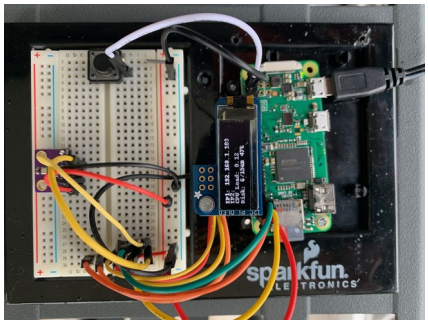
A cool-retro-weatherstation

Making a cool-retro-weatherstation with the PiDP-11

If you want to make one yourself, proceed as follows:

Using the other tutorials in my github repo https://github.com/rricharz/pidp11-2.11bsd, do the following:

- add a user in 2.11 BSD
- set date and time automatically during bootup (to be used to save weather data)
- install cool-retro-term
- prepare a remote pizerow for the weather sensor (see “using rsh to run an command”)
- install a BME280 weather sensor on the remote pizerow, see https://www.raspberrypi-spy.co.uk/2016/07/using-bme280-i2c-temperature-pressure-sensor-in-python/



Picture of my breadboard and final versions of the remote sensor. The weather sensor is the small board at the left of the breadboard and lower right of the final version. Most cables are for a little diagnostic OLED display, which is not required. It uses the same I2C wires at a different address. It could also display the weather data directly on the sensor. I also added a switch to shut down the pizerow gracefully.

- A copy of the python script to read the sensor data is in this repository under weather/sensor

- Make sure that this script works properly on the remote pizerow and put a copy into /home/pi/bin on the remote pizerow
- Log into 2.11BSD using cool-retro-term as the newly created user

```
mkdir weather
cd weather
```

- Use any available means (such as for example FileZilla), to copy the files from weather/BSD into the new weather directory
- Use the “make” command to compile the C programs in your weather directory
- Execute the newly copied command weather in 2.11BSD

weather extract T,P and H from the data received from the remote sensor and displays it once:

```
rene@pdp11:weather$ weather
Contacting remote sensor...
Data received, analyzing ...
**************************************
The sensor reports the following data:
 Temperature  7.0 C
 Pressure  969.8 hPa
 Humidity     66.8 %
**************************************
rene@pdp11:weather$
```

If weather works on your system, make a directory data in ~/weather, where weather2 can store the data received:

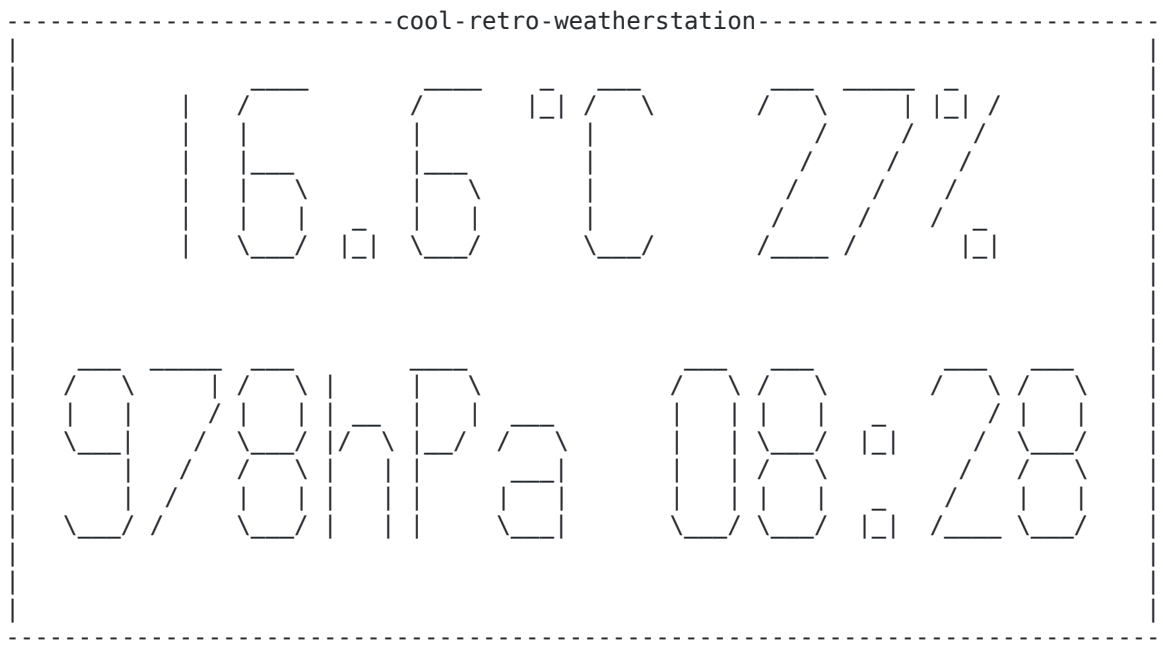
```
cd
cd weather
mkdir data
weather2
```

weather2 will make each day a new file in ~/weather/data with the date as the name of the file, and will append every minute one line in that file with the time, T, P and H.

It will loop indefinitely and update the screen once per minute with a cool-retro display:

```
rene@pdp11:weather$ weather2
Contacting remote sensor...
Data received, analyzing ...
The sensor reports the following data:
 Temperature  6.3 C
 Pressure     970.0 hPa
 Humidity     72 %
Storing data in /home/rene/weather/data/20190308
===========================cool-retro-weatherstation===========================
|             ____        ___     _   ___        _____        _               |
|            /          /   \   |_| /   \           |   |   |_| /            |
|            |               |       |               /   |      /             |
|            |___         ___|       |              /    |     /              |
|            |   \           |       |             /     |    /               |
|            |   |           |       |            /      |   /  _             |
|            \___/   *   \___/       \___/       /       |     |_|            |
|                                                                             |
===============================================================================
Sleeping 60 seconds, type <ctrl>c to abort
```

weather3 works like weather2, but displays also pressure and time. It makes use of the library curses to handle the positioning of characters on the screen.



What you could do yourself:

- Analyze and display current and past weather data
- Make a simple forecast based on pressure changes

Good luck!