



[Supports](#) [Features](#) [Download](#) [About WMR\(RU\)](#) [Usage\(RU\)](#) [Controlling 4 Line\(RU\)](#)

Supports

Oregon Scientific LW301
Oregon Scientific WMR88
Oregon Scientific WMR100
Oregon Scientific WMR200
Oregon Scientific WMRS200
Oregon Scientific RMS300A
Oregon Scientific RMS600
Oregon Scientific I300
Oregon Scientific I600

Requisites

Developer lib:

- [libhid] package installed.
- [libusb 0.x] package installed.
- [libsqlite3] package installed.
- [rrdtool] needed to weather grapher.

Features

*SQL sqlite3 database support for store data from weather station. *Store data from weather station to log file in txt format. *Correct all data from sensor (temperature, pressure, rain, wind). *Build graphical statistic from sql data (rrd-tools). *Posted to popularity weather web site weather data.

Download

Latest tarball source found in: <http://www.nkl.ru/support/wmr/wmr-oregon.tar.gz>

Introduction

The WMR/RMS/I series from Oregon Scientific, use a proprietary protocol for their USB connection. It's very useful for enthusiasts running a headless Linux box to collect and analyze data from this link, but unfortunately the protocol isn't openly documented, nor are clients provided for this platform.

This C program handles the USB protocol of the WMRxxx, or RMSxxx, or Ixxx, and translates it an ASCII/SQLite3/RRD format, easy for parsing/analysing. See directory scrips/ for example.

You'll need to setup the udev rules (see udev/README).

This is due to how libhid accesses the USB ports.

SQLite

First run script/wmr_create_db_sqlite3.sh, this create sqlite3 DB structure in /var/weather/weather.db See structure sql database: contrib/wmr.sql

Alarm event

Edit section alarm in [config file](#). For event alarming, wmr-oregon runing script. Default: /usr/bin/wmr_alarm.sh - see script/wmr_alarm.sh & edit for you job. - see script/controlling/C2000-CP1.php for controlling 4 line from [C2000-CP1 Bolid\(C\)trademark](#)

```
Format command line send to wmr_alarm.sh:
/path/to/script/wmr_alarm.sh 'TYPE SENSOR' 'NUM SENSOR' 'VALUE SENSOR' 'TYPE ALARM MIN/MAX'
sample:
/path/to/script/wmr_alarm.sh TEMP 1 -22 MIN
/path/to/script/wmr_alarm.sh PRESSURE 0 778 MAX
/path/to/script/wmr_alarm.sh HUMIDITY 3 12 MIN
... etc
If sensor type = BATTERY, format command:
/path/to/script/wmr_alarm.sh BATTERY 'NUM SENSOR' 'VALUE SENSOR' 'TYPE SENSOR'
sample:
/path/to/script/wmr_alarm.sh BATTERY 1 7 TEMP
/path/to/script/wmr_alarm.sh BATTERY 0 11 MAIN
Valid type of sensor:
All data sensor: "TEMP", "HUMIDITY", "PRESSURE", "WIND", "RAIN", "UV"
Battery check sensor: "MAIN", "WATER", "TEMP", "WIND", "RAIN", "UV"
```

Compile

Before install or compile development library: libhid, libusb 0.x, libsqlite3 After, check path to development library in Makefile, and run make this project:

```
make clean ; make all ; make install
vi /etc/wmr.conf
```

```
lsmod | grep usbhid
/usr/bin/wmr-oregon -d -s -c /etc/wmr.conf
```

Use

To run:

```
/usr/bin/wmr-oregon -c /etc/wmr.conf
```

flag -c /path/name.ext = path and name to config file [/etc/wmr.conf](#).

default [./wmr.conf](#), or:

```
/usr/bin/wmr-oregon -d -c /etc/wmr.conf
```

flag -d = start as a daemon server mode. (logging only syslog)

flag -s automatic enable.

```
/usr/bin/wmr-oregon -s -c /etc/wmr.conf
```

flag -s = print all info & debug info to local syslog server.

```
/usr/bin/wmr-oregon -v
```

flag -v = print all config token from config file,

this options require flag -c <path/to/config/name.ext> before.

To autostart:

```
cp contrib/wmr.init /etc/rc.d/init.d/
ln -s /etc/rc.d/init.d/wmr.init /etc/rc.d/rc3.d/S95wmr
ln -s /etc/rc.d/init.d/wmr.init /etc/rc.d/rc5.d/S95wmr
```

To reread config file, send:

```
kill -HUP `pidof wmr-oregon`
```

To reread config file and re init USB device, send:

```
kill -USR1 `pidof wmr-oregon`
```

To logrotate data file - log/sql/rrd, send:

```
kill -USR2 `pidof wmr-oregon`
```

- run & see script/wmr_logrotate.sh

To stop wmr-oregon in daemon mode, send:

```
kill -TERM `pidof wmr-oregon`
```

To update internet weather hosting, support:

```
www.pwsweather.com,
www.pywws.com,
www.wunderground.com
```

- see script/update_weather/README, .sh

Logrotate data LOG/SQL/RRD

Copy contrib/wmr_logrotate.sh to /usr/bin/wmr_logrotate.sh , or edit config file path to logrotae script - wmr_logrotate.sh

Run - type: /usr/bin/wmr_logrotate.sh -logrotate or, insert cron job new line:

for every day:

```
0 0 * * * /usr/bin/wmr_logrotate.sh -logrotate
```

for every month:

```
0 0 1 * * /usr/bin/wmr_logrotate.sh -logrotate
```

for every year:

```
0 0 1 1 * /usr/bin/wmr_logrotate.sh -logrotate
```