Weather23k

Index

1 Intention

This project “weather23k“ I developed since a couple of years.

I prefer running linux on my computers so I can‘t use “heavyweahter” that was included in the WS2300 package I bought. I needed to write my on application to get the weather data to my homepage.

I searched thru the internet and found the “open2300“ library what was the solution I needed.
I implemented a small application around it.

First I ran it on a PC but this consumed a lot of power when running 7/24. So I switched to a small Alix PC.
When the Raspberry Pi was announced I decided this would be the perfect solution and I got some of these.

But the compilation of “open2300” library was very slow on the RaspPi. Therefore I decided to develop a completely new application. It should use the basics of “open2300” but not include it‘s powefull gadgets. And it will be done for linux only!

So “weather23k“ has following features to keep it small and simple :

- read data out of a WS2300 compatible weather station
- push current data as text to a web server
- push data to a log file on a web server
- configurable via a config file
- no GUI
- linux only
- C coded
- no framework needed
- code as small as possible
- open source

2 Installation

2.1 Prerequisits

To get this program compiled and linked you need some programs and libraries on your computer. Sometimes these are preinstalled, sometime not. What is missing you can get with the package manager of your preferred linux system.

```
# pacman -Syu
# pacman -S gcc
# pacman -S make
# pacman -S libcurl libcurl-devel
```

Arch Linux

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get install make
$ sudo apt-get install gcc
$ sudo apt-get libcurl libcurl-devel
```

Debian Linux

2.2 Get the program

The programm itself is available as source code only. You have to get it from GitHub by cloning it :

```
git clone https://github.com/KlabautermannSW/weather23k.git
```

or download the source as a .zip file :

```
wget https://github.com/KlabautermannSW/weather23k/archive/master.zip
```

Both will result in a directory named `weather23k` with some subdirectories.
In this directory run

```
make
```

This will start compiling and linking all necessary files together and results in a binary in the subdirectory `bin`.

3 Hardware

I use this program on a Raspberry Pi 1 Model B.
The weatherstation WS2307 ist connected to the Rasbpi via an USB to serial converter.
The connection to the internet is done via cable bound ethernet.

For more information visit the internet pages :
http://www.ur9.de/infos.shtml and
http://www.ur9.de/ws2300.shtml

4 Configuration file

The configuration file has several sections that are described in the next paragrafs.
A line beginning with the “#” character is interpreted as a single line comment.

4.1 [FTP]

This describes the hadling and use of the ftp server.

```
server =
user =
key =
file =
logpath =
```

All data are encoded using the encode fucntion.

4.1.1 server

This is the server string as used for the ftp server connection.

4.1.2 user

The ftp account’s name.

4.1.3 key

The ftp account’s password

4.1.4 file

The text file to be included into the web page.

4.1.5 logpath

The path where the daily logfiles will be stored on the ftp server.

4.2 [File]

```
# if no logpath is given log will saved in the current directory
# logpath =
```

4.2.1 logpath

The path where the daily logfiles will be stored on the local computer. Default is the current directoty.

4.3 [Port]

```
port = /dev/ttyUSB0
# port = /dev/ttyAMA0
```

4.3.1 port

The port at which the weather station is connected. The second line is the first serial port of a Raspberry Pi.

4.4 [Template]

The template key has to be the last key!

All text following this key will be put character by character into the file “data.txt” on the ftp server. The only exception is the string "<*var=_name_*>" wherein _name_ is described in the following list. This string will be changed to the value of the given variable or to an empty string if _name_ is unknown.

- temp temperature
- press relative air pressure
- hum relative air humidity
- winddir wind direction
- speed_m wind speed [m/sec]
- speed_kmh wind speed [kmh]
- speed_kn wind speed [kn]
- speed_bf wind speed [bft]
- dew dewpoint temperature
- chill windchill temperature
- rph rain per hour
- rpd rain per day
- dirstr wind direction as text
- time current time stamp

```
<table border="0" cellspacing="0" cellpadding="0">
      <tr>
            <td width="200">Messzeit</td>
            <td width="120"><*var=time*> Uhr</td>
      </tr>
      <tr>
            <td>Temperatur</td>
            <td><*var=temp*> &#176;C</td>
      </tr>
      <tr>
            <td>relativer Luftdruck</td>
            <td><*var=press*> hPa</td>
      </tr>
      <tr>
            <td>Luftfeuchtigkeit</td>
            <td><*var=hum*> &#37;</td>
      </tr>
      <tr>
            <td>Windrichtung</td>
            <td><*var=winddir*> &#176;</td>
            <td><*var=dirstr*></td>
      </tr>
      <tr>
            <td>Windgeschwindigkeit</td>
            <td width="100"><*var=speed_m*> m/sec</td>
            <td width="90"><*var=speed_kmh*> km/h</td>
            <td width="90"><*var=speed_kn*> kn</td>
            <td width="90"><*var=speed_bf*> bft</td>
      </tr>
      <tr>
            <td>Taupunkt</td>
            <td><*var=dew*> &#176;C</td>
      </tr>
      <tr>
            <td>gef&uuml;hlte Temperatur</td>
            <td><*var=chill*> &#176;C</td>
      </tr>
```

```
      <tr>
            <td>Regen</td>
            <td width="80"><*var=rph*> l/h</td>
            <td width="100"><*var=rpd*> l/24h</td>
      </tr>
</table>
```

Template example

5 PHP files

There are several php scripts included as samples how to visualize the data from the log files.
The results can be seen on
http://www.ur9.de/graphs.shtml