



Composers' ✕ Desktop ✕ Project

CDP SYNTH Functions

(with Command Line Usage)

Functions to SYNTHESISE (Generate) soundfiles

(Names in brackets mean that these are separate programs. The others are sub-modules of SYNTH.)

CHORD

Generate a chord with a simple waveform

CLICKS

Create a click track from tempo, meter and barring data

[NEWSYNTH SYNTHESIS]

Generate complex spectra from fundamental and partial balance information in one or more textfiles

NOISE

Generate noise

SILENCE

Make silent soundfile

SPECTRA

Generate both channels of a stereo spectral band

WAVE

Generate simple waveforms

For more comprehensive synthesis facilities, users are recommended to explore the public domain program **Csound** (formerly distributed with the CDP system).



SYNTH CHORD – Generate a chord with a simple waveform

Usage

synth chord mode *outfile datafile sr chans dur* [-**aamp**] [-**ttabsize**]

Modes

- 1 *Datafile* contains a list of MIDI pitch values
- 2 *Datafile* contains a list of frequencies in Hz

Parameters

outfile – output file with synthesised chord

datafile – contains the MIDI pitch values or frequencies of the chord to be produced
sr – Sample Rate: may be 96000, 88200, 48000, 24000, 44100, 22050, 32000, or 16000

chans – number of channels (1-16)

dur – duration of the output sound, in seconds

-aamp – amplitude of the output sound (0.0 to 1.0). 1.0 is maximum amplitude, and the Default value.

-ttabsize – the size of the table storing the waveform. It defaults to 4096: the input value is always rounded to a multiple of 4.

Understanding the SYNTH CHORD Process

All notes of the chord produced begin and end at the same time. A pure sine tone is used to generate the chord.

Musical Applications

This program was written to provide a way of generating chordal sounds for the Tonal Harmony Workshop of the Musical Testbed in *Sound Loom*. It is not expected to be used as sonic source material, but given the versatility of the CDP software, the unexpected could happen. For example, it could provide a way to create synthetic mechanical sounds, further developed with functions such as wavecycle distortion, downwards transposition etc.: e.g., to make a background hum for a generator or spaceship.

End of SYNTH CHORD



SYNTH CLICKS – Create a click track from tempo, meter and barring data

Usage

synth clicks mode *outname clickfile* [-sstart -eend -zzero]

Modes

- 1 *Start* and *end* are **times**
- 2 *Start* and *end* are **data line numbers**

Parameters

outname – output soundfile click track

clickfile – contains a numbered sequence of data lines followed by 2, 3 or 4 data items, separated by spaces. The click track goes from *start* to *end*, with the music (optionally) starting at data line zero. Format options start with a line number followed by 2 data items (first format option), OR a line number followed by 3 or 4 data items (second format option) – the 3 data item format omits specifying an accent pattern. See **Format of the Clickfile**, below.

[-sstart] – (optional) start time of click track

[-eend] – (optional) end time of click track

[-zzero] – option to start music at line zero

Format of the Clickfile:

- **Time and Pauses format** – The following items are placed on each (numbered) data line:
 - *time* – (jump to) a time point, such as 141.52 sec. It must be later than the previous event. E.g., **1 TIME 141.52** (Note the line number. An equals sign is not used.)
 - *GP* – initiate a General Pause of specified duration, with or without an accent at the beginning: '0' for no accent and '1' for an accent). E.g., **2 GP 1 23.7** (Line 2, General Pause of 23.7 seconds, starting with an accent – heard as a click.)

Example:

```
1 TIME 10 [Line 1, time set to 10 seconds]
2 GP 1 12 [Line 2, General Pause of 12 seconds, with an accent]
```



- **Fully specified Tempo and Barring** – The following items are placed on each (numbered) data line:
 1. *tempo* – in either of these two forms, with no spaces around the equals sign:
 - **1=144** – '1' signifies a crotchet (quarter-note) and '144' signifies the tempo MM=144. Thus '0.5' signifies a quaver (eighth-note) and '1.5' a dotted crotchet, etc.
 - **0.5=144to85.3** – this specifies a gradual tempo change from quaver=144 to quaver=85.3.
 2. *barring* – this takes the form 4:4 or 6:8 or 7:16 etc., with no spaces around the colon (':').
 3. *count* – this is the integer (whole- number) number of bars in this format.
 4. [*accent pattern*] – optional: may be omitted. The format for this uses numbers and dots, as follows:
 - **1.....** – a strong beat followed by 5 weak beats
 - **1..1..** – a strong beat (start) followed by a secondary beat (at 4), weak inbetween
 - **100100** – strong and secondary beats, but *no* intermediate beats

Example: (Explanations in square brackets are *not* part of the *clickfile*)

```

1 1=60 3:4 4 1.1 [Line 1, MM crotchet=60,
3 crotchets per bar (4/4), 4 bars, with a main accent on
the first beat and a secondary accent on the third beat]
2 0.5=90 6:8 4 100100 [Line 2, MM quaver=90,
6 quavers per bar (6/8), 4 bars, with a main accent on the
first beat and a secondary accent on the fourth beat, and no
intermediate accents]
3 0.5=90to60 6:8 4 1..1.. [Line 3, MM changes
from quaver=90 to quaver=60, 4 bars, with accents on first and
fourth as in data line 1 and weaker intermediate beats]

```

Accentuation defaults: If the accent pattern is omitted, default patterns are used. These comprise:

- a. an accent on the start of each bar plus secondary accents every 3rd beat in: 6:8, 9:8, 12:8, 15:8 etc. OR 6:4, 6:16, 6:32, 9:4, 9:16, etc.
- b. an accent on the start of each bar, with *no* secondary accents in *all other meters*.

Understanding the SYNTH CLICKS Function

SYNTH CLICKS makes a click track from *start* to *end*, the music starting at dataline 1, unless 'start at dataline 0' is selected.

Provision is made for jumps to time points, General Pauses, tempo, acceleration and deceleration (i.e., gradual change from one tempo to another), barring (i.e., meters) and accent patterns.

End of SYNTH CLICKS



NEWSYNTH SYNTHESIS – Generate complex spectra from fundamental and partial balance information in one or more textfiles

Usage

newsynth synthesis 1 *outsndfile spectrum srate dur frq*
newsynth synthesis 2 *outsndfile spectrum srate dur frq [-nnarrowing] [-ccentring]*
newsynth synthesis 3 *outsndfile spectrum srate dur frq chns maxrange rate [-urise] [-dfall][fsteady][-sssplice] [-nN] [-a] [-z] [-x] [-tspacetype] [-rrotspeed] [[-m] [-j] [-efrom -Etime] [-cto -Ctime]]*

Example command line to create complex spectra:

```
newsynth synthesis 1 outsndfile spectrum 44100 5.0 110
```

Modes

- 1 Generates tones with any number of (possibly varying) partials
- 2 Generates wave-packet streams with any number of (possibly varying) partials
- 3 Multi-channel mode in which partials spread over *N* octaves fade in and out randomly

Parameters

All Modes:

outsndfile – output soundfile generated by the data

spectrum – textfile listing partial ratios and their relative levels, against time:

- The data is a text file of lines of data.
- Every line must have the same number of entries.
- The first entry on each line is a time.
- Times must start at zero and increase.
- All **even numbered entries are partial numbers**.
- For tone-generation, the first partial number on each line must be 1.
- Partial numbers must increase from entry to entry.
- All other **odd numbered entries are partial levels**.
- Levels may have any value.
- Negative values invert the phase of the partial.

srate – the sample rate of the resultant synthesised sound

dur – the duration of the resultant synthesised sound

frq – the (possibly time-varying) fundamental frequency of the output

Add for Mode 2:

-nnarrowing – narrowing of the packet envelope (Range: 0 to 1000).

- Values below 1.0 broaden the packet.
- Values very close to zero may produce clicks because the envelope approaches a square-wave shape.



- Very high values with very high frequencies may produce click-impulses or silence.
- ccentring** – position of the centre of the packet envelope's peak:
 - **0** places the peak at the centre
 - **-1** places it at the start, and
 - **1** places it at the end.

Add for Mode 3:

chns – the number of output channels

maxrange – the maximum range of the transposition of spectral components: specify as number of whole octaves

rate – the average time between changes to the partial content of the output

-**urise** – time over which to expand to the maximum range

-**dfall** – time over which to return to the initial range, before the end is reached

-**fsteady** the duration of steady-stage sound at the end–

-**ssplice** – splice-width in milliseconds for partial entry and exit

-**nN** – *N* is the same fixed number of partials that will be chosen for each event

-**tspacetype** – the type of output spatialisation

-**rrotspeed** – the rotation speed (for certain spatialisation types)

-**a** – activate an initial increase in the number of partials starting from only the fundamental

-**z** – activate a decrease in the number of partials as the sound moves from steady-state back to the fundamental

-**x** – (Xclusive): change all partials (as far as possible) from event to event

-**m** – (Move): distribute partials in space

-**j** – (Jump): all partials are assigned to the same location for any one event

-**efrom** (Emerge): the sound emerges from channel *from*, used with

-**Etime** – (Emerge): the *time* from the beginning of the sound over which the sound emerges

-**cto** – (Converge): the sound converges to channel *to*, used with

-**Ctime** – (Converge): the *time* from the end over which the sound converges

NB: flags **-j**, **-e**, **-E**, **-c** and **-C** are only operational if **-m** is set. They work together as an optional set of parameters, but note the following:

NB: flags with NO parameters must be placed on the command line AFTER any flags WITH parameters.

Understanding the NEWSYNTH Process

Modes **1** and **2** generate a waveform based on a pre-defined and possibly time-varying spectrum.

The partial number can be harmonic (e.g. 2) or inharmonic (e.g. 2.1), except for the fundamental, which must be 1.

- Mode **2** further allows a narrowing of each wave-cycle envelope (too high a value produces silence), which ultimately reduces to clicks, and a centring of this envelope's peak on the start, middle or end of each cycle.
- Mode **3** produces a multi-channel output, and its partial content changes by random transposition every *rate* seconds, within *maxrange*, with further parameters to control this change.

Care should be taken, especially with middle to high fundamental frequencies, that no partial is specified which would be above the Nyquist limit (22050 Hz).

End of NEWSYNTH



SYNTH NOISE – Generate noise

Usage

synth noise *outfile sample_rate channels duration* [-**aamplitude**] [-**f**]

Parameters

outfile – noise soundfile generated
sample_rate – sample rate of *outfile*

Standard sample rates supported:
 96000, 88200, 48000, 44100, 32000, 24000, 22050, 16000, 11025, or
 8000
 rates below 8000 are not supported – a warning is issued for other rates

channels – number of channels in *outfile* (1-16)
duration – duration of *outfile* in seconds
 -**aamplitude** – amplitude of *outfile* (0.0 < Range < 1.0 max & default)

Amplitude may vary over time

-**f** – write soundfile as 32-bit floats (Default: 16-bit shorts)

Understanding the SYNTH NOISE Process

This process creates a single type of white noise. Variants on this noise can be achieved with filtering.

Musical Applications

You are recommended not to just take the noise file as it comes. A bit of pre-processing will enhance results when making use of it for music composition. Here are a few ideas.

Noise can be used as a complex sound source. With a rapidly changing amplitude shape, it may be a useful source for various subtractive forms of sound transformation, such as [HILITE TRACE](#) or [HILITE BLTR](#). It can add grit, such as by being a component in [COMBINE INTERLEAVE](#) (short bursts of sound alternating with other sonic material), or by being mixed at a discrete amplitude level with another sound. [COMBINE MAX](#) is an interesting way to mingle noise with another sound.

Noise can be very uniform, in spite of its complex waveform. So some work to make it more changeable might be in order before expecting very much to happen when subjecting the material to other transformations. Given amplitude variations, these may be enhanced with [ENVEL WARP](#) functions, and any spectral envelope present can be exaggerated with [FOCUS EXAG](#).

End of SYNTH NOISE



SYNTH SILENCE – Make a silent soundfile

Usage

synth silence *outfile* *sample_rate* *channels* *duration* [-f]

Parameters

outfile – silent soundfile generated
sample_rate – sample rate of *outfile*

Standard sample rates supported:

96000, 88200, 48000, 44100, 32000, 24000, 22050, 16000, 11025, or
8000

rates below 8000 are not supported – a warning is issued for other rates

channels – number of channels in *outfile* (1-16)

duration – duration of *outfile* in seconds

-f – write soundfile as 32-bit floats (Default: 16-bit shorts)

Understanding the SYNTH SILENCE Process

All the samples of a silent soundfile have zero amplitude.

Musical Applications

Silent soundfiles can be used to enforce pauses before, between, or after other sonic material. For example, a 2 second pause may be wanted between a list of soundfiles. This is done with the splicing function, SFEDIT JOIN, with a silent soundfile of 2 seconds duration spliced between each of the other soundfiles.

When specifying the sample rate, remember that it needs to be consistent with your other soundfiles so that it can be used in a mix.

Similarly, splicing with stereo soundfiles requires a stereo silent file.

End of SYNTH SILENCE



SYNTH SPECTRA – Generate both channels of a stereo spectral band

Usage

synth spectra *outfile* *dur* *frq* *spread* *max-foc* *min-foc* *timevar* [-p]

Parameters

outfile – generates output files named 'outfile0' and 'outfile1'
dur – duration of output sound, in seconds
frq – centre frequency of the band
spread – width of band in Hz (default) or as transposition ratio
min-foc – minimum degree to which the band energy is focused on (concentrated at) the centre frequency (Range: 0.001 to 1000)
max-foc – maximum degree to which the band energy is focused on (concentrated at) the centre frequency (Range: 0.001 to 1000)
timevar – degree to which the band varies over time (Range: 0 to 1)
 -p – if this flag is set, *spread* is a transposition ratio

Understanding the SYNTH SPECTRA Process

SYNTH SPECTRA generates tones which can be spread across a bandwidth and be focused to different degrees on a specified pitch. This pitch focus may be made to wander around randomly within the band by the *timevary* parameter.

Focus (*min-* and *max-*) is how tightly the energy is focused around the frequency you specify.

- If **focus** is very high, you can get almost a pure tone (with a narrow bandwidth – *spread*, and *timevar* at 0).
- With a low **focus**, the band is broader, less focused.
- With different values, the **focus** wobbles, so the sound is somewhat coloured.

The *timevar* parameter can be used to move the point of focus in the frequency band.

- When *timevar* is set at 1, e.g., with a highly focused band, the point of focus will move randomly around the bandwidth. Thus the pitch will be highly focused (very little noise component) but wobble randomly within the band.
- When *timevar* is set to 0, the focus will adopt some position in the band and stay there without moving.
- Intermediate values for *timevar* produce results inbetween these two extremes.

Musical Applications

If you want to get as close as possible to a pure tone, with no warble, set *min-foc* and *max-foc* to 1000.

End of SYNTH SPECTRA



SYNTH WAVE – Generate synthetic waveforms

Usage

synth wave mode *outfile* *sample_rate* *channels* *duration* *frequency* [-**a***amplitude*] [-**t***tablesiz*e] [-**f**]

Modes

- 1 Sine wave
- 2 Square wave
- 3 Triangle wave
- 4 Ramp (sawtooth) wave

Parameters

outfile – soundfile generated
sample_rate – sample rate of *outfile*

Standard sample rates supported:
 96000, 88200, 48000, 44100, 32000, 24000, 22050, 16000, 11025, or
 8000
 rates below 8000 are not supported – a warning is issued for other rates

channels – number of channels in *outfile* (1-16)
duration – duration of *outfile* in seconds
frequency – frequency of *outfile* in Hz. *Frequency* may vary over time
 -**a***amplitude* – amplitude of *outfile* (0.0 < Range < 1.0 max & default). *Amplitude* may vary over time
 -**t***tablesiz*e – size of table storing the waveform (Default: 256). The input value is always rounded to a multiple of 4.
 -**f** – write soundfile as 32-bit floats (Default: 16-bit shorts)

Understanding the SYNTH WAVE Process

A table of the appropriate size is created to store the waveform. It is then used to generate the soundfile with properties as specified by the parameters.

The breakpoint file option gives some additional shaping control, e.g., to create a softer attack and ensure that there will not be clicks.

Musical Applications

Geometric waveforms of this type (square, sine, ramp) do produce upper partials, but are mostly used for test purposes or to illustrate some feature of the software in a clear manner, rather than as source material for musical composition.

End of SYNTH WAVE