



Davide Bagnoli (DADO91)

WEB STAZIONE METEO CON ARDUINO [2] - LA RILEVAZIONE DEI DATI

5 January 2012

Eccomi di nuovo qui a parlare del mio progettino, oggi parleremo di quali dati meteorologici voglio misurare e di quali sensori ho deciso di utilizzare per le rilevazioni.



Wheater-duino.png

Premessa

Ho intenzione di sviluppare il tutto senza spendere grosse cifre, dato che stiamo parlando di rilevazioni amatoriali e soprattutto di un'occasione per imparare a far interagire tra loro mondi tanto diversi, ma anche molto simili come l'informatica e l'elettronica.

I Dati

Come dicevamo nella scorsa puntata, ho deciso di cominciare con le rilevazioni della sola **temperatura** e **umidità relativa**. Ogni sedicente meteorologo potrebbe subito obbiettare:

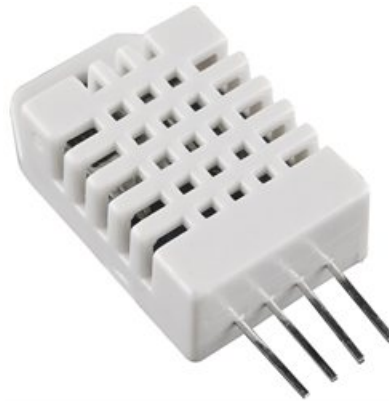
"Come puoi fare una qualsivoglia previsione del tempo se non conosci la pressione atmosferica ? "

Beh, diciamo che per adesso non voglio prendere il posto di Giuliacci, ma mi basta avere delle informazioni più dettagliate rispetto ad una sveglia con sensore esterno, che mi fornisce solo la temperatura. Poi in un futuro non si sa mai...

Il Sensore

Rimanendo fedele alle linee guida del progetto ed essendo in tempo di "Crisi", (dato che questa parola va tanto di moda la voglio utilizzare anch'io), ho scovato un sensore abbastanza economico che effettua entrambe le misure,

Stiamo parlando dell' **DHT22**



humidity temperature sensor dht22

Questo sensore utilizza una comunicazione di tipo digitale seriale, quindi vengono meno alcuni problemi come:

- La resistenza del cavo di connessione, che può inficiare la misurazione
- La complessità di connessione, dato che necessita solo di tre cavi (Vcc,Data,Gnd)

Vediamone alcune caratteristiche:

- Tens.Alimentazione : **3.3-6V**
- Assorbimento in Misura : **1-1.5mA**
- Assorbimento in StandBy : **40-50 uA**
- Umidità : **0 / 100% Relativa**
- Temperatura : **-40 / +80 °C**
- Precisione Umidità : **0.1%**
- Precisione Temperatura **0.1°C**
- Errore Umidità : **+ -2% Relativa**
- Errore Temperatura : **+ -0.5 °C**

Per la logica di trasmissione delle informazioni di questo sensore, vi rimando al datasheet: [Datasheet DHT22](#)

L'unico neo del sensore è la sua lentezza, infatti si parla di circa 2 s tra una lettura e la successiva.

La Lettura delle Misure

Come avevamo detto, le misure verranno fatte da una scheda **Arduino Uno** e utilizzando la [Libreria DHT](#) per la lettura dei dati dal sensore. Possiamo sfruttare a nostro vantaggio l'unico neo del nostro sensore, infatti, dato che non dobbiamo solamente rilevare un dato, ma anche inviarlo tramite internet ed aspettare una conferma dell'avvenuta ricezione, posso utilizzare questo tempo morto, per implementare una tecnica di programmazione di Arduino, che permette di avere una specie di Multitasking, e quindi portare avanti vari compiti in "simultanea". Ovviamente non parliamo di un vero multitasking, ma possiamo comunque occupare i tempi morti della cpu per fare altro.

Gestire gli errori

Può capitare ed è bene sempre tenerne conto, di cosa sto parlando? Ovviamente degli **errori!** Analizziamo dove possono avvenire degli errori nelle varie fasi in cui gestisco i dati



Lo schema qui sopra schematizza i vari componenti del progetto. Per oggi ci limitiamo a parlare degli errori nei primi due blocchi : [DHT22] e [Arduino].

DHT22

Gli errori a livello del sensore possono verificarsi in due occasioni:

1. Durante la misura da parte della circuiteria interna
2. Durante la trasmissione del Dato ad Arduino

Possiamo già eliminare qualsiasi preoccupazione per il **secondo punto**, dato che nella comunicazione seriale, il sensore inserisce un byte di CRC per il controllo degli errori.

Per il **primo punto** però, non possiamo avere delle certezze, ma ci possiamo solamente affidare a ciò che il produttore ha costruito. Ci viene in contro però, la statistica, che fornendoci la Media Aritmetica, ci da una mano a minimizzare gli effetti di eventuali errori casuale, che andrebbero ad inficiare una misura. Quindi faremo un serie di misure in un dato arco di tempo per poi farne la media e trovare quindi il valore che cercavamo come misura.

Arduino

Adesso parliamo degli errori che potrebbero accadere in Arduino, più precisamente durante la programmazione. Dato che abbiamo parlato di fare la media di una serie di misurazioni e che tali misurazioni sono restituite dalla libreria DHT come valori **float**, dobbiamo assicurarci durante l'elaborazione dei dati di non utilizzare formule matematiche con valori non a virgola mobile o con una precisione inferiore, per non perdere precisione nelle misure. Altra attenzione deve essere fatta nell'utilizzo di costanti nei calcoli, che in presenza di variabili in virgola mobile devono presentare il "**punto**"

Esempio: *Costante Intera : 100 Costante Floating Point : 100.0*

Codice Esempio : `float r = 66/100 //r = 1`

`float r = 66/100.0 //r. = 0.66`

Conclusioni

Abbiamo fatto una carrellata di ciò che deve essere tenuto in considerazione quando si lavora con dei sensori e con la loro programmazione. Vi aspetto per le prossime puntate. P.S. a breve posterò qualche prova del sensore e dei codici di esempio.

Sitografia

[Multitasking Arduino](#)

Link agli articoli del progetto

[Web Stazione Meteo con Arduino \[1\] - L'idea](#)

[Web Stazione Meteo con Arduino \[3\] - Il Primo Circuito di Prova](#)

[Web Stazione Meteo con Arduino \[4\] - Il circuito definitivo e la comunicazione http](#)

[Web Stazione Meteo con Arduino \[5\] - Appendice 1 : Le Reti spiegate a mia Nonna](#)

[Web Stazione Meteo con Arduino \[6\] : Il Sito Web](#)

[Web Stazione Meteo con Arduino \[7\] : Bug Fixing](#)

Estratto da "<http://www.electroyou.it/mediawiki/index.php?title=UsersPages:Dado91:web-stazione-meteo-con-arduino-2-la-rilevazione-dei-dati-1>"