

Maximilian Marcoll

Nach dieser Erde
Canone Monodico No.3

[Interlock 5]

for digitally controlled sawtooth wave generator, 5 or 6 performers

Commissioned by
Herbordt/Mohren (Die Institution)
for Donaueschinger Musiktage 2019

Duration: ca.2'30"

Introduction

Interlock 5 has been written for 5 or 6 performers controlling a single sawtooth wave generator. The performers control the frequency of the same monophonic sound throughout the entire piece. However, they don't control the frequency *simultaneously*, but are *competing* for control. Only one of the performers can set a new pitch at a time. Since the performers play a polyphonic musical text, it is impossible to determine in advance who will actually be in control at any given moment.

The musical text of the piece is a simple version of Philip Hayes' (1738 - 1797) round "By the waters of Babylon", which was introduced to a wider pop music audience in a simplified version by Don McLean in the early 1970s. Some years later Gerd Kern wrote German Lyrics to the tune, whereupon it became popular in Germany in the 1980s, originally as a protest song against nuclear armament.

*Nach dieser Erde wäre da keine, die eines Menschen Wohnung wär'.
Darum Menschen achtet und trachtet, dass sie so bleibt!
Wem denn wäre sie ein Denkmal, wenn sie still die Sonn' umtreibt?*

Roughly translated:

*After this earth there would be none that could be a human's home.
Therefore people watch it, protect it, so it remains!
Who would bother if it floated quietly around the sun?*

The three voices of the round are to be divided amongst the performers so that each voice is played by two performers simultaneously. The first group starts the performance. When they reach the second line, the second group is starting at the beginning and so on.

If the piece is being performed by only 5 performers, the second voice to start playing is to be performed by only one performer alone.

Once all three voices are playing, the canon is to be played through at least 4 times.

Do not try to play too precise! The less tight the ensemble plays together, the more beautiful the result will be!

Staging & Sound Projection

Since the piece is mono, the ideal staging scenario would be to have one massive full range speaker, or a stack of several speakers, including a LFE, in the center of the stage.

Alternatively, if it proves difficult to arrange a special setup for this piece and if there are lots of loudspeakers installed in the space already, it is recommended to use virtually all of the available speakers to fill the room as much as possible! The piece is to be performed at very high volume levels!

Technical Information

1. What you need to have

- 5 or 6 midi controllers able to send note messages in the range from #57 to #72 (a typical two octave mini midi keyboard will suffice).
- a computer running csound (freely available from www.csound.com)
- an audio interface
- at least one full-range-loudspeaker

2. What you need to do

First, connect the midi controllers to the computer. The controllers should send midi note messages on the same range of pitches, (on a typical 2-octave midi keyboard, find a setting that allows you to send pitches from #48 to #72) but on different midi channels, starting at 1, up to a maximum of 6.

Only the controller sending on midi channel 1 can actually switch on the sound at the beginning of the piece, by hitting the low c (midi #48). Right after the end of the piece, the sound is to be switched off, by hitting the c again. If there are only 5 performers, the second group will consist of only one performer.

| | | |
|-------------------------|----------------|-------------------------------|
| Performer 1: 1st group, | Midi Channel 3 | |
| Performer 2: 1st group, | Midi Channel 4 | |
| Performer 3: 2nd group, | Midi Channel 5 | |
| Performer 4: 2nd group, | Midi Channel 6 | [optional] |
| Performer 5: 3rd group, | Midi Channel 1 | [sound on/off: midi note #48] |
| Performer 6: 3rd group, | Midi Channel 2 | |

There are two versions of the patch for this piece: A and B.

If all performers are capable of playing the musical text on the keyboard, you should go for version A.

Version B has been created for cases in which only three performers are reliable and the other three (or two) are not (small children for example). If you choose version B, the reliable performers should be performers 1, 3 and 5.

If you have not installed Csound on your system, please do so. Csound is open source, available free of charge and the installation should not take more than a few minutes. If you are not familiar with Csound, the following instructions might seem complicated at first, but it's actually very simple.

On a linux machine or on a mac I usually run csound directly from the shell. In case you are unfamiliar with the shell, I included some more information further below.

To run the patch, simply navigate to the CanoneMonodicoNo3_LIVE directory and start csound by entering one of the following lines:

```
csound ndeA.csd  
    for version A, or
```

```
csound ndeB.csd  
    for version B.
```

At first csound will respond with an error message. This is because you need to specify which audio device to use.

In the error message, just above the line that reads
"*** PortAudio: error: device number 100 is out of range" there is a list of available devices and their numbers. Pick the one you want to use and follow these steps:

- Open the file *ndeA.csd* or *ndeB.csd* in your favorite text editor (word processors like "MS Word" will NOT work!).
- Put a semicolon in front of the line that reads "-odac100"
- A few lines above replace the 0 in -odac0 with your device number.
- Save the file.
- That's it! You're ready to go!

IMPORTANT NOTE: To stop Csound, simply hit Ctrl + C

The csound instruments necessary to perform this piece are included at the end of this score.

3. How to start Csound from the shell on a Mac

From the Finder, hit $\text{⌘} + \text{⇧} + \text{U}$ or navigate manually to /Applications/Utilities .
In this folder there lives a little program called "Terminal". Open it, type the following line and hit enter:

which csound

If nothing happens, your installation of csound did not succeed.

The terminal should respond with a line like this:

/usr/local/bin/csound

telling you where csound was installed to.

The easiest way to start the patch for the Canone is to type "csound " (the space in the end is intentional) and then drag the *ndeA.csd* or *ndeB.csd* file onto the terminal window.

The command should then read something like this:

csound /Users/max/Desktop/CanoneMonodicoNo3_LIVE/ndeA.csd

Hitting enter will start the program. If you want to stop csound, simply hit ctrl+c.

To repeat the same command, hit the upwards arrow key. Using the up and downwards arrow keys, you navigate in your command history.

Hitting enter will start whatever command you chose.

For more information on how to use the terminal, please consult this page:

<https://www.imore.com/how-use-terminal-mac-when-you-have-no-idea-where-start>

If you need help or have further questions please do not hesitate to get in touch!


```

.....
;
; Csound instruments for
;
; "Nach dieser Erde"
;
; Canone Monodico No.3
;
; (Interlock #5)
;
; by Maximilian Marcoll
;
; 2019
;
;
; VERSION A
;
.....
<CsoundSynthesizer>
<CsOptions>
.....
;
; M I D I
;

    -Ma

.....
;
; A U D I O
;

    -odac0

; the following setting will produce an error but will show
; you a list of available audio devices and their device numbers:
;
    -odac100
;
; simply comment it out (put a semicolon in front of it) and change the number in the
; setting above to the desired audio output device
;
.....
    -m99

</CsOptions>
<CsInstruments>

    sr = 44100
    nchnls = 1
    ksmps = 1
    Odbfs = 1

.....

```

.....
.....

; Use this to record the live generated signal into "LiveRec_<date&time>.aif" :

; #define WriteToDisk # #

.....
.....

```
giTransStepSize      =          pow(2, 1/12)
giStepSize           =          pow(2, 1/12.5)
giNumSteps           =          12
giStartCPS           =          27.5
giMaxDelay           =          .3
giSubDBTable         ftgen      0, 0, 32768, -6, 0, 81, 0, 75, -90, 32587, -90
giInterRamp          =          0.035
gkOnOff              =          0
giStartRamp          =          0.1
giDBRange            =          30
giOctRange           =          7.25
giMinFreq            =          giStartCPS
giMidiHigh           =          72
giMidiLow            =          57
gkFreq               =          init 1
```

.....
.....

```
turnon "midi"
turnon "saw"
seed 0
massign 0, 0
```

.....
.....

instr detune

```
iStepSize =      giStepSize
iPitch limit p4, giMidiLow, giMidiHigh
ifac pow iStepSize, iPitch - giMidiLow
ichan = p5
istartCps = giStartCPS*pow(giTransStepSize, giNumSteps*ichan-1)
icps = istartCps * ifac
gkFreq expseg i(gkFreq), giInterRamp, icps
```

endin

.....
.....

```
instr midi
```

```
    kstatus, kchan, kdata1, kdata2  midiin
    ktrigger changed kstatus, kchan, kdata1, kdata2
    if ktrigger == 1 && kstatus == 144 then
        if kdata2 > 0 then
            schedkwhen ktrigger, 0, 0, "dist", 0, giInterRamp, kdata1, kchan
            if kdata1 == 48 && kchan == 1 then
                schedkwhen ktrigger, 0, 0, "gate", 0, giStartRamp
            endif
        endif
    endif
endin
```

```
.....
```

```
instr gate
```

```
    ii = i(gkOnOff)
    is = 1-ii
    gkOnOff linseg ii, p3, is
```

```
endin
```

```
.....
```

```
instr dist
```

```
    ir      linrand      giMaxDelay
           turnoff2     "detune", 0, 0
           schedule     "detune", ir, giInterRamp, p4, p5
           turnoff
```

```
endin
```

```
.....
```

```
instr saw
```

```
    kFreq =      gkFreq
    koct   log2   kFreq/giMinFreq;
    kdBFac =     koct/giOctRange
    kdamp  =     kdBFac * giDBRange;
    kdB    =     0 - kdamp;
    kLinVol =    pow(10, kdB/20)
    kLinVol limit kLinVol, 0, 1
    aSaw   vco2   1, kFreq, 0, 0, 0, .5
    kSubdB tablei kFreq, giSubDBTable
    kSubAmp db   kSubdB
    aSub    poscil kSubAmp * .75, kFreq * .5
    aOut    =     (aSaw * .75 + aSub)*kLinVol
           out    aOut * gkOnOff
```

```
    #ifdef WriteToDisk
```

```
        itim   date
        Stim   dates itim
        Syear  strsub Stim, 20, 24
        Smonth strsub Stim, 4, 7
        Sday   strsub Stim, 8, 10
        iday   strtod Sday
        Shor   strsub Stim, 11, 13
        Smin   strsub Stim, 14, 16
        Ssec   strsub Stim, 17, 19
        Sfilnam sprintf "LiveRec_%s_%s_%02d_%s_%s_%s.aif", Syear, Smonth, iday, Shor, Smin, Ssec
        fout   Sfilnam, 24, aOut
```

```
    #end
```

```
endin
```

```
.....
```

```
</Csinstruments>
```

```
<CsScore>
```

```
    ;#include "ndeA.sco"
```

```
</CsScore>
```

```
</CsoundSynthesizer>
```

```

.....
;
; Csound instruments for
;
; "Nach dieser Erde"
;
; Canone Monodico No.3
;
; (Interlock #5)
;
; by Maximilian Marcoll
;
; 2019
;
;
; VERSION B
;
.....
<CsoundSynthesizer>
<CsOptions>
.....
;
; M I D I
;

    -Ma

.....
;
; A U D I O
;

    -odac0

; the following setting will produce an error but will show
; you a list of available audio devices and their device numbers:
;
    -odac100
;
; simply comment it out (put a semicolon in front of it) and change the number in the
; setting above to the desired audio output device
;
.....
    -m99

</CsOptions>
<CsInstruments>

    sr = 44100
    nchnls = 1
    ksmps = 1
    Odbfs = 1
.....

```

.....
.....

; Use this to record the live generated signal into "LiveRec_<date&time>.aif" :

; #define WriteToDisk # #

.....
.....

```
giTransStepSize      =          pow(2, 1/12)
giStepSize           =          pow(2, 1/12.5)
giNumSteps           =          12
giStartCPS           =          27.5
giMaxDelay           =          .3
giSubDBTable         ftgen      0, 0, 32768, -6, 0, 81, 0, 75, -90, 32587, -90
giInterRamp          =          0.035
gkOnOff              =          0
giStartRamp          =          0.1
giDBRange            =          30
giOctRange           =          7.25
giMinFreq            =          giStartCPS
giMidiHigh           =          72
giMidiLow            =          57
gkFreq               init      1
giControlF           =          27.5
```

.....
.....

```
turnon "midi"
turnon "saw"
seed 0
massign 0, 0
```

.....
.....

instr detune

```
iStepSize =      giStepSize
iPitch limit p4, giMidiLow, giMidiHigh
ifac pow iStepSize, iPitch - giMidiLow
ichan = p5
istartCps = giStartCPS*pow(giTransStepSize, giNumSteps*ichan-1)
icps = istartCps * ifac
gkFreq expseg i(gkFreq), giInterRamp, icps
giControlF = icps
```

endin

.....
.....


```
instr distB
    ir    linrand giMaxDelay
          turnoff2    "detune", 0, 0
          turnoff2    "detuneB", 0, 0
          schedule    "detuneB", ir, giInterRamp, p4, p5
          turnoff
    endin
```

```
.....
```

```
instr saw
    kFreq =    gkFreq
    koct  log2  kFreq/giMinFreq;
    kdBFac =   koct/giOctRange
    kdamp =   kdBFac * giDBRange;
    kdB =     0 - kdamp;
    kLinVol = pow(10, kdB/20)
    kLinVol limit kLinVol, 0, 1
    aSaw  vco2  1, kFreq, 0, 0, 0, .5
    kSubdB tablei kFreq, giSubDBTable
    kSubAmp db  kSubdB
    aSub  poscil kSubAmp * .75, kFreq * .5
    aOut  =      (aSaw * .75 + aSub)*kLinVol
          out    aOut * gkOnOff
```

```
#ifdef WriteToDisk
    itim  date
    Stim  dates itim
    Syear strsub Stim, 20, 24
    Smonth strsub Stim, 4, 7
    Sday  strsub Stim, 8, 10
    iday  strtod Sday
    Shor  strsub Stim, 11, 13
    Smin  strsub Stim, 14, 16
    Ssec  strsub Stim, 17, 19
    Sfilnam sprintf "LiveRec_%s_%s_%02d_%s_%s_%s.aif", Syear, Smonth, iday, Shor, Smin, Ssec
          fout  Sfilnam, 24, aOut
#end
```

```
endin
```

```
.....
```

```
</CsInstruments>
<CsScore>

    ;#include "ndeB.sco"

</CsScore>
</CsoundSynthesizer>
```


www.marcoll.de