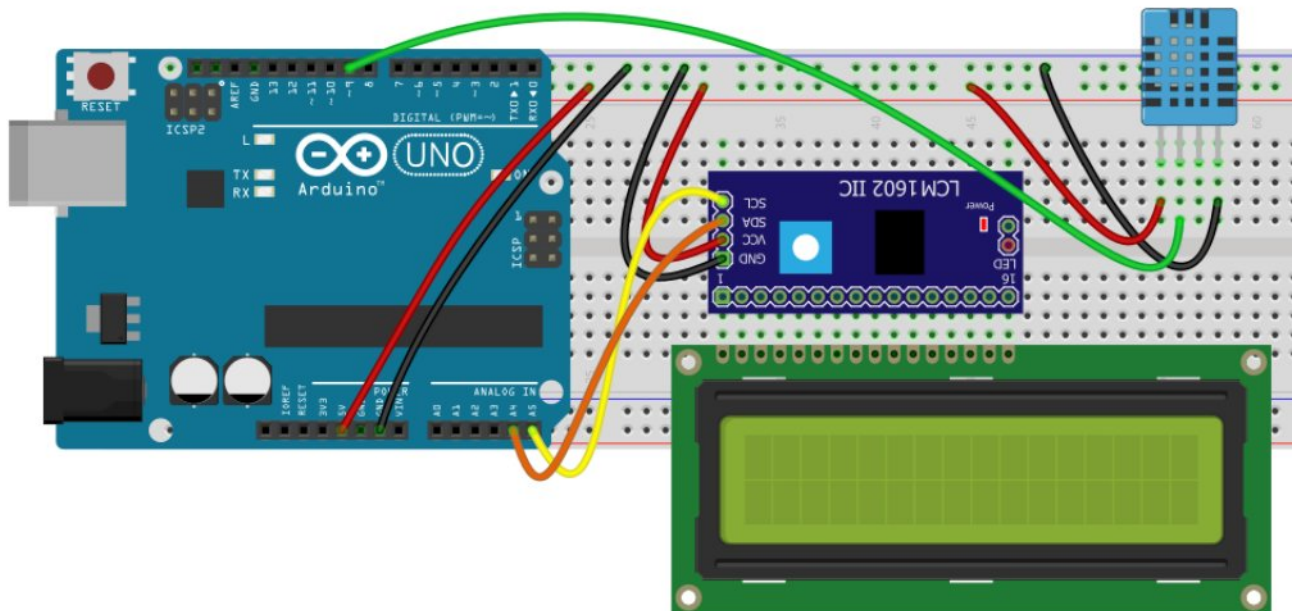
Circuit electrònic:

Desconnectem tots els cables i la resistència variable del circuit anterior i connectem el controlador LCD I2C fent coincidir els seus pins amb els del display.





Muntatge:



Llibreria i codi:

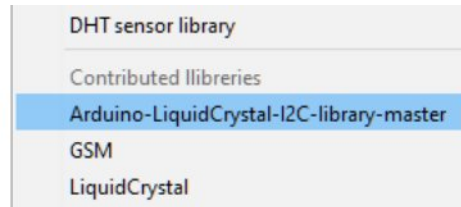
Primer de tot, descarreguem i instal·lem la llibreria LiquidCrystal_I2C.h des d'aquest enllaç

[Arduino-LiquidCrystal-I2C-library-master.zip](https://github.com/liquidcrystal/Arduino-LiquidCrystal-I2C-library-master.zip)

Per instal·lar-la ho farem des del Menú de l'IDE d'Arduino:

[\[Menú\] / Sketch / Inclou Llibreria / Afegir Llibreria .Zip](#)

I seleccionem la llibreria descarregada. Després podrem comprovar que s'ha instal·lat correctament



Clickant sobre la llibreria instal·lada “Arduino-LiquidCrystal-I2C-library-master” s’afegeix l’ordre `#include <LiquidCrystal_I2C.h>` a la primera fila de la pantalla de programació.

Del codi de la pràctica anterior només caldrà canviar alguna línia de programació en la definició de variable i llibreries i en el setup:

```
/* TERMÒMETRE – HIGRÒMETRE DIGITAL , LCD i CONTROLADOR LCD I2C

Mesurar temperatura i humitat mitjançant el sensor DHT11 i visualitzar la temperatura en graus Celcius,
Fahrenheit i Kelvins i la humitat relativa en un display LCD 16x2 utilitzant un controlador LCD I2C.
-----
*/

/* VARIABLES I LLIBRERIES PEL SENSOR DHT11 */

#include <DHT.h>                                //Inclou la llibreria DHT.h per identificar el sensor DHT11
#define DHTPIN 9                                //El sensor DHT11 està connectat al pin digital 9 d'Arduino
#define DHTTYPE DHT11                           //Defineix el tipus de DHT = DHT 11
DHT dht(DHTPIN, DHTTYPE);                       //Defineix la variable dht de la llibreria DHT

/* VARIABLES I LLIBRERIES PEL LCD Display i LCD I2C */

#include <Wire.h>                                // Inclou la llibreria I2c que en Arduino es diu Wire
#include <LiquidCrystal_I2C.h>                   // Inclou la llibreria del controlador LCD I2C

LiquidCrystal_I2C lcd(0x3F, 16, 2);              // Inicialitza l’LCD en la direcció 0x3F, amb 16 caràcters y 2 línies
/* Precaució: en molts tutorials que podem trobar per internet es fixa la direcció 0x27 però en el nostre model
hem d'utilitzar la 0x3F.
*/

void setup()
{
  Serial.begin(9600);                            // Inicialitza la comunicació amb el port sèrie a 9600 bauds
  Serial.println("DHT11 test!");
  dht.begin();

  lcd.begin();                                   // Inicialitza l’LCD
}

void loop()
{
  /* Igual que el programa anterior */
}
```