

Diagramma di temperatura di una settimana lincato da Pakube.com (ora cosm.com)



Metto di seguito il programmino desunto in gran parte dalla rete ma per ora riesco solo a inviare il valore di un sensore.

Qualcuno riesce a inviarne 2 o più??

Per contatti info@fotoarrigo.com

/*

Pachube client --- 1 sensore

Questo programma collega un sensore SHT11 tramite Pachube
(<http://www.pachube.com>)

Circuito:

- * Sensore collegato pin 2 e 3
- * sensore luminosità in A3
- * Ethernet shield collegato ai piedini 10, 11, 12, 13

UPDATE:

- * Utilizzando Pachube di API V2
- * IP pachube.com
- * Per il momento scrive solo la Temperatura

*/

```
#include <SHT1x.h>
#include <SPI.h>
#include <Ethernet.h>
#include <Wire.h>
```

```
// assegnare un indirizzo MAC per il controller ethernet.
```

```
// inserisci il tuo indirizzo qui:
```

```
byte mac[] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
```

```
// assegnare un indirizzo IP per il controller:
```

```
byte ip[] = {192,168,0,75};
```

```
byte gateway[] = {192,168,0,1};
```

```
byte subnet[] = {255, 255, 255, 0 };
```

```
// L'indirizzo del server a cui si desidera connettersi (pachube.com):
```

```

    byte server[] = { 173,203,98,29 };
// inizializza l'istanza alla libreria:
    Client client(server, 80);
// variabili
    int a;
    int luce;
    float temp_c;
    int humidity;
//=====
    long lastConnectionTime = 0;    // ultima volta che ci si è collegati al server, in
millisecondi
    boolean lastConnected = false;  // stato della connessione l'ultima volta attraverso il
ciclo principale
    const int postingInterval =7000; // definisce l'intervallo di tempo tra gli aggiornamenti
Pachube.com
// definizione per SHT11 - temperatura esterna
    #define dataPin 2
    #define clockPin 3
    SHT1x sht1x(dataPin, clockPin);

void setup()
{
// start della connessione ethernet
    Ethernet.begin(mac, ip);
// dare tempo al modulo ethernet di avviarsi:
    delay(300);
}

void loop()
{
// start della connessione ethernet
    Ethernet.begin(mac, ip);
// dare tempo al modulo ethernet di avviarsi:
    delay(300);
//rilevazione tempertura con correzione valore costante
    temp_c = (sht1x.readTemperatureC()+(-1));
    delay(300);
// Se non c'è connessione di rete, ma c'era l'ultima volta
// Attraverso il ciclo, quindi arresto il client:
    if (!client.connected() && lastConnected)
    {
        Serial.println();
        Serial.println("disconnecting.");
        client.stop();
    }
// Se non si è connessi, e "n" secondi sono passati da quando è avvenuta

```

```

// La tua ultima connessione, collegare di nuovo e inviare i dati:
if(!client.connected() && (millis() - lastConnectionTime > (postingInterval*1)))
{
    sendData(temp_c);
    delay(300);
}
//Memorizzare lo stato della connessione per la prossima volta che attraverso il ciclo
lastConnected = client.connected();

// for(a=1;a<10;a++){
//  delay(5000);}
}

//=====
// Questo metodo effettua una connessione HTTP al server:
void sendData(int thisData) {
// se c'è una connessione:
if (client.connect()) {
    Serial.print("Temperatura: ");
    Serial.println(thisData);
    Serial.println("connecting...");

// Invia la richiesta HTTP PUT.
// Inserisci qui il tuo codice Feed (es.57100) e l'ID relativo (es. 1)
    client.print("PUT /v2/feeds/57100/datastreams/1 HTTP/1.1\n");
    client.print("Host: api.pachube.com\n");
// Inserisci qui la tua Pachube API key
    client.print("X-PachubeApiKey:
DbpHB5lYctFpS478jEkwB5lYctFpSxRTWFMMeGN5amxQWT0g\n");
    client.print("Content-Length: ");
// Calcola la lunghezza della lettura del sensore in byte:
    int thisLength = getLength(thisData);
    client.println(thisLength, DEC);
// ultima parte della richiesta HTTP PUT
    client.print("Content-Type: text/csv\n");
    client.println("Connection: close\n");
// questo è il contenuto effettivo della richiesta PUT:
    client.println(thisData, DEC);
// rileva il tempo utilizzato per la connessione
    lastConnectionTime = millis();
//=====
// attivo la seriale per avere un controllo dei sensori
// lo scrivo solo quando invio i dati a Pachube
    Serial.begin(38400);
    humidity = sht1x.readHumidity();
    luce= analogRead(A3);

```

```

Serial.begin(38400);
Serial.print("temperatura esterna ");
Serial.println(int(temp_c)) ;
Serial.print("Umidita' esterna ");
Serial.println(humidity);
Serial.print("Percentuale Luce ");
Serial.print (int(luce/10.3));
Serial.println(" %");
//chiudo la seriale
Serial.end();
//tempo imposto tra due trasmissioni di dati
for (a=1; a<41; a++){
delay (9000);}
//=====
}
else
{
// se non posso instaurare la connessione:
Serial.println("connection failed");
}
}

/* Questo metodo calcola il numero di cifre del valore della lettura del Sensore.
Dal momento che ogni cifra del decimale ASCII è un byte, il numero di cifre
è pari al numero di byte:*/

int getLength(int someValue) {

// c'è almeno 1 byte
int digits = 1;

// Si divide sempre il valore per 10 aggiungendo 1 al conteggio delle cifre
// Si continua a dividere fino ad ottenere un risultato = 0

int dividend = someValue /10;
while (dividend > 0)
{
dividend = dividend /10;
digits++;
}
// restituisce il numero di digits:
return digits;
}

```