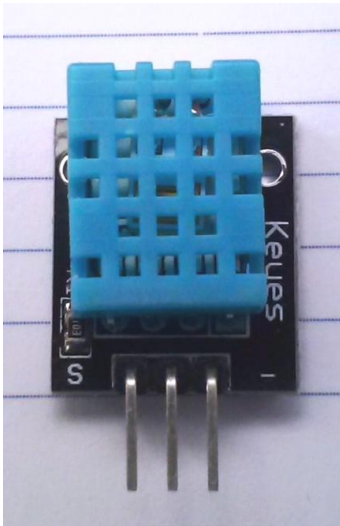


28- DHT11 umidità e temperatura – temperature & moisture

(some notes at section end)



Il modulo DHT11 è un sensore composto che combina, in un sol corpo, un termometro ed un igrometro.

Esistono parecchi moduli della serie DHT ed il modulo 11, trattato in questo esercizio, è il più semplice ed il più economico.

Il modulo misura i valori di umidità e temperatura e, attraverso un microcontrollore ad 8 bit in esso racchiuso, li trasforma in segnali digitali.

Si tratta di quindi di un componente evoluto, composto da un sensore di umidità di tipo resistivo, un sensore di temperatura di tipo NTC (Negative Temperature Coefficient - un sensore di analogico che diminuisce l'impedenza di una resistenza all'aumentare della temperatura) e da un microcontrollore.

Questo sensore, su sollecitazione di Arduino, trasmette sul pin S (signal) un treno di quaranta bit:

- 8 bit per indicare la parte intera del valore di umidità
- 8 bit per indicare la parte decimale dell'umidità
- 8 bit per indicare la parte intera della temperatura
- 8 bit per indicare la parte decimale della temperatura
- 8 bit per indicare il numero di controllo (per validare il valore dei precedenti 32 bit)

La gestione del segnale è ovviamente a carico di una libreria (la DHT.h) che deve essere scaricata ed installata prima della compilazione del programma.

Per scaricare ed installare la libreria:

- Aprire questo link (github è un sito sicuro) : <https://github.com/RobTillaart/Arduino>
- Scaricare tutte le librerie premendo il bottone "download ZIP" (sulla destra dello schermo)
- Disimpaccare il file e ottenere una cartella con decine di librerie, tra cui la libreria DHTlib che a sua volta contiene una cartella example, e tre files: dht.cpp e dht.h ed un readme
- Rinominare in "DHT" la libreria "DHTlib"
- Installare la libreria DHT nella libreria di arduino (IDE->sketch->importa libreria->add library ->indirizzo della nuova libreria DHT->apri)
- Chiudere e riaprire l'IDE per rendere operativa la libreria
- Compilare il programma presente in IDE->file->esempi_>DHT->DHT11_test per verificare l'avvenuta installazione della libreria

Se tutto è andato bene si può procedere alla esecuzione di questo esercizio che misura umidità e temperatura e li riporta su di un display LCD con driver I2C.

In assenza del display lcd è comunque possibile leggere umidità e temperatura anche sul monitor seriale.

Nota: Questo esercizio e questa nota sono parte di una serie che vede protagonisti arduino ed alcuni dei componenti ad esso collegabili. Per la maggior parte degli esercizi e' anche disponibile un filmato su youtube.

- [Esercizi facenti parte della raccolta](#)
- [Filmati presenti su youtube](#)
- [Informazioni su arduino e sui componenti collegabili \(PDF scaricato nell'area di download\)](#)
- [Breve manuale di programmazione \(PDF scaricato nell'area di download\)](#)

Per eventuali chiarimenti o suggerimenti sul contenuto di questa scheda scrivere a giocarduino@libero.it

Here some notes about this project, translated by google translator



The DHT11 device, is a composite sensor that combines, in one body, a thermometer and an hygrometer. There are several device in DHT series and the 11, treated in this project, is the most simple and economic. Module measures moisture & temperature values and, via an enclosed 8-bit microcontroller, transforms them into digital signals. Is therefore an evolved component, composed of an humidity sensor (resistive type), an NTC temperature sensor (Negative Temperature Coefficient - an analog sensor that decreases a resistor impedance as the temperature increases) and by a microcontroller.

This device, on Arduino solicitation, transmits on pin S (signal) a forty-bit train:

- 8 bits to indicate the moisture integer part
- 8-bit to indicate the moisture decimal part
- 8 bits to indicate the temperature integer part
- 8 bits to indicate the temperature decimal part
- 8-bit to indicate the control number (to validate the previous 32-bit value)

The signal is managed by a library (the DHT.h) that must be downloaded and installed before program compiling

To library download and install:

- Open this link (github is a secure site): <https://github.com/RobTillaart/Arduino>
- Download all libraries pressing "Download ZIP" button (on the right)
- Unzipp file and get a folder with dozens of libraries, including the library DHTlib which in turn contains an example folder, three files: dht.cpp and dht.h and a readme
- Rename "DHTlib" folder to "DHT"
- Install DHT library: Arduino (IDE-> sketch-> Library-> add library -> address of the new library DHT-> open)
- Close and reopen the IDE to operationalize the library
- Fill out a program in IDE-> file-> examples_> DHT-> DHT11_test, to verify the library successful installation

If everything went well you can proceed with the execution of this project which measures moisture and temperature and brings them on an LCD display, driven by a I2C driver. In absence of an LCD display you can still read moisture and temperature even on the serial monitor.

Note: This project and this note is part of a series that sees, as main characters, Arduino and some of connectable components. For most projects there is also a video on youtube.

- [Projects collection](#)
- [Movies on youtube](#)

Arduino: DHT11 - umidità e temperatura – Temperature & moisture

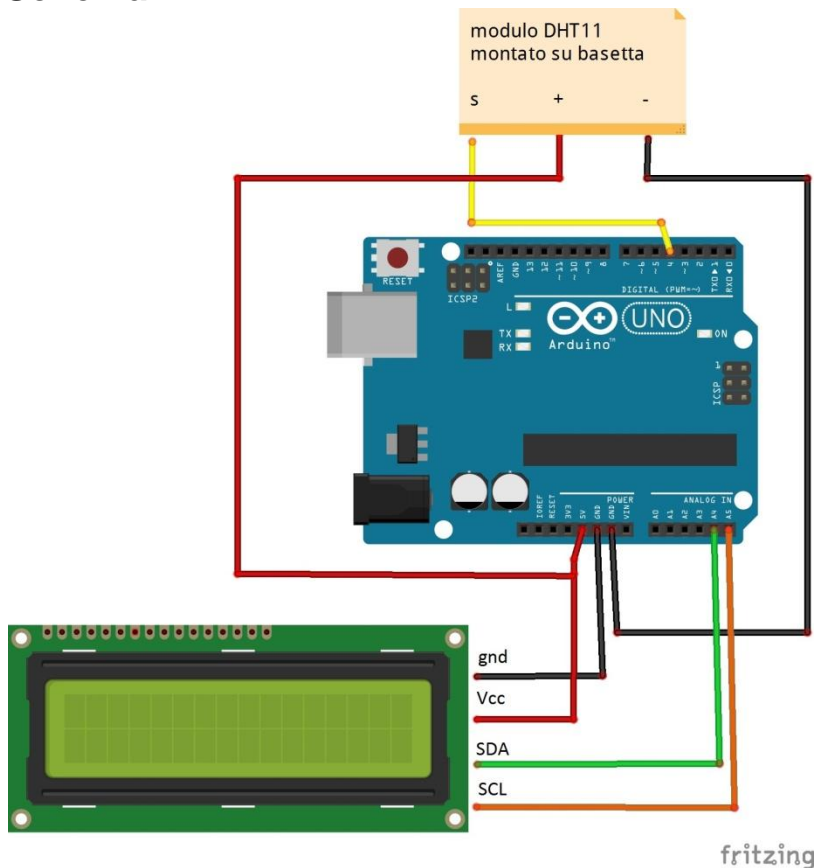
- [About Arduino and components \(italian; pdf will be downloaded in your download area\)](#)
- [Quick programming guide \(almost english; pdf will be downloaded in your download area\)](#)

For any questions or suggestions about this note (and on its english translation), please write to giocarduino@libero.it (simple words and short sentences, please)

Materiali

- Un modulo DHT11 montato su basetta keys
- Un display lcd con driver LCMI602 (opzionale)
- Un po' di cavetteria

Schema



Programma

```
/* Attenzione: facendo il copia/incolla dal PDF all'IDE si perde la formattazione del testo.
Per rendere piu' facilmente leggibile il programma e' opportuno formattarlo subito dopo il
trasferimento nell'IDE, premendo CTRL+T - Warning: cut&paste from PDF to IDE loses formatting.
To restore it press CTRL + T.
*/
```

```
#include <dht.h> // richiama la libreria DHT

/* le prossime quattro righe sono da cancellare se non si dispone di un display lcd con driver LCMI
- Next four lines must be erased if you don't have a 1602 lcd display with a I2C driver */
#include <Wire.h> // libreria wire presente, di default, nell'IDE
#include <LiquidCrystal_I2C.h> // libreria di gestione del display lcd
//.....addr en,rw,rs,d4,d5,d6,d7,bl,blpol
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE); // definisce la tipologia del display

dht DHT;
int chk; // variabile di memorizzazione del risultato della lettura
```

Arduino: DHT11 - umidita' e temperatura – Temperature & moisture

```
#define DHT11_PIN 4

void setup()
{
  lcd.begin(16, 2);    // cancella se no lcd - erase if no lcd
  Serial.begin(9600); // inizializza il monitor seriale
}

void loop()
{
  Serial.print("DHT11, \t");
  int chk = DHT.read11(DHT11_PIN); // legge i valori forniti dal sensore
  switch (chk)
  {
    case DHTLIB_OK:
      Serial.println("OK,\t");
      break;
    case DHTLIB_ERROR_CHECKSUM:
      Serial.println("errore di trasmissione,\t");
      break;
    case DHTLIB_ERROR_TIMEOUT:
      Serial.println("Timeout,\t");
      break;
    default:
      Serial.println("errore sconosciuto,\t");
      break;
  }
  // ***** display dei dati su lcd - display data on lcd *****
  lcd.setCursor(0, 0); // cancella se no lcd - erase if no lcd
  lcd.print("Tem:"); // cancella se no lcd - erase if no lcd
  lcd.print(DHT.temperature,1); // cancella se no lcd - erase if no lcd
  lcd.print(" C"); // cancella se no lcd - erase if no lcd
  lcd.setCursor(0, 1); // cancella se no lcd - erase if no lcd
  lcd.print("Hum:"); // cancella se no lcd - erase if no lcd
  lcd.print(DHT.humidity,1); // cancella se no lcd - erase if no lcd
  lcd.print(" %"); // cancella se no lcd - erase if no lcd
  // ***** display dei dati su monitor seriale display data on serial monitor *****
  Serial.print("Temperatura: ");
  Serial.print(DHT.temperature,1);
  Serial.println(" C ");
  Serial.print("Umidita: ");
  Serial.print(DHT.humidity,1);
  Serial.println(" %");
  delay(2000); // attende due secondi prima di controllare nuovamente umidita' e temperatura
}
```