

Bijsluiter Presets 2014 t.b.v. NLT-module *Sound Design*

Nov 2014

Hier vind je de belangrijkste nieuwe functies en eigenschappen van WaveWizard en enkele opmerkingen over de opdrachten van de module *Sound Design* die door de nieuwe presets kleine veranderingen hebben ondergaan, waardoor je minder handelingen hoeft te verrichten.

Download presets WaveWizard 2014

Voor de nieuwe versie van WaveWizard is een aantal presets van de module *Sound Design* aangepast. De nieuwe presets kun je [hier downloaden](#).

(1)

Functies met *return-variabelen*

Zoals in alle programmeertalen kun je in WaveWizard namen opgeven voor variabelen (instructie **Declareer**). WaveWizard declareert ook zelf een aantal namen voor variabelen en constanten die we *systeemvariabelen* of *-constanten* noemen (voor een overzicht klik op menu Help! → Overzicht gereserveerde variabelen).

Een aantal van die systeemvariabelen horen bij instructies en worden vaak *return-variabelen* genoemd: de instructie voert een bepaalde taak uit en "keert terug" met informatie die wordt opgeslagen in een variabele. Bijvoorbeeld de instructie **Open WAV** opent een wav-bestand en slaat de lengte van de wav, uitgedrukt in aantal samples, op in **WAV_lengte**. Dus **WAV_lengte** is de return-variabele van de instructie **Open WAV**.

Een instructie kan ook *meerdere* return-variabelen hebben (bijv. de instructie **Amplitudo, Maximum, Minimum, Som, RMS**).

Met behulp van return-variabelen kun je processen verder automatiseren en code schrijven die minder data-afhankelijk is. Ook in de presets_2014 van de module *Sound Design* kom je geregeld return-variabelen tegen.

(2)

Geluidssporen S1, S2 en S3 zijn 32-bits *floating point*

De geluidssporen S1, S2 en S3 zijn niet langer buffers met 16-bits *gehele* getallen, maar met 32-bits *floating point* getallen. Daardoor neemt de geluidskwaliteit enorm toe. Recursieve bewerkingen, zoals de *snaarplukvergelijking* (zie H7.2) kun je nu ook direct als recursieve differentievergelijking op een geluidsspoor realiseren, wat in de oude versie t.g.v. accumulerende afrondingsfouten al snel zou hebben geleid tot hoorbare afwijkingen.

enorm dynamisch bereik

Je kunt nu op een geluidsspoor S1, S2 of S3 astronomisch grote getallen zetten, bijv. 10^{30} . Dat is veel en veel groter dan het 16-bits getalbereik van de geluidskaart ($\pm 2^{15}$). WaveWizard geeft bij een aantal functies een *melding* wanneer een uitwijking op een geluidsspoor S1, S2 of S3 groter is dan de geluidskaart aan kan. Met name als je veel bewerkingen achter elkaar doet, zul je die melding geregeld zien. Dat wil niet zeggen dat je dan iets fout hebt gedaan! Het is alleen maar een waarschuwing dat het signaal zachter gemaakt moet worden voordat je het kunt afspelen. Juist die grote dynamiek geeft je de ruimte om allerlei bewerkingen uit te voeren, zonder dat je je zorgen hoeft te maken of het volume niet te groot wordt. Het volume op- of neerschalen doe je als laatste bewerking; dat gaat heel gemakkelijk met de nieuwe functie **Amplitudo**, **Maximum**, **Minimum**, **Som**, **RMS**. Zie hieronder.

(3)

Amplitudo, Maximum, Minimum, Som, RMS

Met de nieuwe versie kun je een geluid het juiste volume geven zonder dat je de versterkings- of verzwakkingsfactor kent! Gebruik daarvoor de nieuwe functie: **Amplitudo**, **Maximum**, **Minimum**, **Som**, **RMS**.

Bijvoorbeeld je hebt op spoor S1 de onderstaande bewerking uitgevoerd:

```
Bewerk signaal
n0           0
n1           2000
Bewerking   S1[n] = 1,01^n
```

Nu wil je weten wat de maximale uitwijking is. Die vind je als volgt:

```
Amplitudo, Maximum, Minimum, Som, RMS
Buffer   S1[0]
Aantal   2000

Print Amplitudo
```

Na uitvoering van de code lees je in Memo: **Amplitudo** = 362708928. We noemen **Amplitudo** een *return-variabele*: de instructie onderzoekt wat de amplitude is van het signaalfragment dat je hebt opgegeven en slaat de uitkomst op in **Amplitudo**. Je kunt **Amplitudo** gebruiken in de code die volgt op de instructie. Stel je wilt dat de uitwijking maximaal 8000 is. Dan voer je de volgende bewerking uit:

```
x = 8000 / Amplitudo

Bewerk signaal
n0           0
n1           2000
Bewerking   S1[n] = x*S1[n]
```

(4)

Bufferwaarden met negatieve indices gedefinieerd als nul

In de oude versie kreeg je een foutmelding als je noteerde **Print F1[-123]**, want *negatieve* indices verwijzen naar buffergetallen die eenvoudig niet bestaan. In de nieuwe versie is **F1[-123]** gelijk aan nul.

Toepassing: het wordt nu heel eenvoudig om filters en echo-effecten direct te schrijven in de vorm van een differentievergelijking met beginconditie nul (wat meestal het geval is). Zie Sound Design H4.3.

In de Presets_2014 van *Sound Design* vind je daar vele voorbeelden van. Typisch geval is een echo-effect. Je kunt nu simpelweg noteren:

```
Bewerk signaal
n0           0
n1           Fs
Bewerking   S2[n] = S1[n] + S1[n-100]
```

Als bijv. $n=0$, dan luidt de bewerking: $s2[0] = s1[0] + s1[-100] = s1[0] + 0 = s1[0]$.

(5)

Automatische lineaire interpolatie van alle buffers

Stel, je hebt de volgende code:

```
S1[3] = 3000
S1[4] = 4000
Print S1[pi]
```

Als je deze code draait op de oude versie, dan krijg je als antwoord: $S1[pi] = 3000$ want in de oude versie wordt eerst van $pi = 3,14\dots$ het deel achter de komma verwijderd om er een geldig index-getal van te maken, zodat je krijgt $pi = 3$. Dan wordt gelezen: $S1[3]$.

Als je de code draait op de nieuwe versie, dan is het resultaat:

```
S1[pi] = 3141,59265358979
```

In de nieuwe versie vindt dus automatisch *lineaire interpolatie* plaats tussen twee opeenvolgende getallen in een buffer; zowel bij de audio-buffers S1, S2 en S3 als de double float buffers. Daardoor zijn de (discrete) buffers "quasi-continu" geworden.

Toepassingen:

- alle bewerkingen waarin tijdschaalveranderingen voorkomen, bijv. **Chorus**, **Flanger**;
- stuursignalen, zoals een envelop generator (zie H4) of laagfrequente ruis;
- opschaling indices grafieken.

Soms is automatische interpolatie ongewenst. Dan kun je noteren:

$S1[\text{int}(pi)]$. Zie **int(x)** en **frac(x)**.

(6)

nieuwe functies: int(x) en frac(x)

geven van *floating point* getal x resp. het deel vóór en achter de komma:

```
int(pi) = 3
frac(pi) = 0,141592653589793
```

(7)

Spectrumanalyser twee frequentieresoluties

In de oude versie had de spectrum analyzer slechts één frequentieresolutie, nl. 43 Hz. Dat is de resolutie van de FFT met 1024 samples ($F_s / 2^{10} = 43,066\dots$). In heel veel praktische gevallen heb je graag een wat hogere resolutie met name in het lage frequentiegebied. Daarom heeft de nieuwe versie een extra instelling gekregen: 0 - 5000 Hz. Daardoor wordt de FFT berekend over 4096 samples, zodat de

frequentieresolutie gelijk wordt aan $F_s / 2^{12} = 10,76660$ Hz. Alleen het gebied van 0 tot 5000 Hz wordt afgebeeld.

Een andere verbetering is dat de spectrumanalyzer de Gemiddelden en Maxima ook afbeeldt op de logaritmische representaties.

(8)

nieuwe functie: Scoop en Spectrumanalyzer

Je kunt het venster van Scoop en Spectrumanalyzer niet alleen openen via een menu of toets m, maar ook via een preset-instructie. Het voordeel daarvan is dat je de instellingen van Scoop en Spectrumanalyzer vooraf kunt regelen. Dit wordt, waar nodig, in de presets_2014 toegepast, waardoor je in de opdrachten zinnen als: "klik op Max" of "stel Y-slider van Scoop zo in dat..." kunt negeren.

```
Scoop en Spectrumanalyzer
spoor (S1, S2 of S3)    S1
Max                    n          ! knop 'Max' al dan niet activeren
Gem                    n          ! knop 'Gem' al dan niet activeren
Amplitude in dB        n
Frequentie Log         n
Hann-venster           j
0-5000 Hz              n
Trigger Frequentie     Fs/10    ! formule of constante
Y Scoop                1          ! verticale vergroting scoopsignaal
Y Spectrum              4          ! verticale vergroting spectrum
```

N.B. Deze instructie is altijd de *laatste* instructie in een preset. Instructies die je daaronder zet worden niet uitgevoerd!

(9)

nieuwe functie: Open WAV

Een geluidsbestand openen kun je op twee manieren doen: (1) direct met de muis klikken in het bekende bestand-selectievenster; (2) in de nieuwe versie ook als presetcommando **Open WAV**. Daarmee kun je een onbeperkt aantal geluidsbestanden automatisch achter elkaar te openen en op verschillende tijdstippen van verschillende sporen zetten.

N.B.!!

1. De wav-bestanden die je wilt openen d.m.v. **Open WAV**, dienen in dezelfde map te staan als WaveWizard.exe.
2. in veel opdrachten wordt *eerst* gevraagd om een wav te openen en *vervolgens* om een bepaalde preset te openen. Bij WaveWizard_2014 open je *eerst* de preset; daarin staat dan een instructie **Open WAV** die het wav-bestand voor je opent.

De instructie luidt:

```
Open WAV
WAV naam    Bach_vioolconcert.wav
vanaf       240*Fs
duur        ?
buffer      S1[20*Fs]
```

Je kunt hiermee het bestand inladen vanaf en plaatsen op elk gewenst tijdstip. Je kunt ook slechts een fragment inladen.

WAV naam de naam van het WAV-bestand dat je wilt openen.

vanaf zeker voor lange geluidsbestanden is het handig als je het kunt inladen vanaf een zeker tijdstip. Druk dat tijdstip uit in aantal samples. Dat kan ook in formule-vorm, bijv. $240 * F_s$: inladen vanaf tijdstip 4 minuten ($4 \times 60 = 240$ sec; 1 sec = F_s samples).

duur uitgedrukt in aantal samples, zoals **vanaf**. Je kunt dus ook slechts een deel van een bestand laden. Als je het *hele* bestand wilt inladen, of als je niet weet hoe lang het duurt, kun je invullen: '?'

buffer de buffer en de index op van de eerste sample van het WAV-bestand. Bijv. $S1[20 * F_s]$.
Als het WAV-bestand stereo is, dan wordt het tweede spoor geplaatst in de eerstvolgende buffer; bijv. als je opgeeft $S1[0]$, dan wordt het tweede spoor op $S2[0]$ gezet.

de return-variabele **WAV_lengte**

Het aantal samples van het geopende bestand slaat WaveWizard op in de variabele **WAV_lengte**. Gebruik **WAV_lengte** om bijvoorbeeld het aantal bewerkingen op te geven dat je wilt uitvoeren op het geopende WAV-bestand.

Als je bij **duur** geen vraagteken hebt gezet, zal **WAV_lengte** gelijk zijn aan wat je bij **duur** hebt ingevuld.

(10)

nieuwe functie: **Open MID**

Een MIDI-bestand openen kun je eveneens op twee manieren doen: (1) direct met de muis klikken in het bekende bestand-selectievenster; (2) in de nieuwe versie ook als presetcommando **Open MID**.

N.B.!!

1. De mid-bestanden die je wilt openen d.m.v. **Open MID**, dienen in dezelfde map te staan als WaveWizard.exe.
2. in opdrachten wordt *eerst* gevraagd om een mid te openen en *vervolgens* om een bepaalde preset te openen. Bij WaveWizard_2014 open je *eerst* de preset; daarin staat dan een instructie **Open MID** die het mid-bestand voor je opent.

return-variabele van **Open MID** is: **BPM** (Beats Per Minute).