Nanode Collection Documentation

Release 0.1

Ville Säävuori

Sep 27, 2017

Contents

Documentation for my Nanode-related code. This code is available at https://bitbucket.org/Uninen/nanode-collection

CHAPTER 1

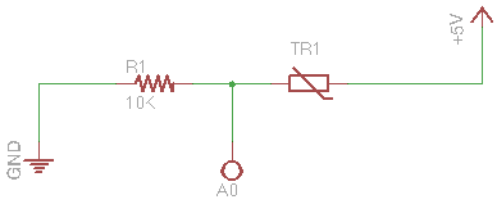
Contents

Uploading temperature to Pachube

This is a very simple sketch for updating one temperature measurement to Pachube from Nanode connected to Internet via ethernet. It uses the EtherCard library (as opposed to the no longer supported EtherSketch library). It's designed to be simple, expandable and readable. Even with comments the code is less than 100 lines.

See the live Pachube feed and the sketch code.

Schematic



The schematic is very simple. All you need is one 10K resistor and one two-pin 10K thermistor.

The code

The code is in three parts: setup, loop and custom functions. For now there is only one custom function for reading thermistor value.

Setup handles the basic initialization of board IP address, DNS server etc. Process then moves to loop where temperature data is read and then posted to Pachube feed. The posting is done with a low-level HTTP PUT query using *X-PachubeApiKey* header to authenticate the request. The loop runs forever, pausing after each upload for *POST_DELAY* milliseconds.

Most of the code is very simple and self explanatory. There are, however, a couple of things that are not obvious for beginner;

Converting thermistor value to temperature

A thermistor is a resistor whose resistance varies with temperature. The program the reads analogue resistance value from analog pin 0 and converts it to Kelvins using Steinhart-Hart equation. The conversion is done in *read_temperature* function.

Using a template for preparing the data

Pachube accepts multiple data formats for uploading new data points. This sketch uses CVS because it's the simplest when using only one variable. The data is prepared in a string template because it's very easy to modify.

For example, the same data could be sent in JSON format using following template (and changing the feed URL accordingly):

```
static char feed_template[] = "{\"version\":\"1.0.0\",\"datastreams\":[{\"id\":\
↪"temperature\", \"current_value\":\"%s\"}]}";
```

Converting double to string

Using *sprintf* to convert float or double to string does not work with Arduino language, but typically returns just '?'. Instead, we have to use dtostrf which works for floats and doubles. The resulting string can then be concatenated with the data template string.

Related reading

- http://www.arduino.cc/playground/ComponentLib/Thermistor2
- http://jeelabs.org/2011/06/19/ethercard-library-api/
- http://jeelabs.net/projects/cafe/wiki/EtherCard
- https://github.com/jcw/ethercard

Colophon

Following tools were used in making of this software:

- BitBucket for getting the code online.
- Eagle editor for drawing the schematics.
- Mercurial for version control.
- Pachube for storing data.
- Sphinx and ReadTheDocs.org for making this documentation.
- TextMate editor for the Arduino code.
- search