# Waonc Quick Reference Manual

0.12

Generated by Doxygen 1.8.17

# 1 Waonc 0.12

**Date**

2013-10-27 to 2022-05-31

Waonc is a minor cleanup and documentation of the waon application suite.

The `libwaonc` directory contains routines for analysis and processing. The `waonc` directory contains the main program and some small sample/test MIDI files. The `pvc` directory contains code for the *Waon* phase vocoder. The `waonc` directory contains a *Gtk* wrapper for the *Waon* converter. All source code is in the *C* language.

The command `waonc -help` provides a good listing of the program options.

# 2 Waonc 0.12

**Date**

2015-03-28 to 2022-05-31

This section project *Kengo*'s tips on using *Waon*.

## 2.1 Inside Waonc

Intro! This description is *Kengo*'s, with additional material.

### 2.1.1 Policy of Event Selections

**2.1.1.1 Policy of On/Off Event Selections** Selection of sound in frequency domain is relatively straightforward, but there seems to be no general way of doing that in the time domain. It is obvious that this determines the quality of this kind of software. If you have some idea, please let me (Kengo) know!

At present, I'm trying two ways to detect on/off events:

1. Detection of over-threshold (no option -k/–peak), where a note-on event is detected when the power is larger than the on-threshold value hardwired in the source code.
2. Peak detection (option -k/–peak).

A. Off-event:

An off-event is generated if (i_lsts[i] <= off_threshold), where off_threshold = 0 as defined in the source code. Each 'i_lsts[]' value is velocity hopefully scaled in the range [0, 127].

```
         velocity ^      ..
                2 |.     ..
                1 |..   ...
off_threshold ->0 +--*------> time
                     ^
                     |
                      --- the position where the off-event is placed
```

B. On-event:

B.1. If this note is off at last step, and if (i_lsts[i] > on_threshold), then an on-event is generated:

```
         velocity ^
               10 |.      .
                9 |.     *.
on_threshold -> 8 |.     ..
                  ---------
                2 |.     ..
                1 |..   ...
off_threshold ->0 +--*------> time
                     ^
                     |
                      --- the position where on-event is placed
```

B.2. If this note is on at last step:

B.2.1. With and without -k option, and if (i_lsts[i] > (*on_lst[i])), where *on_lst[i] is the velocity of this note, then the velocity is overwritten by i_lsts[i].

```
         velocity ^
               11 |.
               10 |.      *.
                9 |.     *..
on_threshold -> 8 |.     ...
                  ----------
                2 |.     ...
                1 |..   ....
off_threshold ->0 +--*---*---> time
                     ^
                     |
                      --- velocity is overwritten by larger values here
```

B.2.2. Only with -k option, if (i_lsts[i] >= (∗on_lst[i] + peak_threshold)) where peak_threshold is given by -k (–peak) option.

```
           velocity ^
                 36 |            *.
                 35 |            ...
 peak_threshold-->34 |            ...
      \          33 |            ...
        \           ----------------
          \      11 |.           ...
           \---->10 |.      *..  ...
                  9 |.     *...  ...
   on_threshold -> 8 |.     .........
                     ----------------
                  2 |.     .........
                  1 |..   ..........
   off_threshold ->0 +--*---*---------> time
                                  ^
                                  |
                                   --- off- & on-event is placed here
```

The resultant MIDI signal depends on the -k/–peak option:

Without the -k/–peak option?

```
|--    O<36>----
|  X
+--*---*-----*---> time
```

With the -k/–peak option:

```
|--    O<10>-O<36>-
|  X         X
+--*---*-----*---> time
```

where

```
O : note-on event
X : note-off event
```

#### 2.1.1.2 Policy of Note Event Selections Policy of Selection of Note (Frequency)

- Without –patch option:
  1. Search for the maximum in the power spectrum and get the frequency.
  2. Set 0 into the power spectrum around the maximum. This area subtracted is up to the frequency of the local minimum of power in both sides.
  3. Search again until there is no more peak in power spectrum.
- With –patch option:
  1. Search for the maximum in the power spectrum and get the frequency.
  2. Subtract the power of the –patch option, scaled to the maximum frequency.
  3. Search again until there is no peak in power spectrum.

# 3 Licenses, Waonc project.

**Library: waonc_suite and its libraries, applications, and documents**

**Author**

Chris Ahlstrom

**Version**

0.12

**Date**

2013-11-13 to 2022-05-31

**License: $XPC_SUITE_GPL_LICENSE$**

Included in the Waonc project are notes and code from the Waon project:

```
Copyright (C) 1998,1999 Kengo Ichiki <kichiki@users.sourceforge.net>
```

## 3.1 License Terms for the waonc project.

Wherever the tag $XPC_SUITE_GPL_LICENSE$ appears, substitute the appropriate license text, depending on whether the project is a library, application, documentation, or server software. These licenses apply to each sub-project and file artifact.

## 3.2 Application License

The application license is the **GNU GPLv3**. http://www.gnu.org/licenses/gpl-3.0.txt

Copyright (C) 2008-2022 by Chris Ahlstrom

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the

```
Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1301, USA.
```

## 3.3 Library License

The library license is the **GNU LGPLv3**. http://www.gnu.org/licenses/lgpl-3.0.txt



This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Lesser Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the