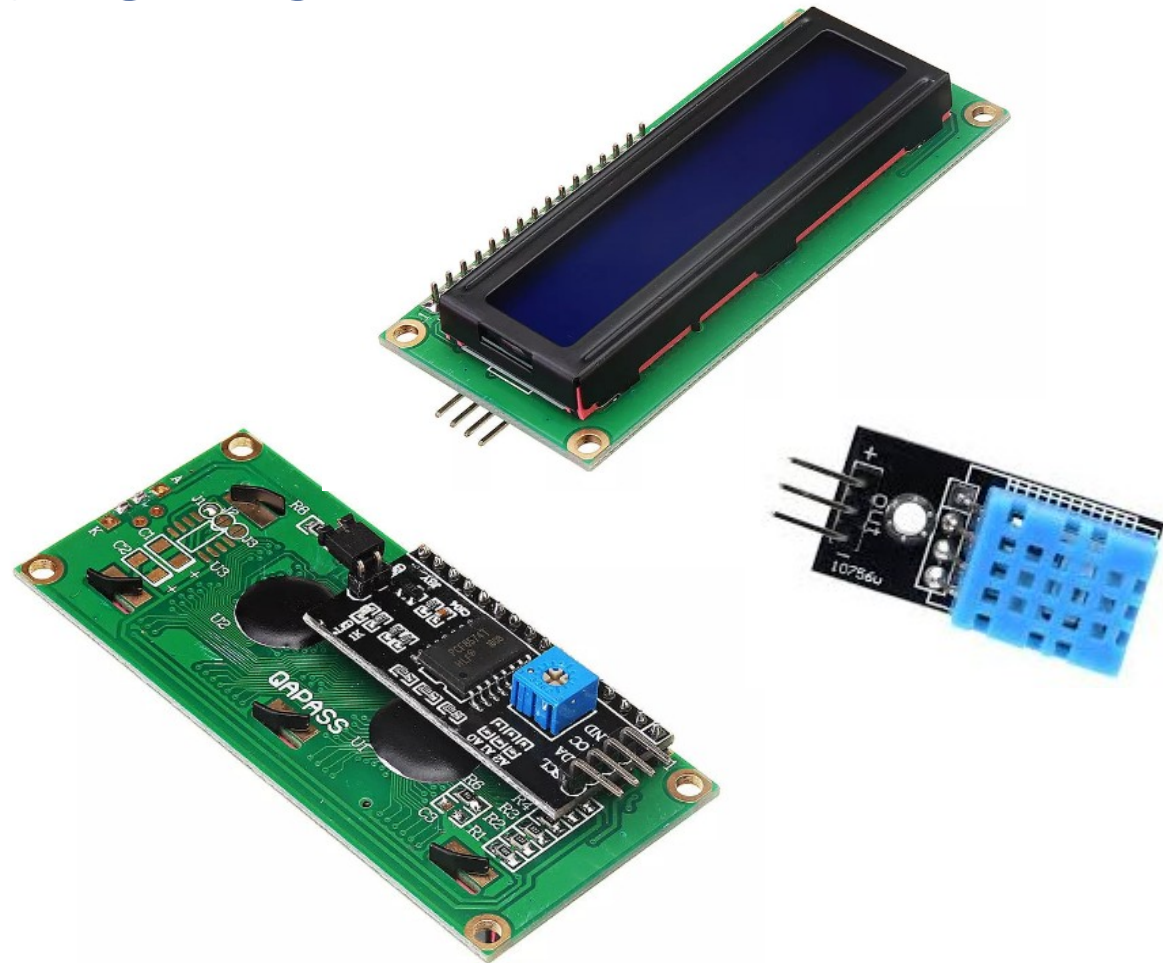


DISPLAY LCD I2C (PCF8574T) ARDUINO



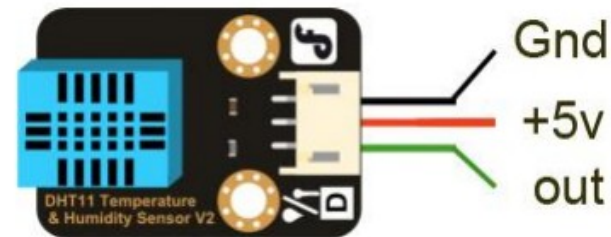
Come abbiamo visto il DHT-11 è un sensore digitale in grado di misurare la temperatura e l'umidità nell'aria.

Fornisce in output un segnale digitale proporzionale alla temperatura e all'umidità misurata dal sensore stesso. La tecnologia con cui è realizzato il sensore DHT11 assicura un'elevata affidabilità ed un'eccellente stabilità a lungo termine nonché tempi di reazione molto rapidi.

Ogni componente DHT11 è accuratamente calibrato in laboratorio.

Il coefficiente di calibrazione è memorizzato in una memoria OTP interna e tale valore è utilizzato durante il processo di acquisizione. L'interfaccia seriale a singolo filo rende semplice e veloce l'integrazione del sensore in sistemi digitali.

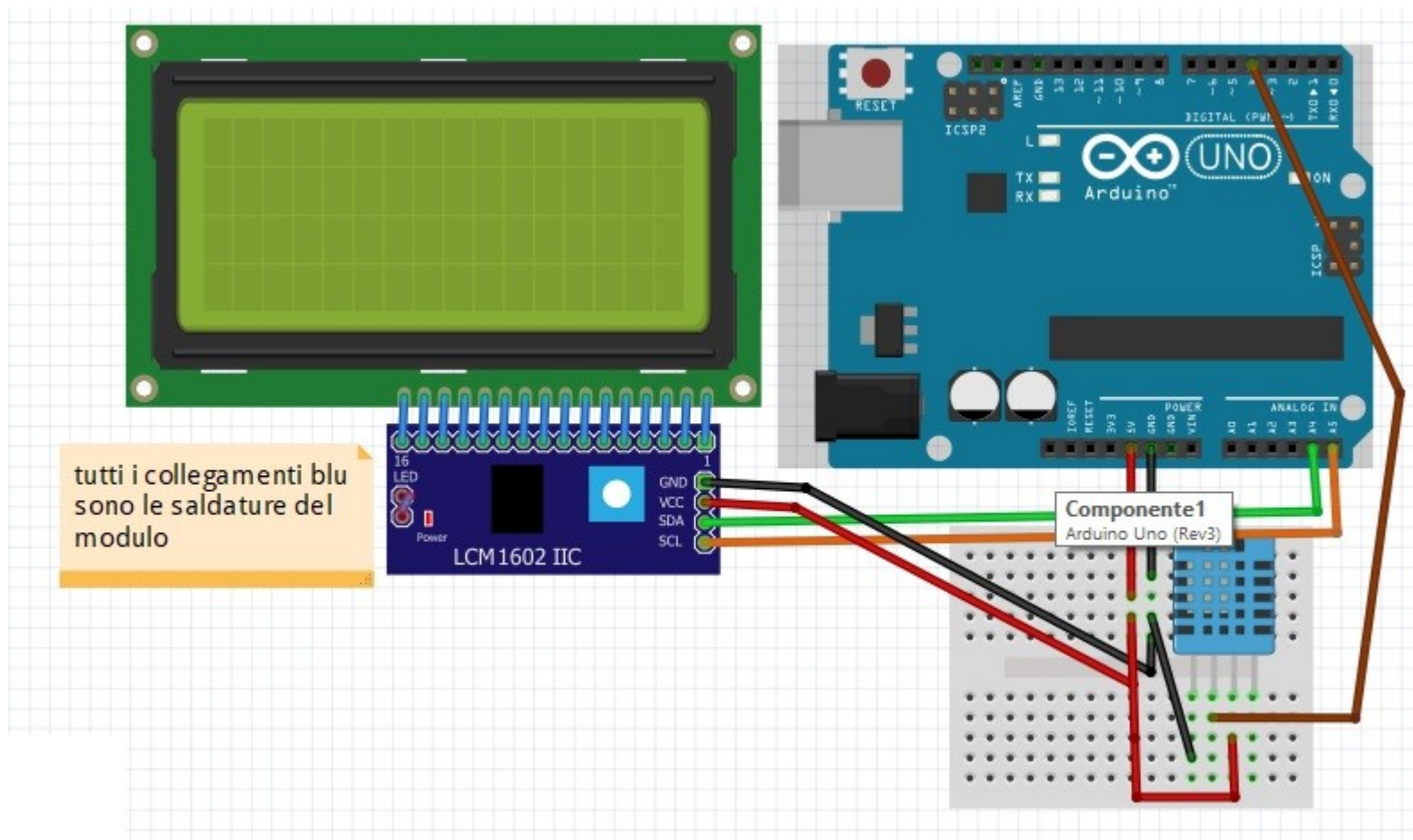
La scheda si interfaccia con l'esterno tramite un connettore a 3-pin . Due sono l'alimentazione e la massa necessari per il funzionamento del sensore, il terzo è il segnale in output del sensore.



Nell'esempio che vedremo andremo ad interfacciare il DHT-11 con Arduino e mostreremo i valori sul Display LCD I²C



Schema del circuito:



andremo ad utilizzare i pin **SDA** e **SCL** presenti sulla board Arduino.
Ecco una tabella con i rispettivi pin SDA e SCL delle board Arduino più utilizzate.

Board	I2C / TWI pins
Uno, Ethernet	A4 (SDA), A5 (SCL)
Mega2560	20 (SDA), 21 (SCL)
Leonardo	2 (SDA), 3 (SCL)
Due	20 (SDA), 21 (SCL), SDA1, SCL1

COLLEGAMENTI

PCF8574T		ARDUINO
SDA	->	SDA
SCL	->	SCL
VCC	->	+5V
GND	->	GND

Prima di caricare il programma, inserire le librerie per il display I2C e per il DHT-11.
Qui i link per scaricare le librerie:

Dispaly I2C

https://drive.google.com/open?id=1ulkl7_PwqIkqq4Ax7gXbD2tPey2pWsn8

DHT-11

<https://drive.google.com/open?id=1WD5w8vCFDjHFACC4iYvo1PWubT2EiMTB>



Nell'esempio la lettura dei valori forniti dal sensore e la lettura dei valori massimi e minimi registrati dal nostro sensore



```
#include <LiquidCrystal_I2C.h> //libreria display LCD
LiquidCrystal_I2C lcd(0x27, 20, 4);
//libreria sensore dht-11
#include <dht11.h>
dht11 DHT;
#define DHT11_PIN 4//dichiarazione pin
int temp = 0; //raccoglie misura temperatura
int um = 0; //raccoglie misura umidità
int temp_min = 0; //si ricorda della temperatura min. registrata
int um_min = 0; //si ricorda dell'umidità min. registrata
int temp_max = 0; //si ricorda della temperatura max. registrata
int um_max = 0; //si ricorda dell'umidità min. registrata
int v1, v2; //variabili iniziali per um_min e temp_min

void setup()
{
  lcd.init();
  lcd.backlight();
}
```



```
void loop()
{
  lcd.clear();
  lcd.setCursor(3, 0);
  lcd.print("DHT-11");
  lcd.setCursor(1, 1);
  lcd.print("Temp and Umidity");
  int chk = DHT.read(DHT11_PIN); //abilito lettura dht11
  temp = DHT.temperature; //raccolgo misura temperatura
  um = DHT.humidity; //raccolgo misura umidità
  //stampo valori sul display lcd
  lcd.setCursor(3, 0);
  lcd.print("DHT-11");
  lcd.setCursor(1, 1);
  lcd.print("Temp and Umidity");
  lcd.setCursor(0, 2);
  lcd.print("Umidita': ");
  lcd.print(DHT.humidity, 1);
  lcd.print("%");
  lcd.setCursor(0, 3);
  lcd.print("Temp: ");
  lcd.print(DHT.temperature, 1);
  lcd.print((char)223);
  lcd.print("C");
```



```
delay(3000);
//riempio variabili um_min e tem_min per la misura iniziale
if (v1 == 0) { um_min = um;}
if (v2 == 0) { temp_min = temp;}
if (temp > temp_max) { temp_max = temp;}
if (um > um_max) { um_max = um; }
if (um_min >= um) { um_min = um; v1++; }
if (temp_min >= temp) { temp_min = temp; v2++; }
//stampo statistiche: um_max, um_min, temp_max, temp_min
lcd.clear();
lcd.setCursor(3, 0);
lcd.print("DHT-11");
lcd.setCursor(1, 1);
lcd.print("Temp and Umidity");
lcd.setCursor(0, 2);
lcd.print("Umid max: ");
lcd.print(um_max);
lcd.print("%");
lcd.setCursor(0, 3);
lcd.print("Temp max: ");
lcd.print(temp_max);
lcd.print((char)223);
lcd.print("C");
```




```
delay(3000);  
lcd.clear();  
lcd.setCursor(3, 0);  
lcd.print("DHT-11");  
lcd.setCursor(1, 1);  
lcd.print("Temp and Umidity");  
lcd.setCursor(0, 2);  
lcd.print("Umid min: ");  
lcd.print(um_min);  
lcd.print("%");  
lcd.setCursor(0, 3);  
lcd.print("Temp min: ");  
lcd.print(temp_min);  
lcd.print((char)223);  
lcd.print("C");  
delay(3000);  
}
```

