ZZC

Claude Heiland-Allen

2010–2018

Contents

1 COPYING

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that

Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or

similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users

beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any

tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option

remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under

this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for

sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is

in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

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, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

  If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

    <program>  Copyright (C) <year>  <name of author>
    This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
    This is free software, and you are welcome to redistribute it
    under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License.  Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

  You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

  The GNU General Public License does not permit incorporating your program
into proprietary programs.  If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library.  If this is what you want to do, use the GNU Lesser General
Public License instead of this License.  But first, please read
<http://www.gnu.org/philosophy/why-not-lgpl.html>.
```

2 README

```
z->z^2+c : Mandelbrot Music
==========================

Quick Start
-----------

Click inside the Mandelbrot image.
See what potential periods are displayed above the 'periodoscope'.
Click 'reset' (green bang) at the top of the 'periodoscope'.
Click factors (green bangs) until it looks evenly coloured to the left.
Click 'ok' (green bang) to add a factor to the list.
Repeat the above two steps until only '+1' remains.
Click the audio bang.
Listen to the rhythmic interplays, 'record' (green bang) if you like.
Explore the periodic attractors of points inside the Mandelbrot Set.


Mandelbrot Controls
-------------------
```

'Left-click' to centre and zoom in.
'Right-click' to centre and zoom out.
'Up' to zoom in on the current target.
'Down' to zoom out on the current target.
'Space' to set a new target without zooming.
'Return' to centre on the target 'space' target.
'Home' to return to base.

Principles Behind The 'Periodoscope'

Iterating the equation z->z^2+c for points inside the Mandelbrot Set tends towards a periodic attractor. The periodic attractor has a structure determined by the path through the Mandelbrot Set from 'mu-atom' (solid region) to 'mu-atom'. Each 'mu-atom' has a period of a certain length, and 'children' of a 'mu-atom' have periods related to the 'parent'. The 'periodoscope' visualizes the positions of the iterates of the equation, and you should be able to see they cluster together. By chosing factors of the period (green bangs) you can classify the points into clusters, recursively: clusters have sub-clusters and so on.

Principles Behind The 'Sonification'

The structure of the periodic attractors lends itself to rhythm - each level of clustering adds another voice, and the sound produced by each voice is determined by the location of the iterates relative to this cluster structure. Each cluster is normalized and converted to polar form, and the resulting angle and distance of the points in that cluster determine parameters of the sound.

Audio Generation Algorithms

There is some dynamic patching to create each 'voice', and each voice has a number of 'subvoices' to give a smoother sound (so each voice is somewhat polyphonic). The main sound generation is using 'vcf~' triggered into ringing with a tiny click (a higher-precision 'vcf~' using 'double' precision (64bit) floating point internally is also provided with a Makefile for GNU/Linux, this improves the sound particularly for long decay times). Then a trick from Miller Puckette's 'Patch For Guitar' (aka 'Smeck') converts the near-sinusoid output of 'vcf~' into phase and amplitude information, which can be manipulated separately (phase can be waveshaped, rate of decay can be altered too).

To add new sonification presets, create or edit a new 'N_voice~.pd' and 'N_osc~.pd' (where N is a number) and add the new option to the presets patch. 'N_voice~.pd' maps from (sub)cluster position to sound control parameters, and the 'N_osc~.pd' generates the audio itself.

Advanced Usage

```
You can set the 'maximum' period that the analysis algorithm will check
for at the top left, but longer periods take more time - O(n^2). So you
can disable the calculation with the orange toggle to the left. The
'settle' control is useful if you get spiral patterns in the
'periodoscope', this is an indication that you should increase the
number of iterations before starting to calculate the period - points
near the boundary of the Mandelbrot Set take a long time to get close to
their periodic attractor.


Legal
-----

z->z^2+c -- Mandelbrot Music
Copyright (C) 2010,2018  Claude Heiland-Allen <claude@mathr.co.uk>

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, see <http://www.gnu.org/licenses/>.
```

3 zzc/1_osc~.pd

```
#N canvas 369 115 450 557 10;
#X obj 141 165 vline~;
#X obj 163 122 vline~;
#X obj 206 320 *~;
#X obj 192 285 vline~;
#X obj 237 285 vline~;
#X obj 138 321 *~;
#X obj 141 207 vcf~ 100;
#X obj 140 101 unpack f f f f f;
#X obj 139 9 inlet;
#X obj 295 346 outlet;
#X msg 141 142 \$1 0 5 \, 0 1 5;
#X obj 140 31 route \$2;
#X obj 17 293 env~ 16384;
#X obj 17 314 > 0;
#X obj 17 335 sel 0;
#X obj 17 356 f 0;
#X obj 45 513 switch~;
#X obj 108 350 bang;
#X obj 108 401 f 1;
#X obj 17 424 spigot 1;
#X obj 77 399 f 0;
#X obj 49 370 delay 10;
#X obj 48 397 f 1;
#X obj 138 345 throw~ \$1-l;
```

```
#X obj 206 345 throw~ \$1-r;
#X msg 229 204 0 5 \, \$1 15 15;
#X msg 209 224 0 5 \, \$1 15 15;
#X obj 140 233 *~;
#X obj 138 290 hip~ 10;
#X obj 138 271 *~;
#X obj 137 251 expr~ tanh($v1)*2;
#X connect 0 0 6 0;
#X connect 1 0 6 1;
#X connect 2 0 24 0;
#X connect 3 0 5 1;
#X connect 4 0 2 1;
#X connect 5 0 23 0;
#X connect 6 0 27 0;
#X connect 6 0 27 1;
#X connect 7 0 10 0;
#X connect 7 1 1 0;
#X connect 7 2 6 2;
#X connect 7 3 26 0;
#X connect 7 4 25 0;
#X connect 8 0 11 0;
#X connect 10 0 0 0;
#X connect 11 0 7 0;
#X connect 11 0 17 0;
#X connect 11 1 9 0;
#X connect 12 0 13 0;
#X connect 13 0 14 0;
#X connect 14 0 15 0;
#X connect 15 0 19 0;
#X connect 17 0 18 0;
#X connect 17 0 20 0;
#X connect 17 0 21 0;
#X connect 18 0 16 0;
#X connect 19 0 16 0;
#X connect 20 0 19 1;
#X connect 21 0 22 0;
#X connect 22 0 19 1;
#X connect 25 0 4 0;
#X connect 26 0 3 0;
#X connect 27 0 30 0;
#X connect 28 0 5 0;
#X connect 28 0 2 0;
#X connect 28 0 12 0;
#X connect 29 0 28 0;
#X connect 30 0 29 0;
#X connect 30 0 29 1;
```

4 zzc/1_voice~.pd

```
#N canvas 0 0 514 697 10;
#X obj 243 603 throw~ \$2;
#X obj 311 603 throw~ \$3;
#X obj 40 14 r \$1;
#X obj 68 78 unpack f f;
#X obj 180 248 mtof;
#X obj 228 18 loadbang;
#X obj 229 74 f \$4;
```

```
#X obj 220 329 t f f;
#X obj 220 360 cos;
#X obj 247 360 sin;
#X obj 221 309 * 3.14159;
#X obj 221 261 wrap;
#X obj 221 282 * 0.5;
#X obj 112 134 + 1;
#X obj 140 203 /;
#X obj 103 204 *;
#X obj 222 213 / 3000;
#X obj 112 159 + 0.3;
#X obj 103 247 /;
#X obj 221 236 * 4000;
#X obj 40 34 route \$4;
#X obj 57 149 t f f;
#X obj 57 170 *;
#X obj 57 191 t f f;
#X obj 57 212 *;
#X obj 56 126 t f f;
#X obj 57 251 *;
#X obj 93 394 pack f f f f f f;
#X obj 39 53 t b a;
#X obj 34 359 f 0;
#X obj 69 341 + 1;
#X obj 130 248 sqrt;
#X obj 209 56 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144 -1
-1;
#X obj 102 225 * 100;
#X obj 247 428 catch~ \$0-l;
#X obj 322 428 catch~ \$0-r;
#X obj 65 423 1_osc~ \$0 0;
#X obj 206 659 delwrite~ \$0-d-l 100;
#X obj 335 658 delwrite~ \$0-d-r 100;
#X obj 206 503 delread~ \$0-d-r 10;
#X obj 340 502 delread~ \$0-d-l 10;
#X obj 207 521 *~ -0.975;
#X obj 301 232 mtof;
#X obj 301 274 /;
#X obj 345 564 hip~ 400;
#X obj 205 565 hip~ 400;
#X obj 321 314 expr exp(-$f1)-1;
#X obj 347 387 * -1;
#X obj 340 523 *~ 0.975;
#X obj 205 585 expr~ tanh($v1);
#X obj 345 584 expr~ tanh($v1);
#X obj 333 626 hip~ 1000;
#X obj 220 629 hip~ 1000;
#X obj 246 452 expr~ sin($v1);
#X obj 321 475 expr~ sin($v1);
#X obj 122 183 * 500;
#X obj 316 584 *~ 2;
#X obj 260 564 *~ 2;
#X obj 65 443 1_osc~ \$0 1;
#X obj 65 463 1_osc~ \$0 2;
#X obj 65 483 1_osc~ \$0 3;
#X obj 65 503 1_osc~ \$0 4;
#X obj 65 523 1_osc~ \$0 5;
```

```
#X obj 65 543 1_osc~ \$0 6;
#X obj 65 563 1_osc~ \$0 7;
#X obj 65 583 1_osc~ \$0 8;
#X obj 65 603 1_osc~ \$0 9;
#X obj 65 623 1_osc~ \$0 10;
#X obj 65 643 1_osc~ \$0 11;
#X obj 65 663 1_osc~ \$0 12;
#X obj 65 683 1_osc~ \$0 13;
#X obj 65 703 1_osc~ \$0 14;
#X obj 65 723 1_osc~ \$0 15;
#X obj 301 253 swap 666;
#X obj 345 543 lop~ 4000;
#X obj 205 544 lop~ 4000;
#X obj 289 122 table \$0-chord 16;
#X obj 290 98 s \$0-chord;
#X msg 289 75 0 24 36 43 48 51 55 60 63 67 70 72;
#X obj 245 42 t b b;
#X obj 232 160 tabread \$0-chord;
#X obj 69 362 mod 16;
#X connect 2 0 20 0;
#X connect 3 0 25 0;
#X connect 3 1 16 0;
#X connect 4 0 14 1;
#X connect 4 0 31 0;
#X connect 4 0 27 2;
#X connect 5 0 79 0;
#X connect 6 0 80 0;
#X connect 7 0 8 0;
#X connect 7 1 9 0;
#X connect 8 0 27 4;
#X connect 9 0 27 5;
#X connect 10 0 7 0;
#X connect 11 0 12 0;
#X connect 12 0 10 0;
#X connect 13 0 17 0;
#X connect 14 0 15 1;
#X connect 14 0 27 3;
#X connect 15 0 33 0;
#X connect 16 0 19 0;
#X connect 17 0 15 0;
#X connect 17 0 55 0;
#X connect 18 0 26 1;
#X connect 19 0 11 0;
#X connect 20 0 28 0;
#X connect 21 0 22 0;
#X connect 21 1 22 1;
#X connect 22 0 23 0;
#X connect 23 0 24 0;
#X connect 23 1 24 1;
#X connect 24 0 26 0;
#X connect 25 0 21 0;
#X connect 25 1 13 0;
#X connect 26 0 27 1;
#X connect 27 0 36 0;
#X connect 28 0 29 0;
#X connect 28 1 3 0;
#X connect 29 0 30 0;
```

```
#X connect 29 0 27 0;
#X connect 30 0 81 0;
#X connect 31 0 18 1;
#X connect 32 0 79 0;
#X connect 33 0 18 0;
#X connect 34 0 53 0;
#X connect 35 0 54 0;
#X connect 36 0 58 0;
#X connect 39 0 41 0;
#X connect 40 0 48 0;
#X connect 41 0 57 0;
#X connect 41 0 75 0;
#X connect 42 0 73 0;
#X connect 43 0 40 0;
#X connect 43 0 39 0;
#X connect 43 0 46 0;
#X connect 44 0 50 0;
#X connect 45 0 49 0;
#X connect 46 0 41 1;
#X connect 46 0 47 0;
#X connect 47 0 48 1;
#X connect 48 0 56 0;
#X connect 48 0 74 0;
#X connect 49 0 37 0;
#X connect 50 0 38 0;
#X connect 51 0 38 0;
#X connect 52 0 37 0;
#X connect 53 0 52 0;
#X connect 53 0 0 0;
#X connect 54 0 51 0;
#X connect 54 0 1 0;
#X connect 55 0 14 0;
#X connect 56 0 1 0;
#X connect 57 0 0 0;
#X connect 58 0 59 0;
#X connect 59 0 60 0;
#X connect 60 0 61 0;
#X connect 61 0 62 0;
#X connect 62 0 63 0;
#X connect 63 0 64 0;
#X connect 64 0 65 0;
#X connect 65 0 66 0;
#X connect 66 0 67 0;
#X connect 67 0 68 0;
#X connect 68 0 69 0;
#X connect 69 0 70 0;
#X connect 70 0 71 0;
#X connect 71 0 72 0;
#X connect 73 0 43 0;
#X connect 73 1 43 1;
#X connect 74 0 44 0;
#X connect 75 0 45 0;
#X connect 78 0 77 0;
#X connect 79 0 6 0;
#X connect 79 1 78 0;
#X connect 80 0 4 0;
#X connect 80 0 42 0;
```

```
#X connect 81 0 29 1;
```

5 zzc/2_osc~.pd

```
#N canvas 369 115 450 557 10;
#X obj 141 141 vline~;
#X obj 163 98 vline~;
#X obj 207 463 *~;
#X obj 288 421 vline~;
#X obj 333 421 vline~;
#X obj 139 464 *~;
#X obj 141 172 vcf~ 100;
#X obj 140 77 unpack f f f f f;
#X obj 139 9 inlet;
#X obj 314 488 outlet;
#X msg 325 370 0 5 \, \$1 5 5;
#X msg 305 390 0 5 \, \$1 5 5;
#X msg 141 118 \$1 0 5 \, 0 1 5;
#X obj 141 199 hilbert~;
#X obj 140 264 *~ 512;
#X obj 140 285 +~ 1;
#X obj 16 96 table \$0-saw 515;
#X obj 16 74 send \$0-saw;
#X msg 17 51 sinesum 512 1 0.5 0.25 0.125 0.0625 0.03125 \, normalize
1;
#X obj 19 28 loadbang;
#X obj 138 309 tabread4~ \$0-saw;
#X obj 137 338 *~;
#X obj 141 229 expr~ atan2($v2 \, $v1)/(2.0*3.1415926)+0.5 \; cbrt($v1*$v1+$v2*$v2)
;
#X obj 5 302 env~ 16384;
#X obj 5 323 > 0;
#X obj 5 344 sel 0;
#X obj 5 365 f 0;
#X obj 77 464 switch~;
#X obj 96 359 bang;
#X obj 96 410 f 1;
#X obj 5 433 spigot 1;
#X obj 65 408 f 0;
#X obj 37 379 delay 10;
#X obj 36 406 f 1;
#X obj 139 486 throw~ \$1-l;
#X obj 207 486 throw~ \$1-r;
#X obj 139 32 route \$2;
#X connect 0 0 6 0;
#X connect 1 0 6 1;
#X connect 2 0 35 0;
#X connect 3 0 5 1;
#X connect 4 0 2 1;
#X connect 5 0 34 0;
#X connect 6 0 13 0;
#X connect 7 0 12 0;
#X connect 7 1 1 0;
#X connect 7 2 6 2;
#X connect 7 3 11 0;
#X connect 7 4 10 0;
```

```
#X connect 8 0 36 0;
#X connect 10 0 4 0;
#X connect 11 0 3 0;
#X connect 12 0 0 0;
#X connect 13 0 22 0;
#X connect 13 1 22 1;
#X connect 14 0 15 0;
#X connect 15 0 20 0;
#X connect 18 0 17 0;
#X connect 19 0 18 0;
#X connect 20 0 21 0;
#X connect 21 0 5 0;
#X connect 21 0 2 0;
#X connect 21 0 23 0;
#X connect 22 0 14 0;
#X connect 22 1 21 1;
#X connect 23 0 24 0;
#X connect 24 0 25 0;
#X connect 25 0 26 0;
#X connect 26 0 30 0;
#X connect 28 0 29 0;
#X connect 28 0 31 0;
#X connect 28 0 32 0;
#X connect 29 0 27 0;
#X connect 30 0 27 0;
#X connect 31 0 30 1;
#X connect 32 0 33 0;
#X connect 33 0 30 1;
#X connect 36 0 7 0;
#X connect 36 0 28 0;
#X connect 36 1 9 0;
```

6 zzc/2_voice~.pd

```
#N canvas 0 0 532 639 10;
#X obj 191 585 throw~ \$2;
#X obj 311 587 throw~ \$3;
#X obj 40 14 r \$1;
#X obj 68 78 unpack f f;
#X obj 171 265 mtof;
#X obj 229 20 loadbang;
#X obj 229 74 f \$4;
#X obj 216 299 t f f;
#X obj 216 330 cos;
#X obj 243 330 sin;
#X obj 217 279 * 3.14159;
#X obj 217 231 wrap;
#X obj 217 252 * 0.5;
#X obj 112 134 + 1;
#X obj 140 203 /;
#X obj 103 204 *;
#X obj 187 138 / 3000;
#X obj 112 159 + 0.3;
#X obj 103 247 /;
#X obj 217 206 * 4000;
#X obj 171 237 +;
#X obj 40 34 route \$4;
```

```
#X obj 57 149 t f f;
#X obj 57 170 *;
#X obj 57 191 t f f;
#X obj 57 212 *;
#X obj 56 126 t f f;
#X obj 57 251 *;
#X obj 93 394 pack f f f f f f;
#X obj 39 53 t b a;
#X obj 34 359 f 0;
#X obj 69 341 + 1;
#X obj 130 248 sqrt;
#X obj 209 56 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144 -1
-1;
#X obj 142 180 * 10000;
#X obj 102 225 * 100;
#X obj 111 363 *;
#X obj 126 290 + 1;
#X obj 151 363 *;
#X obj 126 314 t f f;
#X obj 128 335 *;
#X obj 171 207 / 48;
#X obj 310 564 expr~ sin($v1/3)/4;
#X obj 191 564 expr~ sin($v1/3)/4;
#X obj 190 541 catch~ \$0-l;
#X obj 310 539 catch~ \$0-r;
#X obj 65 423 2_osc~ \$0 0;
#X obj 309 131 table \$0-chord 16;
#X obj 310 107 s \$0-chord;
#X msg 309 84 0 24 36 43 48 51 55 60 63 67 70 72;
#X obj 230 45 t b b;
#X obj 232 160 tabread \$0-chord;
#X obj 65 443 2_osc~ \$0 1;
#X obj 65 463 2_osc~ \$0 2;
#X obj 65 483 2_osc~ \$0 3;
#X obj 65 503 2_osc~ \$0 4;
#X obj 65 523 2_osc~ \$0 5;
#X obj 65 543 2_osc~ \$0 6;
#X obj 65 563 2_osc~ \$0 7;
#X obj 65 583 2_osc~ \$0 8;
#X obj 65 603 2_osc~ \$0 9;
#X obj 65 623 2_osc~ \$0 10;
#X obj 65 643 2_osc~ \$0 11;
#X obj 65 663 2_osc~ \$0 12;
#X obj 65 683 2_osc~ \$0 13;
#X obj 65 703 2_osc~ \$0 14;
#X obj 65 723 2_osc~ \$0 15;
#X obj 69 362 mod 8;
#X connect 2 0 21 0;
#X connect 3 0 26 0;
#X connect 3 1 16 0;
#X connect 4 0 14 1;
#X connect 4 0 28 2;
#X connect 4 0 32 0;
#X connect 5 0 50 0;
#X connect 6 0 37 0;
#X connect 6 0 51 0;
#X connect 7 0 8 0;
```

```
#X connect 7 1 9 0;
#X connect 8 0 28 4;
#X connect 9 0 28 5;
#X connect 10 0 7 0;
#X connect 11 0 12 0;
#X connect 12 0 10 0;
#X connect 13 0 17 0;
#X connect 14 0 15 1;
#X connect 14 0 38 0;
#X connect 15 0 35 0;
#X connect 16 0 19 0;
#X connect 16 0 41 0;
#X connect 17 0 15 0;
#X connect 17 0 34 0;
#X connect 18 0 27 1;
#X connect 19 0 11 0;
#X connect 20 0 4 0;
#X connect 21 0 29 0;
#X connect 22 0 23 0;
#X connect 22 1 23 1;
#X connect 23 0 24 0;
#X connect 24 0 25 0;
#X connect 24 1 25 1;
#X connect 25 0 27 0;
#X connect 26 0 22 0;
#X connect 26 1 13 0;
#X connect 27 0 36 0;
#X connect 28 0 46 0;
#X connect 29 0 30 0;
#X connect 29 1 3 0;
#X connect 30 0 31 0;
#X connect 30 0 28 0;
#X connect 31 0 67 0;
#X connect 32 0 18 1;
#X connect 33 0 50 0;
#X connect 34 0 14 0;
#X connect 35 0 18 0;
#X connect 36 0 28 1;
#X connect 37 0 39 0;
#X connect 38 0 28 3;
#X connect 39 0 40 0;
#X connect 39 1 40 1;
#X connect 40 0 38 1;
#X connect 40 0 36 1;
#X connect 41 0 20 0;
#X connect 42 0 1 0;
#X connect 43 0 0 0;
#X connect 44 0 43 0;
#X connect 45 0 42 0;
#X connect 46 0 52 0;
#X connect 49 0 48 0;
#X connect 50 0 6 0;
#X connect 50 1 49 0;
#X connect 51 0 20 1;
#X connect 52 0 53 0;
#X connect 53 0 54 0;
#X connect 54 0 55 0;
```

```
#X connect 55 0 56 0;
#X connect 56 0 57 0;
#X connect 57 0 58 0;
#X connect 58 0 59 0;
#X connect 59 0 60 0;
#X connect 60 0 61 0;
#X connect 61 0 62 0;
#X connect 62 0 63 0;
#X connect 63 0 64 0;
#X connect 64 0 65 0;
#X connect 65 0 66 0;
#X connect 67 0 30 1;
```

7 zzc/3_osc~.pd

```
#N canvas 369 115 450 557 10;
#X obj 141 165 vline~;
#X obj 163 122 vline~;
#X obj 207 463 *~;
#X obj 288 421 vline~;
#X obj 333 421 vline~;
#X obj 139 464 *~;
#X obj 141 186 vcf~ 100;
#X obj 140 101 unpack f f f f f;
#X obj 139 9 inlet;
#X obj 296 528 outlet;
#X msg 325 370 0 5 \, \$1 5 5;
#X msg 305 390 0 5 \, \$1 5 5;
#X msg 141 142 \$1 0 5 \, 0 1 5;
#X obj 141 208 hilbert~;
#X obj 140 264 *~ 512;
#X obj 140 285 +~ 1;
#X obj 137 338 *~;
#X obj 140 31 route \$2;
#X obj 17 293 env~ 16384;
#X obj 17 314 > 0;
#X obj 17 335 sel 0;
#X obj 17 356 f 0;
#X obj 45 513 switch~;
#X obj 108 350 bang;
#X obj 108 401 f 1;
#X obj 17 424 spigot 1;
#X obj 77 399 f 0;
#X obj 49 370 delay 10;
#X obj 48 397 f 1;
#X obj 139 527 throw~ \$1-l;
#X obj 207 527 throw~ \$1-r;
#X obj 141 229 expr~ atan2($v2 \, $v1)/(2.0*3.1415926)+0.5 \; pow(($v1*$v1+$v2*↙
    ↳ $v2)/128
\, 0.25);
#X obj 138 309 tabread4~ \$1-wave;
#X connect 0 0 6 0;
#X connect 1 0 6 1;
#X connect 2 0 30 0;
#X connect 3 0 5 1;
#X connect 4 0 2 1;
#X connect 5 0 29 0;
```

```
#X connect 6 0 13 0;
#X connect 7 0 12 0;
#X connect 7 1 1 0;
#X connect 7 2 6 2;
#X connect 7 3 11 0;
#X connect 7 4 10 0;
#X connect 8 0 17 0;
#X connect 10 0 4 0;
#X connect 11 0 3 0;
#X connect 12 0 0 0;
#X connect 13 0 31 0;
#X connect 13 1 31 1;
#X connect 14 0 15 0;
#X connect 15 0 32 0;
#X connect 16 0 5 0;
#X connect 16 0 2 0;
#X connect 16 0 18 0;
#X connect 17 0 7 0;
#X connect 17 0 23 0;
#X connect 17 1 9 0;
#X connect 18 0 19 0;
#X connect 19 0 20 0;
#X connect 20 0 21 0;
#X connect 21 0 25 0;
#X connect 23 0 24 0;
#X connect 23 0 26 0;
#X connect 23 0 27 0;
#X connect 24 0 22 0;
#X connect 25 0 22 0;
#X connect 26 0 25 1;
#X connect 27 0 28 0;
#X connect 28 0 25 1;
#X connect 31 0 14 0;
#X connect 31 1 16 1;
#X connect 32 0 16 0;
```

8 zzc/3_voice~.pd

```
#N canvas 0 0 439 696 10;
#X obj 259 532 throw~ \$2;
#X obj 327 532 throw~ \$3;
#X obj 40 14 r \$1;
#X obj 68 78 unpack f f;
#X obj 171 265 mtof;
#X obj 212 110 f \$4;
#X obj 216 299 t f f;
#X obj 216 330 cos;
#X obj 243 330 sin;
#X obj 217 279 * 3.14159;
#X obj 217 231 wrap;
#X obj 217 252 * 0.5;
#X obj 112 134 + 1;
#X obj 140 203 /;
#X obj 103 204 *;
#X obj 187 138 / 3000;
#X obj 112 159 + 0.3;
#X obj 103 247 /;
```

```
#X obj 217 206 * 4000;
#X obj 171 237 +;
#X obj 40 34 route \$4;
#X obj 57 149 t f f;
#X obj 57 170 *;
#X obj 57 191 t f f;
#X obj 57 212 *;
#X obj 56 126 t f f;
#X obj 57 251 *;
#X obj 93 394 pack f f f f f f;
#X obj 39 53 t b a;
#X obj 34 359 f 0;
#X obj 69 341 + 1;
#X obj 130 248 sqrt;
#X obj 209 56 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144 -1
-1;
#X obj 142 180 * 10000;
#X obj 102 225 * 100;
#X obj 247 428 catch~ \$0-l;
#X obj 322 428 catch~ \$0-r;
#X obj 171 207 / 18;
#X obj 320 476 expr~ sin($v1)/4;
#X obj 246 452 expr~ sin($v1)/4;
#X obj 65 423 3_osc~ \$0 0;
#X obj 252 134 s \$0-chord;
#X obj 240 60 loadbang;
#X obj 212 160 tabread \$0-chord;
#X obj 221 88 t b b;
#X obj 251 181 table \$0-chord 16;
#X obj 286 218 loadbang;
#X obj 286 287 send \$0-wave;
#X obj 286 308 table \$0-wave 515;
#X msg 286 251 sinesum 512 1 0 0.333333 0 0.2 0 0.142857 0 0.111111
0 0.0909091 \, normalize 1;
#X obj 65 443 3_osc~ \$0 1;
#X obj 65 463 3_osc~ \$0 2;
#X obj 65 483 3_osc~ \$0 3;
#X obj 65 503 3_osc~ \$0 4;
#X obj 65 523 3_osc~ \$0 5;
#X obj 65 543 3_osc~ \$0 6;
#X obj 65 563 3_osc~ \$0 7;
#X obj 65 583 3_osc~ \$0 8;
#X obj 65 603 3_osc~ \$0 9;
#X obj 65 623 3_osc~ \$0 10;
#X obj 65 643 3_osc~ \$0 11;
#X obj 65 663 3_osc~ \$0 12;
#X obj 65 683 3_osc~ \$0 13;
#X obj 65 703 3_osc~ \$0 14;
#X obj 65 723 3_osc~ \$0 15;
#X obj 69 362 mod 12;
#X msg 251 111 0 26 38 50 62 74 86;
#X connect 2 0 20 0;
#X connect 3 0 25 0;
#X connect 3 1 15 0;
#X connect 4 0 13 1;
#X connect 4 0 27 2;
#X connect 4 0 31 0;
```

```
#X connect 5 0 43 0;
#X connect 6 0 7 0;
#X connect 6 1 8 0;
#X connect 7 0 27 4;
#X connect 8 0 27 5;
#X connect 9 0 6 0;
#X connect 10 0 11 0;
#X connect 11 0 9 0;
#X connect 12 0 16 0;
#X connect 13 0 14 1;
#X connect 13 0 27 3;
#X connect 14 0 34 0;
#X connect 15 0 18 0;
#X connect 15 0 37 0;
#X connect 16 0 14 0;
#X connect 16 0 33 0;
#X connect 17 0 26 1;
#X connect 18 0 10 0;
#X connect 19 0 4 0;
#X connect 20 0 28 0;
#X connect 21 0 22 0;
#X connect 21 1 22 1;
#X connect 22 0 23 0;
#X connect 23 0 24 0;
#X connect 23 1 24 1;
#X connect 24 0 26 0;
#X connect 25 0 21 0;
#X connect 25 1 12 0;
#X connect 26 0 27 1;
#X connect 27 0 40 0;
#X connect 28 0 29 0;
#X connect 28 1 3 0;
#X connect 29 0 30 0;
#X connect 29 0 27 0;
#X connect 30 0 65 0;
#X connect 31 0 17 1;
#X connect 32 0 44 0;
#X connect 33 0 13 0;
#X connect 34 0 17 0;
#X connect 35 0 39 0;
#X connect 36 0 38 0;
#X connect 37 0 19 0;
#X connect 38 0 1 0;
#X connect 39 0 0 0;
#X connect 40 0 50 0;
#X connect 42 0 44 0;
#X connect 43 0 19 1;
#X connect 44 0 5 0;
#X connect 44 1 66 0;
#X connect 46 0 49 0;
#X connect 49 0 47 0;
#X connect 50 0 51 0;
#X connect 51 0 52 0;
#X connect 52 0 53 0;
#X connect 53 0 54 0;
#X connect 54 0 55 0;
#X connect 55 0 56 0;
```

```
#X connect 56 0 57 0;
#X connect 57 0 58 0;
#X connect 58 0 59 0;
#X connect 59 0 60 0;
#X connect 60 0 61 0;
#X connect 61 0 62 0;
#X connect 62 0 63 0;
#X connect 63 0 64 0;
#X connect 65 0 29 1;
#X connect 66 0 41 0;
```

9 zzc/4_osc~.pd

```
#N canvas 369 115 450 557 10;
#X obj 141 165 vline~;
#X obj 163 122 vline~;
#X obj 207 463 *~;
#X obj 288 421 vline~;
#X obj 333 421 vline~;
#X obj 139 464 *~;
#X obj 141 186 vcf~ 100;
#X obj 140 101 unpack f f f f f;
#X obj 139 9 inlet;
#X obj 296 528 outlet;
#X msg 325 370 0 5 \, \$1 5 5;
#X msg 305 390 0 5 \, \$1 5 5;
#X msg 141 142 \$1 0 5 \, 0 1 5;
#X obj 141 208 hilbert~;
#X obj 140 264 *~ 512;
#X obj 140 285 +~ 1;
#X msg 16 52 sinesum 512 1 0.5 0.25 0.125 0.0625 0.03125 \, normalize
1;
#X obj 29 7 loadbang;
#X obj 137 338 *~;
#X obj 16 124 send \$0-wave;
#X obj 16 145 table \$0-wave 515;
#X obj 138 309 tabread4~ \$0-wave;
#X obj 140 31 route \$2;
#X obj 17 293 env~ 16384;
#X obj 17 314 > 0;
#X obj 17 335 sel 0;
#X obj 17 356 f 0;
#X obj 45 513 switch~;
#X obj 108 350 bang;
#X obj 108 401 f 1;
#X obj 17 424 spigot 1;
#X obj 77 399 f 0;
#X obj 49 370 delay 10;
#X obj 48 397 f 1;
#X obj 139 527 throw~ \$1-l;
#X obj 207 527 throw~ \$1-r;
#X obj 141 229 expr~ atan2($v2 \, $v1)/(2.0*3.1415926)+0.5 \; pow(($v1*$v1+$v2*↙
    ↳ $v2)/128
\, 2);
#X connect 0 0 6 0;
#X connect 1 0 6 1;
#X connect 2 0 35 0;
```

```
#X connect 3 0 5 1;
#X connect 4 0 2 1;
#X connect 5 0 34 0;
#X connect 6 0 13 0;
#X connect 7 0 12 0;
#X connect 7 1 1 0;
#X connect 7 2 6 2;
#X connect 7 3 11 0;
#X connect 7 4 10 0;
#X connect 8 0 22 0;
#X connect 10 0 4 0;
#X connect 11 0 3 0;
#X connect 12 0 0 0;
#X connect 13 0 36 0;
#X connect 13 1 36 1;
#X connect 14 0 15 0;
#X connect 15 0 21 0;
#X connect 16 0 19 0;
#X connect 17 0 16 0;
#X connect 18 0 5 0;
#X connect 18 0 2 0;
#X connect 18 0 23 0;
#X connect 21 0 18 0;
#X connect 22 0 7 0;
#X connect 22 0 28 0;
#X connect 22 1 9 0;
#X connect 23 0 24 0;
#X connect 24 0 25 0;
#X connect 25 0 26 0;
#X connect 26 0 30 0;
#X connect 28 0 29 0;
#X connect 28 0 31 0;
#X connect 28 0 32 0;
#X connect 29 0 27 0;
#X connect 30 0 27 0;
#X connect 31 0 30 1;
#X connect 32 0 33 0;
#X connect 33 0 30 1;
#X connect 36 0 14 0;
#X connect 36 1 18 1;
```

10 zzc/4_voice~.pd

```
#N canvas 0 0 692 696 10;
#X obj 259 532 throw~ \$2;
#X obj 327 532 throw~ \$3;
#X obj 40 14 r \$1;
#X obj 68 78 unpack f f;
#X obj 171 265 mtof;
#X obj 228 54 loadbang;
#X obj 231 112 f \$4;
#X obj 216 299 t f f;
#X obj 216 330 cos;
#X obj 243 330 sin;
#X obj 217 279 * 3.14159;
#X obj 217 231 wrap;
#X obj 217 252 * 0.5;
```

```
#X obj 112 134 + 1;
#X obj 140 203 /;
#X obj 103 204 *;
#X obj 187 138 / 3000;
#X obj 112 159 + 0.3;
#X obj 103 247 /;
#X obj 217 206 * 4000;
#X obj 171 237 +;
#X obj 40 34 route \$4;
#X obj 57 149 t f f;
#X obj 57 170 *;
#X obj 57 191 t f f;
#X obj 57 212 *;
#X obj 56 126 t f f;
#X obj 57 251 *;
#X obj 93 394 pack f f f f f f;
#X obj 39 53 t b a;
#X obj 34 359 f 0;
#X obj 69 341 + 1;
#X obj 130 248 sqrt;
#X obj 207 52 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144 -1
-1;
#X obj 142 180 * 10000;
#X obj 102 225 * 100;
#X obj 69 362 mod 16;
#X obj 247 428 catch~ \$0-l;
#X obj 322 428 catch~ \$0-r;
#X obj 180 301 + 1;
#X obj 151 370 *;
#X obj 113 361 *;
#X obj 177 329 t f f;
#X obj 177 350 *;
#X obj 320 476 expr~ sin($v1);
#X obj 246 452 expr~ sin($v1);
#X obj 171 207 / 60;
#X obj 65 423 4_osc~ \$0 0;
#X obj 65 443 4_osc~ \$0 1;
#X obj 65 463 4_osc~ \$0 2;
#X obj 65 483 4_osc~ \$0 3;
#X obj 65 503 4_osc~ \$0 4;
#X obj 65 523 4_osc~ \$0 5;
#X obj 65 543 4_osc~ \$0 6;
#X obj 65 563 4_osc~ \$0 7;
#X obj 65 583 4_osc~ \$0 8;
#X obj 65 603 4_osc~ \$0 9;
#X obj 65 623 4_osc~ \$0 10;
#X obj 65 643 4_osc~ \$0 11;
#X obj 65 663 4_osc~ \$0 12;
#X obj 65 683 4_osc~ \$0 13;
#X obj 65 703 4_osc~ \$0 14;
#X obj 65 723 4_osc~ \$0 15;
#X obj 381 131 table \$0-chord 16;
#X obj 309 132 s \$0-chord;
#X obj 232 160 tabread \$0-chord;
#X obj 230 85 t b b;
#X msg 308 104 0 31 38 43 59 62 67 74;
#X connect 2 0 21 0;
```

```
#X connect 3 0 26 0;
#X connect 3 1 16 0;
#X connect 4 0 14 1;
#X connect 4 0 28 2;
#X connect 4 0 32 0;
#X connect 5 0 66 0;
#X connect 6 0 39 0;
#X connect 6 0 65 0;
#X connect 7 0 8 0;
#X connect 7 1 9 0;
#X connect 8 0 28 4;
#X connect 9 0 28 5;
#X connect 10 0 7 0;
#X connect 11 0 12 0;
#X connect 12 0 10 0;
#X connect 13 0 17 0;
#X connect 14 0 15 1;
#X connect 14 0 40 0;
#X connect 15 0 35 0;
#X connect 16 0 19 0;
#X connect 16 0 46 0;
#X connect 17 0 15 0;
#X connect 17 0 34 0;
#X connect 18 0 27 1;
#X connect 19 0 11 0;
#X connect 20 0 4 0;
#X connect 21 0 29 0;
#X connect 22 0 23 0;
#X connect 22 1 23 1;
#X connect 23 0 24 0;
#X connect 24 0 25 0;
#X connect 24 1 25 1;
#X connect 25 0 27 0;
#X connect 26 0 22 0;
#X connect 26 1 13 0;
#X connect 27 0 41 0;
#X connect 28 0 47 0;
#X connect 29 0 30 0;
#X connect 29 1 3 0;
#X connect 30 0 31 0;
#X connect 30 0 28 0;
#X connect 31 0 36 0;
#X connect 32 0 18 1;
#X connect 33 0 66 0;
#X connect 34 0 14 0;
#X connect 35 0 18 0;
#X connect 36 0 30 1;
#X connect 37 0 45 0;
#X connect 38 0 44 0;
#X connect 39 0 41 1;
#X connect 39 0 42 0;
#X connect 40 0 28 3;
#X connect 41 0 28 1;
#X connect 42 0 43 0;
#X connect 42 1 43 1;
#X connect 43 0 40 1;
#X connect 44 0 1 0;
```

```
#X connect 45 0 0 0;
#X connect 46 0 20 0;
#X connect 47 0 48 0;
#X connect 48 0 49 0;
#X connect 49 0 50 0;
#X connect 50 0 51 0;
#X connect 51 0 52 0;
#X connect 52 0 53 0;
#X connect 53 0 54 0;
#X connect 54 0 55 0;
#X connect 55 0 56 0;
#X connect 56 0 57 0;
#X connect 57 0 58 0;
#X connect 58 0 59 0;
#X connect 59 0 60 0;
#X connect 60 0 61 0;
#X connect 61 0 62 0;
#X connect 65 0 20 1;
#X connect 66 0 6 0;
#X connect 66 1 67 0;
#X connect 67 0 64 0;
```

11 zzc/analysis.pd

```
#N canvas 340 0 353 256 10;
#X obj 11 11 cnv 15 318 126 empty empty empty 4 12 0 24 -203904 -16662
0;
#X obj 17 50 nbx 7 18 -1e+37 1e+37 0 0 \$0-target-re-s \$0-target-re-r
re 0 -8 0 16 -262130 -16662 -16662 0 256;
#X obj 138 50 nbx 7 18 -1e+37 1e+37 0 0 \$0-target-im-s \$0-target-im-r
im 0 -8 0 16 -262130 -16662 -16662 0 256;
#X obj 259 50 nbx 3 18 -1e+37 1e+37 0 0 \$0-target-zoom-s \$0-target-zoom-r
zoom 0 -8 0 16 -262130 -16662 -16662 0 256;
#X obj 20 16 tgl 18 0 \$0-analysis-enable-s \$0-analysis-enable-r enable
24 9 0 16 -4034 -13381 -16662 0 1;
#X obj 17 90 nbx 7 18 1000 100000 0 0 \$0-analysis-max-s \$0-analysis-max-r
maximum 0 -8 0 16 -4034 -13381 -16662 1000 256;
#X obj 137 91 nbx 7 18 10000 100000 0 0 \$0-analysis-settle-s \$0-analysis-
    ↳ settle-r
settle 0 -8 0 16 -4034 -13381 -16662 10000 256;
#X obj 258 77 bng 18 250 50 0 \$0-coords-save-s \$0-coords-save-r save
24 9 0 16 -4034 -13381 -16662;
#X obj 258 99 bng 18 250 50 0 \$0-coords-load-s \$0-coords-load-r load
24 9 0 16 -4034 -13381 -16662;
#X obj 249 119 cnv 18 80 18 empty empty periods 4 9 0 16 -203904 -16662
0;
#X obj 16 116 display;
#X obj 16 144 r \$0-periods-r;
#X obj 15 171 inlet;
#N canvas 0 0 569 304 \$0-guts 0;
#X obj 20 19 inlet;
#X obj 20 96 s \$0-analysis-enable-r;
#X obj 20 73 route enable re im zoom maximum settle save load periods
;
#X obj 57 116 s \$0-target-re-r;
#X obj 94 138 s \$0-target-im-r;
#X obj 131 162 s \$0-target-zoom-r;
```

```
#X obj 168 95 s \$0-analysis-max-r;
#X obj 205 116 s \$0-analysis-settle-r;
#X obj 242 138 s \$0-coords-save-r;
#X obj 279 158 s \$0-coords-load-r;
#X obj 316 93 s \$0-periods-r;
#X obj 20 187 r \$0-analysis-enable-s;
#X obj 162 186 r \$0-analysis-max-s;
#X obj 287 186 r \$0-analysis-settle-s;
#X obj 162 224 r \$0-coords-save-s;
#X obj 282 224 r \$0-coords-load-s;
#X msg 20 209 enable \$1;
#X msg 162 206 maximum \$1;
#X msg 287 205 settle \$1;
#X msg 162 246 save;
#X msg 282 246 load;
#X obj 21 279 outlet;
#X connect 0 0 2 0;
#X connect 2 0 1 0;
#X connect 2 1 3 0;
#X connect 2 2 4 0;
#X connect 2 3 5 0;
#X connect 2 4 6 0;
#X connect 2 5 7 0;
#X connect 2 6 8 0;
#X connect 2 7 9 0;
#X connect 2 8 10 0;
#X connect 11 0 16 0;
#X connect 12 0 17 0;
#X connect 13 0 18 0;
#X connect 14 0 19 0;
#X connect 15 0 20 0;
#X connect 16 0 21 0;
#X connect 17 0 21 0;
#X connect 18 0 21 0;
#X connect 19 0 21 0;
#X connect 20 0 21 0;
#X restore 15 192 pd \$0-guts;
#X obj 15 213 outlet;
#X obj 114 12 cnv 15 200 24 empty empty target-analysis 4 12 0 24 -203904
-16662 0;
#X connect 11 0 10 0;
#X connect 12 0 13 0;
#X connect 13 0 14 0;
#X coords 0 -1 1 1 320 128 2 10 10;
```

12 zzc/#autocorrelate.pd

```
#N canvas 0 0 371 566 10;
#X obj 20 19 inlet;
#X obj 93 24 #dim;
#X obj 93 47 #to_list;
#X msg 93 70 \$1;
#X obj 93 92 div 2;
#X obj 20 196 #for 0 1 1;
#X obj 20 246 #outer +;
#X msg 78 246 2 1 # 0 \$1;
#X obj 20 342 t a a;
```

```
#X obj 20 363 # *;
#X obj 21 384 #fold +;
#X obj 21 404 #fold +;
#X obj 21 426 # /;
#X obj 20 290 #transpose 1 2;
#X obj 20 318 #inner ( 2 d # 1 -1 ) \, seed ( d # 0 );
#X obj 20 502 #grade;
#X obj 20 268 #store;
#X obj 20 46 t a a a;
#X msg 93 139 1 # \$1;
#X obj 93 116 max 1;
#X obj 20 125 until;
#X obj 66 146 + 1;
#X obj 20 102 t f b;
#X msg 59 126 1;
#X obj 20 146 f;
#X obj 20 65 #finished;
#X obj 20 84 f;
#X msg 55 425 1 d # \$1;
#X obj 21 170 t b f f f;
#X obj 75 268 #cast d;
#X obj 20 545 outlet;
#X obj 20 524 # + 1;
#X obj 20 479 #import ( 1 ) d;
#X connect 0 0 17 0;
#X connect 1 0 2 0;
#X connect 2 0 3 0;
#X connect 3 0 4 0;
#X connect 4 0 19 0;
#X connect 5 0 6 0;
#X connect 6 0 16 0;
#X connect 7 0 6 1;
#X connect 8 0 9 0;
#X connect 8 1 9 1;
#X connect 9 0 10 0;
#X connect 10 0 11 0;
#X connect 11 0 12 0;
#X connect 12 0 32 0;
#X connect 13 0 14 0;
#X connect 14 0 8 0;
#X connect 15 0 31 0;
#X connect 16 0 13 0;
#X connect 17 0 25 0;
#X connect 17 1 1 0;
#X connect 17 2 29 0;
#X connect 18 0 32 1;
#X connect 19 0 18 0;
#X connect 19 0 26 1;
#X connect 20 0 24 0;
#X connect 21 0 24 1;
#X connect 22 0 20 0;
#X connect 22 1 23 0;
#X connect 23 0 24 1;
#X connect 24 0 21 0;
#X connect 24 0 28 0;
#X connect 25 0 26 0;
#X connect 26 0 22 0;
```

```
#X connect 27 0 12 1;
#X connect 28 0 5 0;
#X connect 28 1 5 1;
#X connect 28 2 27 0;
#X connect 28 3 7 0;
#X connect 29 0 16 1;
#X connect 31 0 30 0;
#X connect 32 0 15 0;
```

13 zzc/#autoscale.pd

```
#N canvas 56 202 530 419 10;
#X obj 109 107 #fold max;
#X obj 109 134 # -;
#X obj 172 107 #fold min;
#X obj 109 152 #fold max;
#X obj 157 132 # +;
#X obj 260 213 # / ( 1 d # 2 );
#X obj 73 318 # -;
#X obj 73 366 # /;
#X obj 73 7 inlet;
#X obj 73 392 outlet;
#X obj 109 65 #transpose -1 -2;
#X obj 263 97 #dim;
#X obj 263 123 #reverse;
#X msg 263 175 \$1 d #;
#X obj 109 86 t a a;
#X obj 108 234 #outer ignore;
#X obj 353 96 #dim;
#X obj 108 253 #outer ignore ( 1 d # );
#X obj 108 277 #transpose -1 0;
#X obj 108 297 #transpose -1 -2;
#X obj 259 236 #outer ignore ( 1 d # );
#X obj 259 260 #transpose -1 0;
#X obj 259 280 #transpose -1 -2;
#X msg 353 177 \$2 d #;
#X obj 353 152 #to_list;
#X obj 263 150 #to_list;
#X obj 108 321 #transpose 0 1;
#X obj 259 300 #transpose 0 1;
#X obj 108 214 # / ( 1 d # 2 );
#X obj 109 172 t a a;
#X obj 109 193 # +;
#X obj 136 193 # == ( 1 d # 0 );
#X obj 73 45 t a a;
#X obj 73 27 #t;
#X obj 263 48 t a a;
#X connect 0 0 1 0;
#X connect 0 0 4 0;
#X connect 1 0 3 0;
#X connect 2 0 1 1;
#X connect 2 0 4 1;
#X connect 3 0 29 0;
#X connect 4 0 5 0;
#X connect 5 0 20 0;
#X connect 6 0 7 0;
#X connect 7 0 9 0;
```

```
#X connect 8 0 33 0;
#X connect 10 0 14 0;
#X connect 11 0 12 0;
#X connect 12 0 25 0;
#X connect 13 0 15 1;
#X connect 14 0 0 0;
#X connect 14 1 2 0;
#X connect 15 0 17 0;
#X connect 16 0 24 0;
#X connect 17 0 18 0;
#X connect 18 0 19 0;
#X connect 19 0 26 0;
#X connect 20 0 21 0;
#X connect 21 0 22 0;
#X connect 22 0 27 0;
#X connect 23 0 20 1;
#X connect 23 0 17 1;
#X connect 24 0 23 0;
#X connect 25 0 13 0;
#X connect 26 0 7 1;
#X connect 27 0 6 1;
#X connect 28 0 15 0;
#X connect 29 0 30 0;
#X connect 29 1 31 0;
#X connect 30 0 28 0;
#X connect 31 0 30 1;
#X connect 32 0 6 0;
#X connect 32 1 10 0;
#X connect 33 0 32 0;
#X connect 33 1 34 0;
#X connect 34 0 11 0;
#X connect 34 1 16 0;
```

14 zzc/avmode.pd

```
#N canvas 0 0 344 116 10;
#X obj 11 11 cnv 15 190 77 empty empty avmode 4 12 0 24 -203904 -16662
0;
#X obj 16 33 bng 18 250 50 0 \$0-audio-s \$0-audio-r audio 24 9 0 16
-4034 -13381 -16662;
#X obj 116 33 bng 18 250 50 0 \$0-video-s \$0-video-r video 24 9 0
16 -4034 -13381 -16662;
#X obj 115 12 tgl 18 0 \$0-update-s \$0-update-r update 24 9 0 16 -4034
-13381 -16662 0 1;
#X obj 17 67 nbx 5 18 60 240 0 0 \$0-bpm-s \$0-bpm-r bpm 0 -8 0 16
-4034 -13381 -16662 60 256;
#X obj 115 67 nbx 5 18 1 100 0 0 \$0-fps-s \$0-fps-r fps 0 -8 0 16
-4034 -13381 -16662 1 256;
#N canvas 17 159 450 300 \$0-guts 0;
#X obj 19 13 inlet;
#X obj 19 55 route audio video update bpm fps;
#X obj 20 156 r \$0-audio-s;
#X msg 20 177 audio;
#X msg 60 197 video;
#X obj 60 176 r \$0-video-s;
#X obj 146 156 r \$0-bpm-s;
#X msg 146 177 bpm \$1;
```

```
#X msg 146 217 fps \$1;
#X obj 146 196 r \$0-fps-s;
#X obj 233 173 r \$0-update-s;
#X msg 233 194 update \$1;
#X obj 19 79 s \$0-audio-r;
#X obj 55 99 s \$0-video-r;
#X obj 99 78 s \$0-update-r;
#X obj 136 100 s \$0-bpm-r;
#X obj 188 79 s \$0-fps-r;
#X obj 20 242 outlet;
#X connect 0 0 1 0;
#X connect 1 0 12 0;
#X connect 1 1 13 0;
#X connect 1 2 14 0;
#X connect 1 3 15 0;
#X connect 1 4 16 0;
#X connect 2 0 3 0;
#X connect 3 0 17 0;
#X connect 4 0 17 0;
#X connect 5 0 4 0;
#X connect 6 0 7 0;
#X connect 7 0 17 0;
#X connect 8 0 17 0;
#X connect 9 0 8 0;
#X connect 10 0 11 0;
#X connect 11 0 17 0;
#X restore 230 34 pd \$0-guts;
#X obj 230 13 inlet;
#X obj 230 55 outlet;
#X connect 6 0 8 0;
#X connect 7 0 6 0;
#X coords 0 -1 1 1 192 79 2 10 10;
```

15 zzc/banner.pd

```
#N canvas 0 0 450 300 10;
#X obj 11 11 cnv 15 190 46 empty empty z->z^2+c 8 24 0 36 -203904 -16662
0;
#X coords 0 -1 1 1 192 48 2 10 10;
```

16 zzc/factorize.pd

```
#N canvas 0 0 341 412 10;
#X obj 24 22 inlet;
#X obj 24 61 t b f f b;
#X obj 24 42 max 1;
#X obj 36 238 list prepend;
#X obj 37 260 t a a;
#X obj 25 286 list append;
#X obj 25 313 outlet;
#X obj 36 85 until;
#X obj 36 106 f;
#X obj 64 106 + 1;
#X obj 36 134 t f f;
#X obj 59 155 swap;
#X obj 59 176 mod;
```

```
#X obj 59 196 ==;
#X obj 36 217 spigot;
#X msg 75 84 1;
#X connect 0 0 2 0;
#X connect 1 0 5 0;
#X connect 1 1 7 0;
#X connect 1 2 11 1;
#X connect 1 3 3 1;
#X connect 1 3 15 0;
#X connect 2 0 1 0;
#X connect 3 0 4 0;
#X connect 4 0 5 1;
#X connect 4 1 3 1;
#X connect 5 0 6 0;
#X connect 7 0 8 0;
#X connect 8 0 9 0;
#X connect 8 0 10 0;
#X connect 9 0 8 1;
#X connect 10 0 14 0;
#X connect 10 1 11 0;
#X connect 11 0 12 0;
#X connect 11 1 12 1;
#X connect 12 0 13 0;
#X connect 13 0 14 1;
#X connect 14 0 3 0;
#X connect 15 0 8 1;
```

17 zzc/footer.pd

```
#N canvas 0 0 536 106 10;
#X obj 11 11 cnv 15 511 30 empty empty Copyright_(C)_2018_Claude_Heiland-Allen_<
    ↪ claude@mathr.co.uk>
20 15 0 12 -203904 -16662 0;
#X coords 0 -1 1 1 513 32 2 10 10;
```

18 zzc/.gitignore

```
*.pd_linux
*.wav
```

19 zzc/gui.pd

```
#N canvas 0 0 716 697 10;
#X obj 101 101 banner;
#X obj 294 101 analysis;
#X obj 101 582 footer;
#X obj 37 638 outlet;
#X obj 45 28 inlet;
#X obj 101 150 avmode;
#X obj 101 230 periods;
#X obj 668 16 inlet;
#X obj 101 487 timing;
#X obj 294 487 presets;
#X connect 1 0 3 0;
#X connect 4 0 1 0;
#X connect 4 0 5 0;
```

```
#X connect 4 0 6 0;
#X connect 4 0 8 0;
#X connect 4 0 9 0;
#X connect 5 0 3 0;
#X connect 6 0 3 0;
#X connect 7 0 6 1;
#X connect 8 0 3 0;
#X connect 9 0 3 0;
#X coords 0 -1 1 1 515 515 2 100 100;
```

20 zzc/main.pd

```
#N canvas 0 86 566 611 10;
#X obj 11 39 gui;
#X obj 11 12 r \$0-gui-r;
#X obj 11 569 s \$0-gui-s;
#X obj 429 13 r \$0-analysis-r;
#N canvas 0 0 719 421 \$0-guts 0;
#X obj 19 15 r \$0-gui-s;
#X obj 20 267 s \$0-gui-r;
#X obj 11 104 s \$0-analysis-r;
#X obj 89 246 loadbang;
#N canvas 566 373 450 300 \$0-initial 0;
#X obj 17 13 inlet;
#X msg 17 39 re 0 \, im 0 \, zoom 0 \, maximum 3000 \, settle 30000
\, enable 1 \, update 1 \, bpm 120 \, fps 10 \, video \, runtime 0
\, rectime 0 \, preset 1 \, reset;
#X obj 17 94 outlet;
#X connect 0 0 1 0;
#X connect 1 0 2 0;
#X restore 89 267 pd \$0-initial;
#N canvas 24 236 450 300 \$0-analysis 0;
#X obj 162 119 #autocorrelate;
#X obj 162 138 #to_list;
#N canvas 96 47 296 424 \$0-mandelbrot-iteration 0;
#X obj 43 189 until;
#X obj 41 301 # +;
#X obj 56 258 t a a;
#X obj 56 279 # C.*;
#X obj 41 322 spigot;
#X msg 72 302 1;
#X msg 102 304 0;
#X msg 89 237 2 d # 0;
#X obj 42 215 #store ( 2 d # 0 );
#X obj 62 75 #finished;
#X obj 93 123 f 8192;
#X obj 94 169 until;
#X obj 44 122 f 4096;
#X obj 94 146 max 1;
#X obj 139 52 * 2;
#X msg 129 321 reset;
#X obj 59 95 t b b b b b;
#X obj 43 166 t f f;
#X obj 43 142 max 2;
#X msg 169 322 2 # \$1 2;
#X obj 65 49 t a a a;
#X obj 129 355 #import ( 2 2 ) d;
```

```
#X obj 45 3 inlet;
#X obj 139 17 inlet;
#X obj 212 17 inlet;
#X obj 129 386 outlet;
#X obj -7 69 realtime;
#X obj -7 91 print time;
#X obj -7 47 #finished;
#X obj 45 23 t a a b;
#X connect 0 0 8 0;
#X connect 1 0 2 0;
#X connect 1 0 4 0;
#X connect 2 0 3 0;
#X connect 2 1 3 1;
#X connect 3 0 1 1;
#X connect 4 0 21 0;
#X connect 5 0 4 1;
#X connect 6 0 4 1;
#X connect 7 0 2 0;
#X connect 8 0 1 0;
#X connect 9 0 16 0;
#X connect 10 0 13 0;
#X connect 11 0 8 0;
#X connect 12 0 18 0;
#X connect 13 0 11 0;
#X connect 14 0 12 1;
#X connect 15 0 21 0;
#X connect 16 0 12 0;
#X connect 16 1 5 0;
#X connect 16 2 10 0;
#X connect 16 3 6 0;
#X connect 16 4 15 0;
#X connect 17 0 0 0;
#X connect 17 1 19 0;
#X connect 18 0 17 0;
#X connect 19 0 21 1;
#X connect 20 0 9 0;
#X connect 20 1 7 0;
#X connect 20 2 8 1;
#X connect 21 0 25 0;
#X connect 22 0 29 0;
#X connect 23 0 14 0;
#X connect 24 0 10 1;
#X connect 28 0 26 1;
#X connect 29 0 28 0;
#X connect 29 1 20 0;
#X connect 29 2 26 0;
#X restore 19 64 pd \$0-mandelbrot-iteration;
#X obj 19 4 r \$0-target;
#X obj 37 119 s \$0-iteration-data;
#X obj 19 94 t a a a;
#X obj 29 43 r \$0-target2;
#X obj 19 24 spigot 1;
#X obj 161 180 outlet;
#X obj 20 175 outlet;
#X obj 105 6 inlet;
#X obj 156 7 inlet;
#X obj 201 10 inlet;
```

```
#X obj 161 156 list split 8;
#X connect 0 0 1 0;
#X connect 1 0 13 0;
#X connect 2 0 5 0;
#X connect 3 0 7 0;
#X connect 5 0 9 0;
#X connect 5 1 4 0;
#X connect 5 2 0 0;
#X connect 6 0 2 0;
#X connect 7 0 2 0;
#X connect 10 0 7 1;
#X connect 11 0 2 1;
#X connect 12 0 2 2;
#X connect 13 0 8 0;
#X restore 12 81 pd \$0-analysis;
#X obj 20 137 list prepend periods;
#X obj 20 158 list trim;
#N canvas 162 0 983 603 \$0-sonify 0;
#X obj 280 13 inlet;
#X obj 175 8 inlet;
#X obj 582 520 outlet~;
#X obj 638 516 outlet~;
#X obj 175 66 #scan *;
#X obj 241 67 #fold *;
#X obj 175 225 #transpose 0 1;
#X obj 175 202 #redim ( 1 1 2 );
#X obj 175 91 #to_float;
#X obj 175 111 t f f;
#X obj 175 136 pack f f;
#X obj 220 112 inv*;
#X obj 241 87 #to_float;
#X msg 282 202 3 # \$2 \$1 2;
#X obj 202 267 #fold + \, seed ( 2 d # 0 );
#X obj 202 288 # /;
#X msg 234 289 1 d # \$2;
#X obj 175 334 # -;
#X obj 175 245 t a a;
#X obj 202 312 #outer ignore;
#X obj 175 157 t b a a a;
#X msg 294 312 \$2 d #;
#X obj 202 337 #transpose -1 -2;
#X obj 175 358 #autoscale;
#X obj 175 382 # c2p ( 2 d # );
#X obj 365 68 list length;
#X obj 320 154 pack f f;
#X obj 175 520 #import ( 1 1 2 ) d;
#X msg 320 520 3 # \$2 \$1 2;
#X obj 19 348 #store;
#X obj 20 107 f 0;
#X obj 78 105 + 1;
#X obj 19 215 #for 0 1 1;
#X obj 20 82 metro 118;
#X obj 78 127 mod;
#X obj 46 153 pack f f;
#X obj 19 328 #join 1;
#X msg 58 307 \$2 1 # \$1;
#X msg 47 173 \$2;
```

```
#X obj 19 374 #import ( 2 ) d;
#X obj 19 134 t b f b;
#X obj 19 397 #to_list;
#X obj 19 419 t a b;
#X obj 64 420 f;
#X obj 99 421 + 1;
#X obj 19 443 list prepend;
#X msg 79 399 0;
#X obj 21 13 inlet;
#X obj 175 179 #store ( 1 2 d # );
#X obj 19 503 s \$0-audio-control;
#N canvas 0 0 450 300 \$0-audio-voices 0;
#X restore 413 458 pd \$0-audio-voices;
#X obj 412 435 s pd-\$0-audio-voices;
#X obj 402 140 t b f b;
#X msg 402 170 \; pd dsp 1;
#X msg 471 168 \; pd dsp 0;
#X obj 421 238 until;
#X msg 531 351 clear;
#X obj 421 258 f;
#X obj 530 200 loadbang;
#X msg 530 221 bang;
#X obj 510 242 symbol \$0-audio-control;
#X obj 447 259 + 1;
#X msg 460 239 0;
#X obj 421 325 + 10;
#X obj 421 279 t f f;
#X obj 421 302 * 30;
#X msg 412 405 loadbang;
#X obj 412 201 t b f b b;
#X obj 175 47 list prepend 1;
#X obj 365 89 + 1;
#X obj 18 479 spigot;
#X obj 64 480 <;
#X msg 85 83 0;
#X obj 175 28 t a a a b;
#X obj 403 89 t f f;
#X obj 447 88 !=;
#X obj 402 115 spigot;
#X obj 175 403 t a a;
#X obj 175 426 # * ( 2 d # 1 0 );
#X obj 243 448 # * ( 2 d # 0 1 );
#X obj 175 448 #autoscale;
#X obj 175 489 # +;
#X obj 19 261 #reverse;
#X obj 19 287 #outer - ( 1 # 1 );
#X obj 520 266 symbol \$0-audio-out-l;
#X obj 530 294 symbol \$0-audio-out-r;
#X obj 584 443 catch~ \$0-audio-out-l;
#X obj 639 470 catch~ \$0-audio-out-r;
#X obj 583 495 *~ 0.2;
#X obj 638 493 *~ 0.2;
#X obj 71 44 r \$0-tempo;
#X obj 421 349 pack f s s s f f;
#X obj 520 13 inlet;
#X obj 520 35 f;
#X msg 422 383 obj 10 \$1 \$6_voice~ \$2 \$3 \$4 \$5 \$6;
```

```
#X obj 421 220 max 1;
#X connect 0 0 48 1;
#X connect 1 0 73 0;
#X connect 4 0 8 0;
#X connect 5 0 12 0;
#X connect 6 0 18 0;
#X connect 7 0 6 0;
#X connect 8 0 9 0;
#X connect 9 0 10 0;
#X connect 9 1 11 0;
#X connect 10 0 20 0;
#X connect 11 0 10 1;
#X connect 12 0 11 1;
#X connect 12 0 26 0;
#X connect 12 0 34 1;
#X connect 13 0 7 1;
#X connect 14 0 15 0;
#X connect 15 0 19 0;
#X connect 16 0 15 1;
#X connect 17 0 23 0;
#X connect 18 0 17 0;
#X connect 18 1 14 0;
#X connect 19 0 22 0;
#X connect 20 0 48 0;
#X connect 20 1 16 0;
#X connect 20 2 21 0;
#X connect 20 3 13 0;
#X connect 21 0 19 1;
#X connect 22 0 17 1;
#X connect 23 0 24 0;
#X connect 24 0 77 0;
#X connect 25 0 69 0;
#X connect 25 0 71 1;
#X connect 25 0 74 0;
#X connect 26 0 28 0;
#X connect 27 0 29 1;
#X connect 28 0 27 1;
#X connect 29 0 39 0;
#X connect 30 0 31 0;
#X connect 30 0 40 0;
#X connect 31 0 34 0;
#X connect 32 0 82 0;
#X connect 33 0 30 0;
#X connect 34 0 30 1;
#X connect 35 0 37 0;
#X connect 35 0 38 0;
#X connect 36 0 29 0;
#X connect 37 0 36 1;
#X connect 38 0 32 1;
#X connect 39 0 41 0;
#X connect 40 0 32 0;
#X connect 40 1 35 0;
#X connect 40 2 46 0;
#X connect 41 0 42 0;
#X connect 42 0 45 0;
#X connect 42 1 43 0;
#X connect 43 0 44 0;
```

```
#X connect 43 0 45 1;
#X connect 43 0 71 0;
#X connect 44 0 43 1;
#X connect 45 0 70 0;
#X connect 46 0 43 1;
#X connect 47 0 33 0;
#X connect 48 0 7 0;
#X connect 52 0 53 0;
#X connect 52 1 67 0;
#X connect 52 2 54 0;
#X connect 55 0 57 0;
#X connect 56 0 51 0;
#X connect 57 0 61 0;
#X connect 57 0 64 0;
#X connect 58 0 59 0;
#X connect 59 0 60 0;
#X connect 59 0 84 0;
#X connect 59 0 85 0;
#X connect 60 0 91 1;
#X connect 61 0 57 1;
#X connect 62 0 57 1;
#X connect 63 0 91 0;
#X connect 64 0 65 0;
#X connect 64 1 91 4;
#X connect 65 0 63 0;
#X connect 66 0 51 0;
#X connect 67 0 66 0;
#X connect 67 1 95 0;
#X connect 67 2 62 0;
#X connect 67 3 56 0;
#X connect 68 0 4 0;
#X connect 69 0 26 1;
#X connect 69 0 35 1;
#X connect 70 0 49 0;
#X connect 71 0 70 1;
#X connect 72 0 30 1;
#X connect 73 0 68 0;
#X connect 73 1 5 0;
#X connect 73 2 25 0;
#X connect 73 3 72 0;
#X connect 74 0 76 0;
#X connect 74 1 75 0;
#X connect 75 0 76 1;
#X connect 76 0 52 0;
#X connect 77 0 78 0;
#X connect 77 1 79 0;
#X connect 78 0 80 0;
#X connect 79 0 81 1;
#X connect 80 0 81 0;
#X connect 81 0 27 0;
#X connect 82 0 83 0;
#X connect 83 0 36 0;
#X connect 84 0 91 2;
#X connect 85 0 91 3;
#X connect 86 0 88 0;
#X connect 87 0 89 0;
#X connect 88 0 2 0;
```

```
#X connect 89 0 3 0;
#X connect 90 0 33 1;
#X connect 91 0 94 0;
#X connect 92 0 93 0;
#X connect 93 0 91 5;
#X connect 94 0 51 0;
#X connect 95 0 55 0;
#X restore 180 228 pd \$0-sonify;
#X obj 190 249 dac~;
#N canvas 0 0 488 444 \$0-record 0;
#X obj 19 11 inlet~;
#X obj 76 11 inlet~;
#X obj 124 11 inlet;
#X obj 196 122 date GMT;
#X obj 307 121 time GMT;
#X obj 196 92 t b b a;
#X obj 196 240 l2s;
#X msg 241 239 symbol -;
#X obj 196 217 pack s s s s s s;
#X obj 315 241 l2s;
#X msg 346 240 symbol -;
#X obj 310 266 loadbang;
#X obj 196 271 pack s s;
#X obj 196 295 l2s;
#X msg 227 294 symbol --;
#X obj 343 182 makefilename %02d;
#X obj 324 163 makefilename %02d;
#X obj 306 143 makefilename %02d;
#X obj 196 143 makefilename %02d;
#X obj 214 163 makefilename %02d;
#X obj 233 182 makefilename %02d;
#X obj 196 389 outlet;
#X msg 196 323 symbol \$1.wav;
#X msg 103 358 open -bytes 4 \$1 \, start;
#X obj 20 395 writesf~ 2;
#X obj 103 330 symbol;
#X obj 103 61 route start stop;
#X obj 103 83 bang;
#X obj 149 85 bang;
#X msg 149 207 stop;
#X connect 0 0 24 0;
#X connect 1 0 24 1;
#X connect 2 0 26 0;
#X connect 3 0 18 0;
#X connect 3 1 19 0;
#X connect 3 2 20 0;
#X connect 4 0 17 0;
#X connect 4 1 16 0;
#X connect 4 2 15 0;
#X connect 5 0 3 0;
#X connect 5 1 4 0;
#X connect 5 2 9 0;
#X connect 6 0 12 0;
#X connect 7 0 6 1;
#X connect 8 0 6 0;
#X connect 9 0 12 1;
#X connect 10 0 9 1;
```

```
#X connect 11 0 10 0;
#X connect 11 0 7 0;
#X connect 11 0 14 0;
#X connect 12 0 13 0;
#X connect 13 0 22 0;
#X connect 14 0 13 1;
#X connect 15 0 8 5;
#X connect 16 0 8 4;
#X connect 17 0 8 3;
#X connect 18 0 8 0;
#X connect 19 0 8 1;
#X connect 20 0 8 2;
#X connect 22 0 25 1;
#X connect 23 0 24 0;
#X connect 25 0 23 0;
#X connect 25 0 21 0;
#X connect 26 0 27 0;
#X connect 26 1 28 0;
#X connect 26 2 5 0;
#X connect 27 0 25 0;
#X connect 28 0 29 0;
#X connect 29 0 24 0;
#X restore 180 268 pd \$0-record;
#X msg 187 104 0;
#X msg 119 105 1;
#X msg 153 105 0;
#X msg 220 105 1;
#N canvas 0 0 450 300 \$0-record-timer 0;
#X msg 169 219 start;
#X msg 225 244 stop;
#X obj 194 148 delay 2000;
#X msg 137 221 1;
#X obj 224 120 + 2000;
#X obj 224 168 delay;
#X msg 193 242 0;
#X obj 224 221 delay 30000;
#X obj 224 99 f;
#X obj 292 55 inlet;
#X obj 169 56 inlet;
#X obj 187 272 outlet;
#X obj 94 269 outlet;
#X obj 8 267 outlet;
#X obj 332 55 inlet;
#X obj 292 86 swap 15000;
#X obj 292 107 /;
#X obj 292 139 pack f f;
#X obj 292 160 *;
#X obj 331 114 t b f;
#X msg 291 185 runtime \$1;
#X obj 45 116 timer;
#X msg 46 143 rectime \$1;
#X obj 66 43 loadbang;
#X obj 66 64 metro 40;
#X obj 337 159 s \$0-tempo;
#X msg 128 115 stop;
#X obj 169 76 delay 30000;
#X connect 0 0 11 0;
```

```
#X connect 1 0 11 0;
#X connect 2 0 3 0;
#X connect 2 0 21 0;
#X connect 3 0 12 0;
#X connect 4 0 5 0;
#X connect 5 0 6 0;
#X connect 5 0 7 0;
#X connect 6 0 12 0;
#X connect 7 0 1 0;
#X connect 8 0 4 0;
#X connect 9 0 15 0;
#X connect 10 0 27 0;
#X connect 10 0 6 0;
#X connect 14 0 19 0;
#X connect 15 0 16 0;
#X connect 15 1 16 1;
#X connect 16 0 17 0;
#X connect 16 0 25 0;
#X connect 17 0 18 0;
#X connect 18 0 20 0;
#X connect 18 0 8 1;
#X connect 19 0 17 0;
#X connect 19 1 17 1;
#X connect 20 0 13 0;
#X connect 21 0 22 0;
#X connect 22 0 13 0;
#X connect 23 0 24 0;
#X connect 24 0 21 1;
#X connect 26 0 5 0;
#X connect 26 0 2 0;
#X connect 26 0 7 0;
#X connect 27 0 8 0;
#X connect 27 0 0 0;
#X connect 27 0 2 0;
#X restore 186 175 pd \$0-record-timer;
#X obj 390 364 #out window \, title Mandelbrot;
#X obj 384 233 metro 100;
#N canvas 0 0 607 485 \$0-mandelbrot 0;
#X obj 246 397 # + ( d # 0 );
#X obj 277 366 # C.*;
#X obj 277 346 t a a;
#X obj 264 124 #for ( 0 0 ) ( 512 512 ) ( 1 1 );
#X obj 263 203 #cast d;
#X obj 263 185 # - ( 2 # 256 );
#X obj 191 353 #store ( 512 768 2 d # 0 );
#X obj 203 284 t b b;
#X msg 229 331 1;
#X obj 16 217 t a b;
#X msg 211 312 0;
#X msg 14 242 2 # \$1 \$2;
#X msg 16 70 \$3 \$1 \$2;
#X obj 15 184 t a b;
#X obj 261 301 # + ( 2 d # 0 0 );
#X obj 376 196 # * ( 1 d # 128 );
#X obj 376 219 #t;
#X msg 363 175 1 d # 2;
#X obj 62 183 t a b;
```

```
#X msg 415 175 1 d # 0.5;
#X obj 277 324 # put ( 1 d # 0 );
#X obj 16 163 route 256 1024;
#X msg 106 294 2 d # 0 0;
#X obj 263 145 #transpose 0 1;
#X obj 263 167 #reverse 0;
#N canvas 0 0 545 369 \$0-map 0;
#X obj 27 305 #store;
#X obj 60 174 #cast d;
#X obj 171 170 #cast d;
#X obj 170 69 loadbang;
#X obj 60 212 #transpose 0 2;
#X obj 60 235 #transpose 2 3;
#X obj 28 7 inlet;
#X obj 27 29 t b b;
#X obj 60 134 #reverse 1;
#X msg 171 92 1 l # 16;
#X obj 171 149 #outer ignore ( 4 l # );
#X obj 60 256 #redim ( 512 256 4 );
#X obj 27 277 #for ( 0 0 0 ) ( 512 256 2 ) ( 1 1 1 );
#X obj 59 49 #for ( 0 0 0 0 ) ( 64 64 8 4 ) ( 1 1 1 1 );
#X obj 60 154 # - ( 8 4 # 32 );
#X obj 60 192 # / ( 8 4 4 d # 1 );
#X obj 171 116 #outer << ( 8 4 l # 0 1 2 3 4 5 6 7 8 9 10 11 12 13
14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 );
#X obj 28 333 outlet;
#X connect 0 0 17 0;
#X connect 1 0 15 0;
#X connect 2 0 15 1;
#X connect 3 0 9 0;
#X connect 4 0 5 0;
#X connect 5 0 11 0;
#X connect 6 0 7 0;
#X connect 7 0 12 0;
#X connect 7 1 13 0;
#X connect 8 0 14 0;
#X connect 9 0 16 0;
#X connect 10 0 2 0;
#X connect 11 0 0 1;
#X connect 12 0 0 0;
#X connect 13 0 8 0;
#X connect 14 0 1 0;
#X connect 15 0 4 0;
#X connect 16 0 10 0;
#X restore 263 260 pd \$0-map;
#X obj 262 281 #join 1;
#X obj 263 242 #t;
#X obj 535 22 inlet;
#X obj 239 18 inlet;
#X obj 518 454 outlet;
#X obj 151 455 outlet;
#X obj 170 17 inlet;
#X obj 390 456 outlet;
#X obj 265 75 t a a;
#X obj 265 100 #finished;
#X msg 478 175 1 d # 128;
#X obj 70 213 s \$0-zoom-in;
```

```
#X obj 363 149 r \$0-zoom-in;
#X obj 447 150 r \$0-zoom-out;
#X obj 104 240 s \$0-zoom-out;
#X obj 479 130 r \$0-zoom-reset;
#X obj 16 48 route position keypress;
#X obj 97 94 unpack f f f s;
#X obj 148 212 #store ( 2 d # 0 );
#X obj 160 171 t b b;
#X obj 200 171 t b b;
#X obj 120 171 t b b;
#X obj 127 260 s \$0-zoom-reset;
#X obj 263 222 # / ( 1 d # 128 );
#X obj 191 373 shunt 3;
#X obj 243 454 outlet;
#X msg 197 70 \$4 \$1 \$2;
#X obj 195 90 route space;
#X obj 209 192 t a b;
#X msg 195 235 2 # \$1 \$2;
#X msg 238 307 2;
#X obj 121 119 sel Home Up Down Return;
#X obj 114 404 #store ( 2 d # 0 );
#X obj 276 390 # min ( d # 1000 );
#X obj 276 411 # max ( d # -1000 );
#X obj 104 141 sel KP_Home KP_Up KP_Down;
#X obj 16 92 unpack f f f;
#X obj 16 113 & 1280;
#X obj 16 132 pack f f f;
#X connect 0 0 2 0;
#X connect 0 0 30 0;
#X connect 1 0 59 0;
#X connect 2 0 1 0;
#X connect 2 1 1 1;
#X connect 3 0 23 0;
#X connect 4 0 49 0;
#X connect 5 0 4 0;
#X connect 6 0 50 0;
#X connect 7 0 6 0;
#X connect 7 1 8 0;
#X connect 8 0 50 1;
#X connect 9 0 11 0;
#X connect 9 1 10 0;
#X connect 10 0 50 1;
#X connect 11 0 6 0;
#X connect 12 0 62 0;
#X connect 13 0 9 0;
#X connect 13 1 37 0;
#X connect 14 0 6 1;
#X connect 14 0 20 0;
#X connect 15 0 16 0;
#X connect 16 0 49 1;
#X connect 16 1 15 1;
#X connect 16 1 33 0;
#X connect 17 0 15 0;
#X connect 18 0 9 0;
#X connect 18 1 40 0;
#X connect 19 0 15 0;
#X connect 20 0 2 0;
```

```
#X connect 21 0 13 0;
#X connect 21 1 18 0;
#X connect 22 0 31 0;
#X connect 23 0 24 0;
#X connect 24 0 5 0;
#X connect 25 0 26 0;
#X connect 26 0 14 0;
#X connect 27 0 25 0;
#X connect 27 1 26 1;
#X connect 28 0 42 0;
#X connect 29 0 7 0;
#X connect 32 0 34 0;
#X connect 34 0 35 0;
#X connect 34 1 14 1;
#X connect 34 1 44 1;
#X connect 35 0 3 0;
#X connect 36 0 16 0;
#X connect 38 0 17 0;
#X connect 39 0 19 0;
#X connect 41 0 36 0;
#X connect 42 0 12 0;
#X connect 42 1 43 0;
#X connect 42 1 52 0;
#X connect 43 3 57 0;
#X connect 44 0 31 0;
#X connect 45 0 44 0;
#X connect 45 1 37 0;
#X connect 46 0 44 0;
#X connect 46 1 40 0;
#X connect 47 0 22 0;
#X connect 47 1 48 0;
#X connect 49 0 27 0;
#X connect 50 0 31 0;
#X connect 50 1 0 0;
#X connect 50 2 51 0;
#X connect 50 2 58 1;
#X connect 52 0 53 0;
#X connect 53 0 54 0;
#X connect 54 0 55 0;
#X connect 54 1 56 0;
#X connect 55 0 6 0;
#X connect 56 0 50 1;
#X connect 57 0 47 0;
#X connect 57 1 45 0;
#X connect 57 2 46 0;
#X connect 57 3 58 0;
#X connect 57 4 61 0;
#X connect 58 0 31 0;
#X connect 59 0 60 0;
#X connect 60 0 0 1;
#X connect 61 0 47 0;
#X connect 61 1 45 0;
#X connect 61 2 46 0;
#X connect 62 0 63 0;
#X connect 62 1 64 1;
#X connect 62 2 64 2;
#X connect 63 0 64 0;
```

```
#X connect 64 0 21 0;
#X restore 297 256 pd \$0-mandelbrot;
#X obj 297 365 s \$0-target;
#X obj 297 226 r \$0-target;
#X obj 326 322 loadbang;
#X msg 326 343 2 d # 0;
#X obj 328 300 s \$0-target2;
#N canvas 115 0 572 691 \$0-innercolour 0;
#X obj 49 53 inlet;
#X obj 42 649 outlet;
#X obj 205 54 inlet;
#X obj 269 51 inlet;
#X obj 29 99 # *;
#X obj 29 120 #inner ( 2 d # 1 1 ) \, seed ( d # 0 );
#X obj 30 142 # sqrt ( d # );
#X obj 28 200 t a a;
#X obj 28 232 # * ( d # 1 );
#X obj 29 367 t a a;
#X obj 221 236 sel 1;
#X obj 29 347 # min ( d # 4 );
#X obj 29 269 t a a;
#X obj 43 408 t a a;
#X obj 56 290 # < ( d # 4 );
#X obj 30 167 t a a;
#X obj 30 78 t a a;
#X obj 56 329 # * ( i # 1 );
#X obj 55 309 #cast i;
#X msg 158 201 1 i # \$1;
#X msg 203 269 1 d # 4;
#X obj 43 386 # max ( i # 0 );
#X msg 202 334 1 i # 0;
#X obj 71 201 # < ( d # 4 );
#X obj 42 628 # *;
#X obj 326 306 list split 8;
#X obj 326 340 #scan *;
#X obj 42 510 # ==;
#X obj 42 489 #outer % ( 8 # 1 );
#X obj 42 429 spigot;
#X obj 388 51 inlet;
#X obj 42 530 #inner ( 8 1 # 1 );
#X obj 91 625 #outer ignore ( 3 # );
#X obj 91 605 #cast i;
#X obj 326 285 list append 0 0 0 0 0 0 0 0;
#X obj 356 412 # ==;
#X obj 356 433 #fold +;
#X obj 42 551 # -;
#X obj 97 441 f;
#X obj 43 456 t a b;
#X obj 326 364 #to_list;
#X obj 281 389 listfind;
#X obj 281 412 != -1;
#X obj 281 432 moses 0.5;
#X obj 191 382 t f f;
#X obj 42 572 #store ( 9 3 # 255 0 0 255 188 0 255 255 0 128 255 0
0 255 128 0 128 255 0 0 255 255 0 255 255 255 255 );
#X obj 32 48 #cast d;
#X connect 0 0 46 0;
```

```
#X connect 2 0 19 0;
#X connect 2 0 10 0;
#X connect 2 0 41 0;
#X connect 3 0 34 0;
#X connect 4 0 5 0;
#X connect 5 0 6 0;
#X connect 6 0 15 0;
#X connect 7 0 8 0;
#X connect 7 1 23 0;
#X connect 8 0 12 0;
#X connect 9 1 11 1;
#X connect 9 1 14 1;
#X connect 10 0 20 0;
#X connect 10 0 22 0;
#X connect 11 0 9 0;
#X connect 12 0 11 0;
#X connect 12 1 14 0;
#X connect 13 0 29 0;
#X connect 13 1 21 1;
#X connect 14 0 18 0;
#X connect 15 0 7 0;
#X connect 16 0 4 0;
#X connect 16 1 4 1;
#X connect 17 0 21 0;
#X connect 18 0 17 0;
#X connect 19 0 17 1;
#X connect 20 0 11 1;
#X connect 20 0 14 1;
#X connect 21 0 13 0;
#X connect 22 0 21 1;
#X connect 23 0 8 1;
#X connect 23 0 33 0;
#X connect 24 0 1 0;
#X connect 25 0 26 0;
#X connect 25 0 35 0;
#X connect 26 0 28 1;
#X connect 26 0 40 0;
#X connect 27 0 31 0;
#X connect 28 0 27 0;
#X connect 29 0 39 0;
#X connect 30 0 44 0;
#X connect 31 0 37 0;
#X connect 32 0 24 1;
#X connect 33 0 32 0;
#X connect 34 0 25 0;
#X connect 35 0 36 0;
#X connect 36 0 37 1;
#X connect 37 0 45 0;
#X connect 38 0 29 1;
#X connect 39 0 28 0;
#X connect 39 1 38 0;
#X connect 40 0 41 1;
#X connect 41 0 42 0;
#X connect 42 0 43 0;
#X connect 43 1 29 1;
#X connect 44 0 29 1;
#X connect 44 1 38 1;
```

```
#X connect 45 0 24 0;
#X connect 46 0 16 0;
#X restore 407 339 pd \$0-innercolour;
#N canvas 0 0 450 300 \$0-zoom-level 0;
#X obj 32 71 # - ( 1 d # 7 );
#X obj 32 92 #to_float;
#X obj 32 30 # log ( 1 d # );
#X obj 32 50 # /;
#X obj 59 52 # log ( 1 d # );
#X msg 131 34 1 d # 2;
#X obj 131 13 loadbang;
#X obj 32 114 s \$0-zzzz;
#X obj 34 8 inlet;
#X connect 0 0 1 0;
#X connect 1 0 7 0;
#X connect 2 0 3 0;
#X connect 3 0 0 0;
#X connect 4 0 3 1;
#X connect 5 0 4 0;
#X connect 6 0 5 0;
#X connect 8 0 2 0;
#X restore 358 278 pd \$0-zoom-level;
#N canvas 0 0 450 300 \$0-counter 0;
#X obj 72 97 f;
#X obj 103 97 + 1;
#X obj 87 51 r \$0-target;
#X obj 31 97 t a b;
#X msg 87 73 1;
#X obj 28 23 inlet;
#X obj 29 127 outlet;
#X obj 78 127 outlet;
#X connect 0 0 1 0;
#X connect 0 0 7 0;
#X connect 1 0 0 1;
#X connect 2 0 4 0;
#X connect 3 0 6 0;
#X connect 3 1 0 0;
#X connect 4 0 0 1;
#X connect 5 0 3 0;
#X restore 407 301 pd \$0-counter;
#X obj 397 189 swap 1000;
#X obj 397 210 /;
#X msg 373 387 512 768 3 # 0;
#X obj 387 56 r \$0-target;
#X obj 39 298 r \$0-zzzz;
#N canvas 0 0 489 274 \$0-save-load 0;
#X obj 26 11 inlet;
#X obj 150 14 inlet;
#X obj 45 91 #store ( 2 d # 0 );
#X obj 26 41 savepanel;
#X obj 45 148 #out;
#X msg 26 111 close;
#X obj 290 15 inlet;
#X obj 290 36 openpanel;
#X obj 26 65 t b b a;
#X obj 290 140 outlet;
#X msg 290 59 load \$1 d;
```

```
#X msg 65 111 open \$1;
#X obj 290 99 #in;
#X obj 17 176 #print;
#X connect 0 0 3 0;
#X connect 1 0 2 1;
#X connect 2 0 4 0;
#X connect 2 0 13 0;
#X connect 3 0 8 0;
#X connect 5 0 4 0;
#X connect 6 0 7 0;
#X connect 7 0 10 0;
#X connect 8 0 5 0;
#X connect 8 1 2 0;
#X connect 8 2 11 0;
#X connect 10 0 12 0;
#X connect 11 0 4 0;
#X connect 12 0 9 0;
#X restore 344 87 pd \$0-save-load;
#X obj 344 106 s \$0-target;
#X msg 39 319 zoom \$1;
#X obj 226 204 r \$0-iteration-data;
#X obj 120 299 r \$0-target;
#X obj 120 320 #to_list;
#X msg 120 343 re \$1 \, im \$2;
#X obj 19 38 route enable maximum settle cascade audio video record
bpm total fps targets update save load preset;
#X connect 0 0 39 0;
#X connect 3 0 4 0;
#X connect 4 0 1 0;
#X connect 5 0 2 0;
#X connect 5 1 6 0;
#X connect 6 0 7 0;
#X connect 7 0 1 0;
#X connect 8 0 9 0;
#X connect 8 0 10 0;
#X connect 8 1 9 1;
#X connect 8 1 10 1;
#X connect 11 0 17 0;
#X connect 12 0 8 0;
#X connect 13 0 8 0;
#X connect 14 0 17 0;
#X connect 15 0 1 0;
#X connect 15 1 8 0;
#X connect 15 2 10 2;
#X connect 16 0 18 2;
#X connect 17 0 18 1;
#X connect 18 0 19 0;
#X connect 18 1 23 0;
#X connect 18 2 25 0;
#X connect 18 3 26 0;
#X connect 20 0 18 0;
#X connect 21 0 22 0;
#X connect 21 0 29 0;
#X connect 22 0 19 0;
#X connect 24 0 16 0;
#X connect 26 0 24 0;
#X connect 26 1 24 1;
```

```
#X connect 27 0 28 0;
#X connect 27 1 28 1;
#X connect 28 0 17 1;
#X connect 29 0 16 0;
#X connect 30 0 32 1;
#X connect 31 0 34 0;
#X connect 32 0 33 0;
#X connect 34 0 1 0;
#X connect 35 0 8 2;
#X connect 36 0 37 0;
#X connect 37 0 38 0;
#X connect 38 0 1 0;
#X connect 39 0 5 0;
#X connect 39 1 5 1;
#X connect 39 2 5 2;
#X connect 39 3 8 1;
#X connect 39 3 10 2;
#X connect 39 3 24 2;
#X connect 39 4 11 0;
#X connect 39 4 12 0;
#X connect 39 5 14 0;
#X connect 39 5 13 0;
#X connect 39 6 15 0;
#X connect 39 7 15 1;
#X connect 39 8 15 2;
#X connect 39 9 27 0;
#X connect 39 10 24 2;
#X connect 39 11 24 3;
#X connect 39 12 32 0;
#X connect 39 13 32 2;
#X connect 39 14 8 3;
#X restore 462 566 pd \$0-guts;
#X connect 0 0 2 0;
#X connect 1 0 0 0;
#X connect 3 0 0 1;
```

21 zzc/Makefile

```
all: vcf~.pd_linux

clean:
	rm -f vcf~.pd_linux

%.pd_linux: %.c
	gcc -std=c99 -Wall -pedantic -O3 -fPIC -shared -o $*.pd_linux $*.c -lm
```

22 zzc/periodoscope.pd

```
#N canvas 0 0 721 690 10;
#X obj 35 1137 #store;
#X obj 62 852 #cast i;
#X obj 63 891 #to_list;
#X obj 423 310 inlet;
#X obj 63 871 #import ( 2 );
#X obj 64 911 t a b;
#X obj 139 912 f 0;
```

```
#X obj 133 1014 Gem/hsv2rgb;
#X msg 151 872 0;
#X msg 134 988 \$1 1 1;
#X obj 63 948 unpack f f;
#X obj 65 1080 pack f f f f f;
#X obj 131 1036 unpack f f f;
#X obj 200 1059 * 255;
#X obj 155 1057 * 255;
#X obj 115 1056 * 255;
#X obj 505 310 inlet;
#X obj 111 501 #fold *;
#X obj 111 525 #to_float;
#X obj 72 552 div;
#X obj 45 461 t a a;
#X obj 45 482 #finished;
#X obj 66 574 *;
#X obj 45 531 t f f;
#X obj 45 595 ==;
#X obj 46 638 pack f f f;
#X obj 45 615 sel 1;
#X obj 63 707 #redim ( 1 );
#X obj 63 777 #autoscale;
#X obj 63 728 #transpose 0 1;
#X obj 63 753 #redim ( 1 );
#X obj 140 933 /;
#X msg 144 706 3 # \$2 \$3 2;
#X obj 137 960 wrap;
#X obj 171 912 + 1;
#X obj 132 403 t b f;
#X obj 132 362 swap;
#X obj 132 383 /;
#X obj 46 345 f;
#X msg 144 753 3 # 1 \$2 2;
#X obj 196 239 display;
#X obj 46 365 list prepend;
#X obj 45 408 list append;
#X obj 45 388 t b a;
#X obj 168 308 t b b b;
#X msg 64 1106 put_at \$1 \$2 0 \, 5 5 3 b # \$3 \$4 \$5;
#X obj 282 333 factorize;
#X obj 194 86 #many bng 8 8;
#X obj 239 431 unpack f f f;
#X obj 239 480 +;
#X obj 253 457 * 8;
#X obj 369 464 #to_float;
#X obj 369 486 t b f;
#X obj 400 507 + 1;
#X obj 369 507 f;
#X msg 413 486 0;
#X obj 408 550 div 8;
#X obj 369 550 mod 8;
#X obj 369 573 pack f f f;
#X msg 484 587 spacing 3 36;
#X obj 369 438 t a b b;
#X obj 237 498 listread;
#X msg 369 631 to \$1 \$2 label +\$3 \, to \$1 \$2 color 16 0 22;
#X obj 63 806 # * ( 1 d # 60 );
```

```
#X obj 63 827 # + ( 1 d # 78 );
#X msg 84 1136 reassign \, 160 160 3 b # 255;
#X obj 24 87 #see;
#X obj 442 664 s \$0-factorchooser-in;
#X obj 194 22 r \$0-factorchooser-in;
#X obj 35 1160 s \$0-image-in;
#X obj 24 23 r \$0-image-in;
#X obj 317 69 nbx 5 14 1 1e+37 0 0 \$0-period-out \$0-period-in period
0 -8 0 10 -228856 -1 -1 1 256;
#X obj 505 332 f;
#X obj 505 353 s \$0-period-in;
#X obj 115 475 r \$0-period-out;
#X obj 377 69 nbx 5 14 1 1e+37 0 0 \$0-outer-out \$0-outer-in outer
0 -8 0 10 -228856 -1 -1 3600 256;
#X obj 437 69 nbx 5 14 1 1e+37 0 0 \$0-inner-out \$0-inner-in inner
0 -8 0 10 -228856 -1 -1 1 256;
#X obj 127 546 s \$0-inner-in;
#X obj 93 576 s \$0-outer-in;
#X obj 282 311 r \$0-outer-out;
#X obj 72 441 s \$0-periods-in;
#X obj 179 267 r \$0-periods-in;
#X obj 484 560 loadbang;
#X msg 442 610 delegate label - \, delegate color 10 10 10;
#X obj 256 69 nbx 5 14 1 1e+37 0 0 \$0-count-out \$0-count-in count
0 -8 0 10 -4034 -1 -1 1 256;
#X obj 278 525 s \$0-count-in;
#X obj 59 323 r \$0-count-out;
#X obj 556 69 bng 14 250 50 0 \$0-reset-out \$0-reset-in reset -15
-8 0 10 -258113 -1 -1;
#X msg 168 337 1;
#X obj 168 362 s \$0-count-in;
#X obj 168 288 r \$0-reset-out;
#X obj 423 354 s \$0-data-in;
#X obj 468 425 loadbang;
#X obj 468 446 s \$0-reset-in;
#X obj 369 598 spigot;
#X obj 446 550 < 64;
#X obj 231 69 bng 14 250 50 0 \$0-ok-out \$0-ok-in ok 0 -8 0 10 -4034
-1 -1;
#X obj 202 69 tgl 14 0 \$0-auto-out \$0-auto-in auto 0 -8 0 10 -4034
-1 -1 0 1;
#X obj 234 524 t b f;
#X obj 235 555 f;
#X obj 235 579 sel 1;
#X obj 264 556 r \$0-auto-out;
#X obj 235 603 s \$0-ok-in;
#X obj 46 301 r \$0-ok-out;
#X obj 46 659 t b b b a;
#X obj 181 684 r \$0-data-in;
#X obj 170 935 r \$0-outer-out;
#X obj 200 912 r \$0-incr-out;
#X obj 159 422 s \$0-incr-in;
#X obj 496 69 nbx 5 14 1 1e+37 0 0 \$0-incr-out \$0-incr-in incr 0
-8 0 10 -228856 -1 -1 3600 256;
#X obj 120 647 print periodoscope;
#X msg 120 625 invalid internal state detected;
#X obj 424 280 s \$0-factorchooser-out;
```

```
#X obj 239 405 r \$0-factorchooser-out;
#X obj 369 529 t f f f;
#X obj 423 377 r \$0-periods-in;
#X obj 423 402 outlet;
#X obj 556 241 bng 15 250 50 0 empty empty empty 17 7 0 10 -262144
-1 -1;
#X obj 45 512 f 1;
#X obj 63 684 #store ( 1 2 d # );
#X connect 0 0 69 0;
#X connect 1 0 4 0;
#X connect 2 0 5 0;
#X connect 3 0 91 0;
#X connect 4 0 2 0;
#X connect 5 0 10 0;
#X connect 5 1 6 0;
#X connect 6 0 31 0;
#X connect 6 0 34 0;
#X connect 7 0 12 0;
#X connect 8 0 6 1;
#X connect 9 0 7 0;
#X connect 10 0 11 0;
#X connect 10 1 11 1;
#X connect 11 0 45 0;
#X connect 12 0 15 0;
#X connect 12 1 14 0;
#X connect 12 2 13 0;
#X connect 13 0 11 4;
#X connect 14 0 11 3;
#X connect 15 0 11 2;
#X connect 16 0 72 0;
#X connect 17 0 18 0;
#X connect 18 0 19 1;
#X connect 18 0 22 1;
#X connect 18 0 25 2;
#X connect 18 0 77 0;
#X connect 19 0 22 0;
#X connect 19 0 25 1;
#X connect 19 0 78 0;
#X connect 20 0 21 0;
#X connect 20 1 17 0;
#X connect 21 0 118 0;
#X connect 22 0 24 1;
#X connect 23 0 24 0;
#X connect 23 1 19 0;
#X connect 24 0 26 0;
#X connect 25 0 104 0;
#X connect 26 0 25 0;
#X connect 26 1 111 0;
#X connect 27 0 29 0;
#X connect 28 0 63 0;
#X connect 29 0 30 0;
#X connect 30 0 28 0;
#X connect 31 0 33 0;
#X connect 32 0 27 1;
#X connect 33 0 9 0;
#X connect 34 0 6 1;
#X connect 35 0 42 0;
```

```
#X connect 35 1 108 0;
#X connect 36 0 37 0;
#X connect 36 1 37 1;
#X connect 37 0 35 0;
#X connect 38 0 41 0;
#X connect 39 0 30 1;
#X connect 41 0 43 0;
#X connect 42 0 20 0;
#X connect 43 0 42 0;
#X connect 43 1 41 1;
#X connect 43 1 42 1;
#X connect 43 1 80 0;
#X connect 44 0 88 0;
#X connect 44 1 41 0;
#X connect 44 2 41 1;
#X connect 45 0 0 1;
#X connect 46 0 61 1;
#X connect 46 0 60 0;
#X connect 47 1 112 0;
#X connect 48 0 49 0;
#X connect 48 1 50 0;
#X connect 49 0 61 0;
#X connect 50 0 49 1;
#X connect 51 0 52 0;
#X connect 52 0 54 0;
#X connect 52 1 58 2;
#X connect 53 0 54 1;
#X connect 54 0 53 0;
#X connect 54 0 114 0;
#X connect 55 0 54 1;
#X connect 56 0 58 1;
#X connect 57 0 58 0;
#X connect 58 0 94 0;
#X connect 59 0 67 0;
#X connect 60 0 51 0;
#X connect 60 1 55 0;
#X connect 60 2 83 0;
#X connect 61 0 98 0;
#X connect 62 0 67 0;
#X connect 63 0 64 0;
#X connect 64 0 1 0;
#X connect 65 0 0 1;
#X connect 68 0 47 0;
#X connect 70 0 66 0;
#X connect 72 0 73 0;
#X connect 74 0 118 1;
#X connect 79 0 46 0;
#X connect 79 0 36 1;
#X connect 81 0 40 0;
#X connect 82 0 59 0;
#X connect 83 0 67 0;
#X connect 86 0 38 1;
#X connect 86 0 36 0;
#X connect 88 0 89 0;
#X connect 90 0 44 0;
#X connect 92 0 93 0;
#X connect 94 0 62 0;
```

```
#X connect 95 0 94 1;
#X connect 98 0 99 0;
#X connect 98 1 85 0;
#X connect 99 0 100 0;
#X connect 100 0 102 0;
#X connect 101 0 99 1;
#X connect 103 0 38 0;
#X connect 104 0 0 0;
#X connect 104 1 119 0;
#X connect 104 2 8 0;
#X connect 104 2 65 0;
#X connect 104 3 32 0;
#X connect 104 3 39 0;
#X connect 105 0 119 1;
#X connect 106 0 31 1;
#X connect 107 0 34 1;
#X connect 111 0 110 0;
#X connect 113 0 48 0;
#X connect 114 0 57 0;
#X connect 114 1 56 0;
#X connect 114 2 95 0;
#X connect 115 0 116 0;
#X connect 117 0 41 0;
#X connect 118 0 23 0;
#X connect 119 0 27 0;
#X coords 0 -1 1 1 560 210 1 20 50;
```

23 zzc/periods.pd

```
#N canvas 32 98 544 443 10;
#X obj 11 11 cnv 15 511 254 empty empty periods 4 12 0 24 -203904 -16662
0;
#X obj 15 37 #see;
#X obj 16 275 r \$0-image-r;
#X obj 24 336 inlet;
#N canvas 193 139 954 465 \$0-guts 0;
#X obj 16 16 inlet;
#X obj 285 36 inlet;
#X obj 469 93 s \$0-image-r;
#X msg 467 66 192 192 3 # 255;
#X obj 341 11 loadbang;
#X obj 368 93 s \$0-factors-r;
#X msg 368 61 spacing 3 32;
#X obj 16 85 s \$0-reset-r;
#X obj 52 111 s \$0-auto-r;
#X obj 98 84 s \$0-ok-r;
#X obj 131 109 s \$0-count-r;
#X obj 166 82 s \$0-resend-r;
#X obj 341 34 t b b b;
#X obj 11 304 r \$0-reset-s;
#X obj 106 302 r \$0-ok-s;
#X obj 175 315 r \$0-resend-s;
#X obj 54 331 r \$0-auto-s;
#X obj 140 356 r \$0-count-s;
#X msg 11 354 reset;
#X msg 54 353 auto \$1;
#X msg 106 354 ok;
```

```
#X msg 140 383 count \$1;
#X msg 175 336 resend;
#X obj 16 58 route reset auto ok count resend;
#X obj 285 83 s \$0-data-r;
#X obj 250 1007 #store;
#X obj 277 722 #cast i;
#X obj 278 761 #to_list;
#X obj 278 741 #import ( 2 );
#X obj 279 781 t a b;
#X obj 354 782 f 0;
#X obj 348 884 Gem/hsv2rgb;
#X msg 366 742 0;
#X msg 349 858 \$1 1 1;
#X obj 278 818 unpack f f;
#X obj 280 950 pack f f f f f;
#X obj 346 906 unpack f f f;
#X obj 415 929 * 255;
#X obj 370 927 * 255;
#X obj 330 926 * 255;
#X obj 326 371 #fold *;
#X obj 326 395 #to_float;
#X obj 287 422 div;
#X obj 260 331 t a a;
#X obj 260 352 #finished;
#X obj 281 444 *;
#X obj 260 401 t f f;
#X obj 260 465 ==;
#X obj 261 508 pack f f f;
#X obj 260 485 sel 1;
#X obj 278 577 #redim ( 1 );
#X obj 278 598 #transpose 0 1;
#X obj 278 623 #redim ( 1 );
#X obj 355 803 /;
#X msg 359 576 3 # \$2 \$3 2;
#X obj 352 830 wrap;
#X obj 386 782 + 1;
#X obj 347 273 t b f;
#X obj 347 232 swap;
#X obj 347 253 /;
#X obj 261 215 f;
#X msg 359 623 3 # 1 \$2 2;
#X obj 261 235 list prepend;
#X obj 260 278 list append;
#X obj 260 258 t b a;
#X obj 383 178 t b b b;
#X msg 279 976 put_at \$1 \$2 0 \, 5 5 3 b # \$3 \$4 \$5;
#X obj 454 301 unpack f f f;
#X obj 454 350 +;
#X obj 584 334 #to_float;
#X obj 584 356 t b f;
#X obj 615 377 + 1;
#X obj 584 377 f;
#X msg 628 356 0;
#X obj 584 443 pack f f f;
#X obj 584 308 t a b b;
#X obj 452 368 listread;
#X msg 584 501 to \$1 \$2 label +\$3 \, to \$1 \$2 color 16 0 22;
```

```
#X msg 657 480 delegate label - \, delegate color 10 10 10;
#X msg 383 207 1;
#X obj 341 120 s \$0-reset-in;
#X obj 584 468 spigot;
#X obj 449 394 t b f;
#X obj 450 425 f;
#X obj 450 449 sel 1;
#X obj 261 529 t b b b a;
#X obj 584 399 t f f f;
#X obj 260 382 f 1;
#X obj 278 554 #store ( 1 2 d # );
#X obj 497 203 factorize;
#X obj 278 647 #autoscale;
#X obj 335 517 print periods;
#X msg 335 495 invalid internal state detected \$1;
#X obj 261 171 r \$0-ok-s;
#X obj 274 193 r \$0-count-s;
#X obj 383 158 r \$0-reset-s;
#X obj 383 232 s \$0-count-r;
#X obj 497 181 r \$0-outer-s;
#X obj 657 534 s \$0-factors-r;
#X obj 672 420 < 60;
#X obj 493 395 s \$0-count-r;
#X obj 479 426 r \$0-auto-s;
#X obj 454 275 r \$0-factors-s;
#X obj 374 292 s \$0-incr-r;
#X obj 287 311 s \$0-cascade-r;
#X obj 342 416 s \$0-inner-r;
#X obj 308 446 s \$0-outer-r;
#X obj 450 473 s \$0-ok-r;
#X obj 396 554 r \$0-data-r;
#X obj 415 782 r \$0-incr-s;
#X obj 385 805 r \$0-outer-s;
#X obj 250 1030 s \$0-image-r;
#X obj 54 411 r \$0-cascade-r;
#X obj 54 432 list prepend cascade;
#X obj 278 676 # * ( 1 d # 90 );
#X obj 278 697 # + ( 1 d # 94 );
#X msg 299 1006 reassign \, 192 192 3 b # 255;
#X obj 131 260 s \$0-total-r;
#X obj 330 345 r \$0-total-s;
#X obj 131 173 route periods;
#X obj 131 195 list split 1;
#X obj 131 226 max 1;
#X obj 10 487 outlet;
#X obj 54 453 list trim;
#X obj 131 455 r \$0-total-s;
#X msg 131 478 total \$1;
#X obj 629 419 div 12;
#X obj 584 420 mod 12;
#X obj 468 327 * 12;
#X connect 0 0 23 0;
#X connect 1 0 24 0;
#X connect 3 0 2 0;
#X connect 4 0 12 0;
#X connect 6 0 5 0;
#X connect 12 0 80 0;
```

```
#X connect 12 1 6 0;
#X connect 12 2 3 0;
#X connect 13 0 18 0;
#X connect 14 0 20 0;
#X connect 15 0 22 0;
#X connect 15 0 62 0;
#X connect 16 0 19 0;
#X connect 17 0 21 0;
#X connect 18 0 122 0;
#X connect 19 0 122 0;
#X connect 20 0 122 0;
#X connect 21 0 122 0;
#X connect 22 0 122 0;
#X connect 23 0 7 0;
#X connect 23 1 8 0;
#X connect 23 2 9 0;
#X connect 23 3 10 0;
#X connect 23 4 11 0;
#X connect 23 5 119 0;
#X connect 25 0 111 0;
#X connect 26 0 28 0;
#X connect 27 0 29 0;
#X connect 28 0 27 0;
#X connect 29 0 34 0;
#X connect 29 1 30 0;
#X connect 30 0 53 0;
#X connect 30 0 56 0;
#X connect 31 0 36 0;
#X connect 32 0 30 1;
#X connect 33 0 31 0;
#X connect 34 0 35 0;
#X connect 34 1 35 1;
#X connect 35 0 66 0;
#X connect 36 0 39 0;
#X connect 36 1 38 0;
#X connect 36 2 37 0;
#X connect 37 0 35 4;
#X connect 38 0 35 3;
#X connect 39 0 35 2;
#X connect 40 0 41 0;
#X connect 41 0 42 1;
#X connect 41 0 45 1;
#X connect 41 0 48 2;
#X connect 41 0 105 0;
#X connect 42 0 45 0;
#X connect 42 0 48 1;
#X connect 42 0 106 0;
#X connect 43 0 44 0;
#X connect 43 1 40 0;
#X connect 44 0 87 0;
#X connect 45 0 47 1;
#X connect 46 0 47 0;
#X connect 46 1 42 0;
#X connect 47 0 49 0;
#X connect 48 0 85 0;
#X connect 49 0 48 0;
#X connect 49 1 92 0;
```

```
#X connect 50 0 51 0;
#X connect 51 0 52 0;
#X connect 52 0 90 0;
#X connect 53 0 55 0;
#X connect 54 0 50 1;
#X connect 55 0 33 0;
#X connect 56 0 30 1;
#X connect 57 0 63 0;
#X connect 57 1 103 0;
#X connect 58 0 59 0;
#X connect 58 1 59 1;
#X connect 59 0 57 0;
#X connect 60 0 62 0;
#X connect 61 0 52 1;
#X connect 62 0 64 0;
#X connect 63 0 43 0;
#X connect 64 0 63 0;
#X connect 64 1 62 1;
#X connect 64 1 63 1;
#X connect 64 1 104 0;
#X connect 65 0 79 0;
#X connect 65 1 62 0;
#X connect 65 2 62 1;
#X connect 66 0 25 1;
#X connect 67 0 68 0;
#X connect 67 1 128 0;
#X connect 68 0 76 0;
#X connect 69 0 70 0;
#X connect 70 0 72 0;
#X connect 70 1 74 2;
#X connect 71 0 72 1;
#X connect 72 0 71 0;
#X connect 72 0 86 0;
#X connect 73 0 72 1;
#X connect 74 0 81 0;
#X connect 75 0 69 0;
#X connect 75 1 73 0;
#X connect 75 2 78 0;
#X connect 76 0 82 0;
#X connect 77 0 98 0;
#X connect 78 0 98 0;
#X connect 79 0 96 0;
#X connect 81 0 77 0;
#X connect 82 0 83 0;
#X connect 82 1 100 0;
#X connect 83 0 84 0;
#X connect 84 0 107 0;
#X connect 85 0 25 0;
#X connect 85 1 88 0;
#X connect 85 2 32 0;
#X connect 85 2 116 0;
#X connect 85 3 54 0;
#X connect 85 3 61 0;
#X connect 86 0 127 0;
#X connect 86 1 126 0;
#X connect 86 2 99 0;
#X connect 87 0 46 0;
```

```
#X connect 88 0 50 0;
#X connect 89 0 76 1;
#X connect 89 0 75 0;
#X connect 90 0 114 0;
#X connect 92 0 91 0;
#X connect 93 0 60 0;
#X connect 94 0 60 1;
#X connect 94 0 58 0;
#X connect 95 0 65 0;
#X connect 97 0 58 1;
#X connect 97 0 89 0;
#X connect 99 0 81 1;
#X connect 101 0 83 1;
#X connect 102 0 67 0;
#X connect 108 0 88 1;
#X connect 109 0 56 1;
#X connect 110 0 53 1;
#X connect 112 0 113 0;
#X connect 113 0 123 0;
#X connect 114 0 115 0;
#X connect 115 0 26 0;
#X connect 116 0 25 1;
#X connect 118 0 87 1;
#X connect 119 0 120 0;
#X connect 120 0 121 0;
#X connect 121 0 117 0;
#X connect 123 0 122 0;
#X connect 124 0 125 0;
#X connect 125 0 122 0;
#X connect 126 0 74 1;
#X connect 127 0 74 0;
#X connect 128 0 68 1;
#X restore 24 357 pd \$0-guts;
#X obj 81 337 inlet;
#X obj 24 380 outlet;
#X obj 137 15 bng 18 250 50 0 \$0-reset-s \$0-reset-r reset 24 9 0
16 -4034 -13381 -16662;
#X obj 437 19 tgl 18 0 \$0-auto-s \$0-auto-r auto 24 9 0 16 -4034 -13381
-16662 0 1;
#X obj 437 79 nbx 5 18 1 1e+37 0 0 \$0-count-s \$0-count-r count 0
-8 0 16 -4034 -13381 -16662 3 256;
#X obj 438 44 bng 18 250 50 0 \$0-ok-s \$0-ok-r ok 24 9 0 16 -4034
-13381 -16662;
#X obj 218 244 display;
#X obj 219 272 r \$0-cascade-r;
#X obj 130 242 cnv 15 85 18 empty empty cascade 4 9 0 18 -203904 -16662
0;
#X obj 436 114 nbx 5 18 1 1e+37 0 0 \$0-total-s \$0-total-r total 0
-8 0 16 -262130 -16662 -16662 1260 256;
#X obj 436 150 nbx 5 18 1 1e+37 0 0 \$0-outer-s \$0-outer-r outer 0
-8 0 16 -262130 -16662 -16662 1 256;
#X obj 436 186 nbx 5 18 1 1e+37 0 0 \$0-inner-s \$0-inner-r inner 0
-8 0 16 -262130 -16662 -16662 1260 256;
#X obj 436 220 nbx 5 18 1 1e+37 0 0 \$0-incr-s \$0-incr-r incr 0 -8
0 16 -262130 -16662 -16662 1 256;
#X obj 17 243 bng 18 250 50 0 \$0-resend-s \$0-resend-r resend 24 9
0 16 -4034 -13381 -16662;
```

```
#X obj 118 274 r \$0-factors-r;
#X obj 217 17 #many bng 12 5;
#X obj 421 273 s \$0-factors-s;
#X connect 2 0 1 0;
#X connect 3 0 4 0;
#X connect 4 0 6 0;
#X connect 5 0 4 1;
#X connect 12 0 11 0;
#X connect 19 0 20 0;
#X connect 20 1 21 0;
#X coords 0 -1 1 1 513 256 2 10 10;
```

24 zzc/presets.pd

```
#N canvas 0 0 450 300 10;
#X obj 11 11 cnv 15 318 92 empty empty presets 4 12 0 24 -203904 -16662
0;
#X obj 23 36 bng 15 250 50 0 \$0-preset-01-s \$0-preset-01-r The_Military-↙
    ↳ Industrial_Complex
20 7 0 12 -4034 -13381 -16662;
#X obj 43 52 bng 15 250 50 0 \$0-preset-02-s \$0-preset-02-r ↙
    ↳ After_The_Despoilment
20 7 0 12 -4034 -13381 -16662;
#X obj 63 68 bng 15 250 50 0 \$0-preset-03-s \$0-preset-03-r Race_Against_Time
20 7 0 12 -4034 -13381 -16662;
#X obj 83 84 bng 15 250 50 0 \$0-preset-04-s \$0-preset-04-r Before_Civilisation
20 7 0 12 -4034 -13381 -16662;
#X obj 17 122 inlet;
#N canvas 0 0 450 408 \$0-guts 0;
#X obj 24 18 inlet;
#X obj 24 39 route preset;
#X obj 24 62 sel 1 2 3 4;
#X obj 191 181 outlet;
#X msg 191 99 preset 1;
#X msg 201 119 preset 2;
#X msg 211 139 preset 3;
#X msg 221 159 preset 4;
#X obj 24 84 s \$0-preset-01-r;
#X obj 34 104 s \$0-preset-02-r;
#X obj 44 124 s \$0-preset-03-r;
#X obj 54 144 s \$0-preset-04-r;
#X obj 194 14 r \$0-preset-01-s;
#X obj 204 34 r \$0-preset-02-s;
#X obj 214 54 r \$0-preset-03-s;
#X obj 224 74 r \$0-preset-04-s;
#X connect 0 0 1 0;
#X connect 1 0 2 0;
#X connect 2 0 8 0;
#X connect 2 1 9 0;
#X connect 2 2 10 0;
#X connect 2 3 11 0;
#X connect 4 0 3 0;
#X connect 5 0 3 0;
#X connect 6 0 3 0;
#X connect 7 0 3 0;
#X connect 12 0 4 0;
#X connect 13 0 5 0;
```

```
#X connect 14 0 6 0;
#X connect 15 0 7 0;
#X restore 17 143 pd \$0-guts;
#X obj 17 168 outlet;
#X connect 5 0 6 0;
#X connect 6 0 7 0;
#X coords 0 -1 1 1 320 94 2 10 10;
```

25 zzc/start.sh

```
#!/bin/sh
pd \
  -nrt \
  -jack -channels 2 -r 48000 \
  -path gridflow/abstractions \
  -helppath gridflow/doc/flow_classes \
  -lib vcf~:zexy:Gem:gridflow \
  -stderr -verbose \
  -open main.pd
```

26 zzc/timing.pd

```
#N canvas 0 0 450 300 10;
#X obj 11 11 cnv 15 190 92 empty empty timing 4 12 0 24 -203904 -16662
0;
#N canvas 0 0 450 519 \$0-guts 0;
#X obj 27 47 inlet;
#X obj 27 68 route runtime rectime;
#X obj 27 134 / 1000;
#X obj 27 155 t f f;
#X obj 9 184 div 60;
#X obj 59 185 mod 60;
#X obj 107 185 mod 1000;
#X obj 9 211 s \$0-run-m-r;
#X obj 59 235 s \$0-run-s-r;
#X obj 107 212 s \$0-run-ms-r;
#X obj 217 134 / 1000;
#X obj 217 155 t f f;
#X obj 199 184 div 60;
#X obj 249 185 mod 60;
#X obj 297 185 mod 1000;
#X obj 199 211 s \$0-rec-m-r;
#X obj 249 235 s \$0-rec-s-r;
#X obj 297 212 s \$0-rec-ms-r;
#X obj 24 388 outlet;
#X obj 24 310 r \$0-record-s;
#X msg 24 331 record;
#X connect 0 0 1 0;
#X connect 1 0 2 0;
#X connect 1 0 6 0;
#X connect 1 1 10 0;
#X connect 1 1 14 0;
#X connect 2 0 3 0;
#X connect 3 0 4 0;
#X connect 3 1 5 0;
#X connect 4 0 7 0;
```

```
#X connect 5 0 8 0;
#X connect 6 0 9 0;
#X connect 10 0 11 0;
#X connect 11 0 12 0;
#X connect 11 1 13 0;
#X connect 12 0 15 0;
#X connect 13 0 16 0;
#X connect 14 0 17 0;
#X connect 19 0 20 0;
#X connect 20 0 18 0;
#X restore 57 135 pd \$0-guts;
#X obj 57 114 inlet;
#X obj 57 156 outlet;
#X obj 112 17 bng 18 250 50 0 \$0-record-s \$0-record-r record 24 9
0 16 -4034 -13381 -16662;
#X obj 15 54 nbx 3 18 -1e+37 1e+37 0 0 \$0-run-m-s \$0-run-m-r mins
0 -8 0 16 -262130 -16662 -16662 0 256;
#X obj 73 54 nbx 3 18 -1e+37 1e+37 0 0 \$0-run-s-s \$0-run-s-r secs
0 -8 0 16 -262130 -16662 -16662 0 256;
#X obj 131 54 nbx 4 18 -1e+37 1e+37 0 0 \$0-run-ms-s \$0-run-ms-r msecs
0 -8 0 16 -262130 -16662 -16662 0 256;
#X obj 15 80 nbx 3 18 -1e+37 1e+37 0 0 \$0-rec-m-s \$0-rec-m-r empty
0 -8 0 16 -262130 -16662 -16662 0 256;
#X obj 73 80 nbx 3 18 -1e+37 1e+37 0 0 \$0-rec-s-s \$0-rec-s-r empty
0 -8 0 16 -262130 -16662 -16662 0 256;
#X obj 131 80 nbx 4 18 -1e+37 1e+37 0 0 \$0-rec-ms-s \$0-rec-ms-r empty
0 -8 0 16 -262130 -16662 -16662 0 256;
#X connect 1 0 3 0;
#X connect 2 0 1 0;
#X coords 0 -1 1 1 192 94 2 10 10;
```

27 zzc/vcf~.c

```
#include <math.h>
#include <m_pd.h>

typedef struct vcfctl
{
    double c_re;
    double c_im;
    double c_q;
    double c_isr;
} t_vcfctl;

typedef struct sigvcf
{
    t_object x_obj;
    t_vcfctl x_cspace;
    t_vcfctl *x_ctl;
    float x_f;
} t_sigvcf;

t_class *sigvcf_class;

static void *sigvcf_new(t_floatarg q)
{
    t_sigvcf *x = (t_sigvcf *)pd_new(sigvcf_class);
```

```
25      inlet_new(&x->x_obj, &x->x_obj.ob_pd, &s_signal, &s_signal);
        inlet_new(&x->x_obj, &x->x_obj.ob_pd, gensym("float"), gensym("ft1"));
        outlet_new(&x->x_obj, gensym("signal"));
        outlet_new(&x->x_obj, gensym("signal"));
        x->x_ctl = &x->x_cspace;
30      x->x_cspace.c_re = 0;
        x->x_cspace.c_im = 0;
        x->x_cspace.c_q = q;
        x->x_cspace.c_isr = 0;
        x->x_f = 0;
35      return (x);
    }

    static void sigvcf_ft1(t_sigvcf *x, t_floatarg f)
    {
40      x->x_ctl->c_q = (f > 0 ? f : 0.f);
    }

    static t_int *sigvcf_perform(t_int *w)
    {
45      float *in1 = (float *)(w[1]);
        float *in2 = (float *)(w[2]);
        float *out1 = (float *)(w[3]);
        float *out2 = (float *)(w[4]);
        t_vcfctl *c = (t_vcfctl *)(w[5]);
50      int n = (t_int)(w[6]);
        int i;
        double re = c->c_re, re2;
        double im = c->c_im;
        double q = c->c_q;
55      double qinv = (q > 0? 1.0f/q : 0);
        double ampcorrect = 2.0f - 2.0f / (q + 2.0f);
        double isr = c->c_isr;
        double coefr, coefi;
        double f1;
60      for (i = 0; i < n; i++)
        {
            double cf, r, oneminusr;
            cf = *in2++ * isr;
            if (cf < 0) cf = 0;
65          r = (qinv > 0 ? 1 - cf * qinv : 0);
            if (r < 0) r = 0;
            oneminusr = 1.0f - r;
            coefr = r * cos(cf);
            coefi = r * sin(cf);
70          f1 = *in1++;
            re2 = re;
            *out1++ = re = ampcorrect * oneminusr * f1
                + coefr * re2 - coefi * im;
            *out2++ = im = coefi * re2 + coefr * im;
75      }
        c->c_re = re;
        c->c_im = im;
        return (w+7);
    }
80
    static void sigvcf_dsp(t_sigvcf *x, t_signal **sp)
```

```
{
    x->x_ctl->c_isr = 6.28318f/sp[0]->s_sr;
    dsp_add(sigvcf_perform, 6,
        sp[0]->s_vec, sp[1]->s_vec, sp[2]->s_vec, sp[3]->s_vec,
            x->x_ctl, sp[0]->s_n);
}

void vcf_tilde_setup(void)
{
    sigvcf_class = class_new(gensym("vcf~"), (t_newmethod)sigvcf_new, 0,
        sizeof(t_sigvcf), 0, A_DEFFLOAT, 0);
    CLASS_MAINSIGNALIN(sigvcf_class, t_sigvcf, x_f);
    class_addmethod(sigvcf_class, (t_method)sigvcf_dsp, gensym("dsp"), 0);
    class_addmethod(sigvcf_class, (t_method)sigvcf_ft1,
        gensym("ft1"), A_FLOAT, 0);
}
```