

Opgaven reeks 2 Scilab

Voor iedere opgave geef je de code en grafische weergave weer in een pdf-bestand. Je stuurt ook voor iedere opgave het codebestand mee. Code kan je schrijven in SciNotes en bewaren. Een scilab codebestand heeft de extensie ".sce".

1. Gegeven de vergelijking om een pulstrein te genereren met een bepaalde duty-cycle:

$$u = U_{max} \cdot \frac{\tau}{T} + 2U_{max} \cdot \frac{\tau}{T} \left[\frac{\sin\left(\frac{\pi\tau}{T}\right)}{\pi\tau T} \cos(\omega t) + \frac{\sin\left(\frac{2\pi\tau}{T}\right)}{2\pi\tau T} \cos(2\omega t) + \frac{\sin\left(\frac{3\pi\tau}{T}\right)}{3\pi\tau T} \cos(3\omega t) + \dots \right]$$

Hierbij is $\frac{\tau}{T}$ de duty-cycle van de puls en T de periode van het signaal

Schrijf een code in scilab die een pulstrein weergeeft met duty-cycle 25 % en duty-cycle 75% voor een frequentie van 1 kHz. Om de puls te vormen maak je gebruik van de eerste 7 harmonischen.

Oplossing:

```
clf
dc=0.25
t=0.0:0.000025:0.001*4

f=1000
Umax=5
w=2*%pi*f*t

harm=((sin(1*%pi*dc))/(1*%pi*dc))*cos(1*w) + ..
      ((sin(2*%pi*dc))/(2*%pi*dc))*cos(2*w) + ..
      ((sin(3*%pi*dc))/(3*%pi*dc))*cos(3*w) + ..
      ((sin(4*%pi*dc))/(4*%pi*dc))*cos(4*w) + ..
      ((sin(5*%pi*dc))/(5*%pi*dc))*cos(5*w) + ..
      ((sin(6*%pi*dc))/(6*%pi*dc))*cos(6*w) + ..
      ((sin(7*%pi*dc))/(7*%pi*dc))*cos(7*w)

y=(Umax*dc+2*Umax*dc*harm)

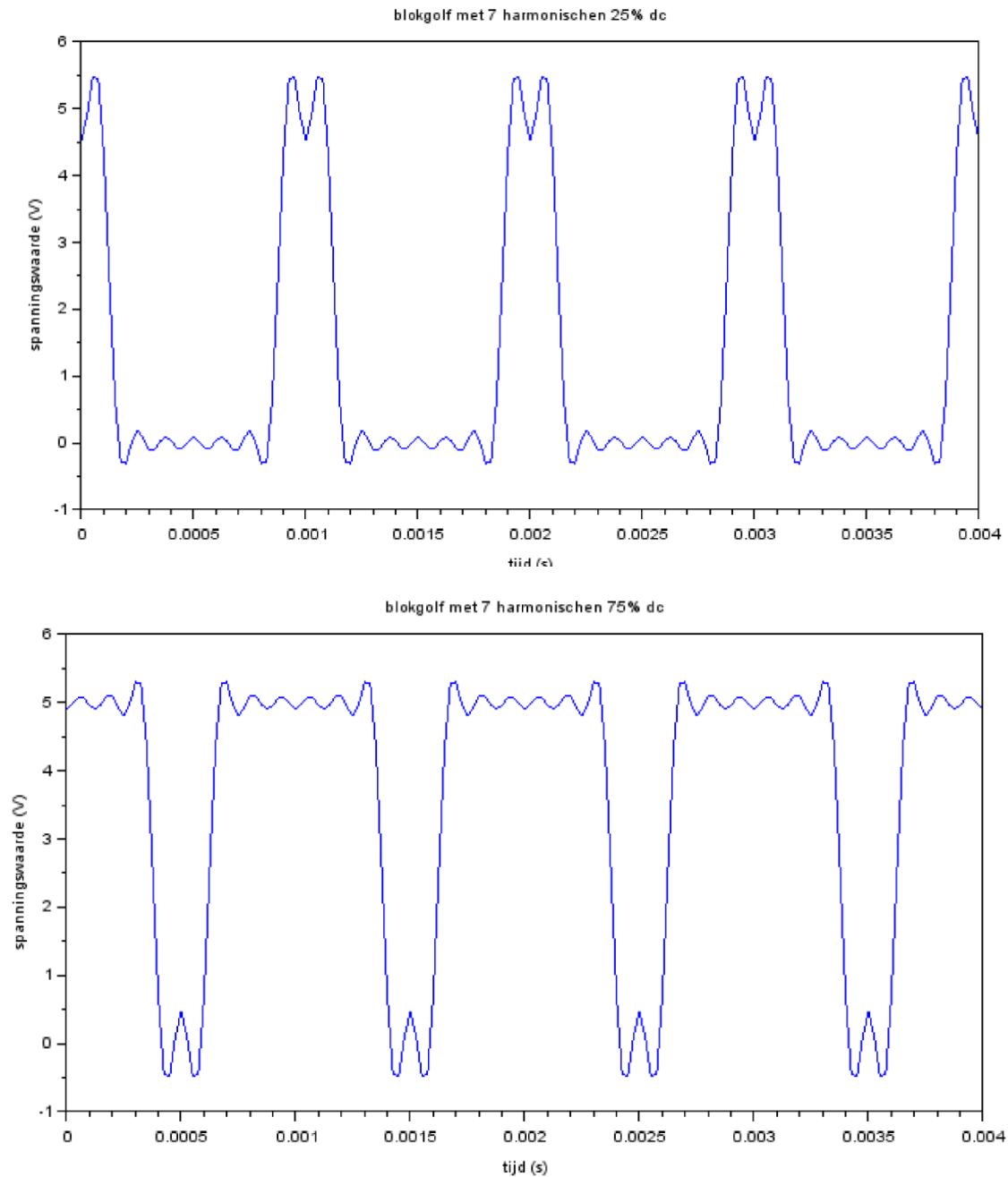
subplot(221)
plot(t,y)
title("blokgolf met 7 harmonischen 25% dc")
ylabel("spanningswaarde (V)")
xlabel("tijd (s)")

dc=0.75
harm=((sin(1*%pi*dc))/(1*%pi*dc))*cos(1*w) + ..
      ((sin(2*%pi*dc))/(2*%pi*dc))*cos(2*w) + ..
      ((sin(3*%pi*dc))/(3*%pi*dc))*cos(3*w) + ..
      ((sin(4*%pi*dc))/(4*%pi*dc))*cos(4*w) + ..
      ((sin(5*%pi*dc))/(5*%pi*dc))*cos(5*w) + ..
      ((sin(6*%pi*dc))/(6*%pi*dc))*cos(6*w) + ..
      ((sin(7*%pi*dc))/(7*%pi*dc))*cos(7*w)
```

```

y=(Umax*dc+2*Umax*dc*harm)
subplot(222)
plot(t,y)
title("blokgolf met 7 harmonischen 75% dc")
ylabel("spanningswaarde (V)")
xlabel("tijd (s)")

```

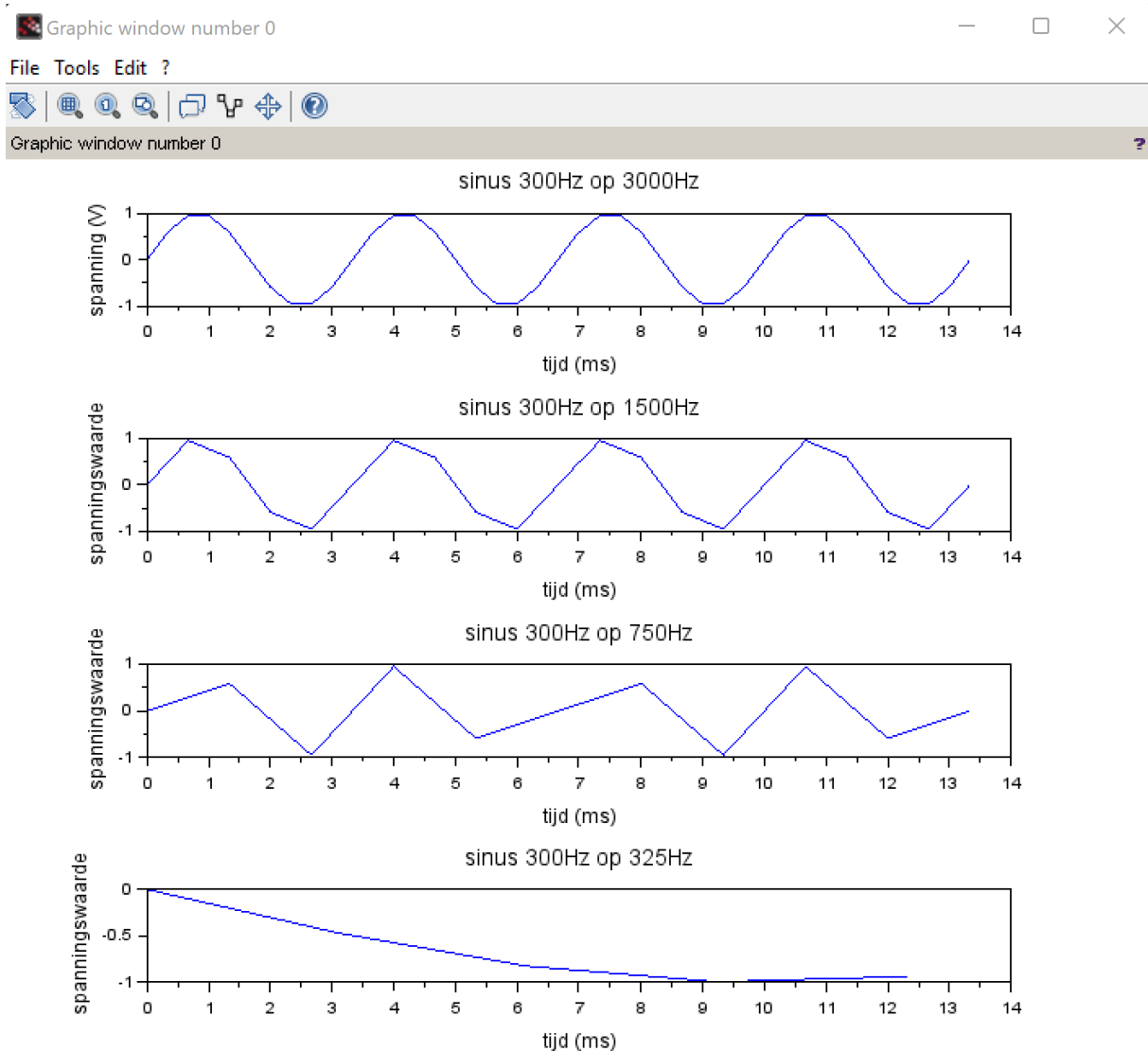


2. A) Geef in scilab onder elkaar in één graphic window weer:
 - Een signaal van 300 Hz gesampled met een frequentie van 3000 Hz
 - Een signaal van 300 Hz gesampled met een frequentie van 1500 Hz
 - Een signaal van 300 Hz gesampled met een frequentie van 750 Hz
 - Een signaal van 300 Hz gesampled met een frequentie van 325 Hz
 Zorg ervoor dat de X-schaal de tijd in ms aangeeft.

b) Bij welke samplefrequenties treedt aliasing op? Geef hiervoor een verklaring hoe dit komt.

Oplossing A:

```
1  clf
2
3  t3kHz=0.0:(1/3000):(1/300)*4
4  sig300Hz=sin(2*pi*300*t3kHz)
5  subplot(411)
6  plot(t3kHz*1000,sig300Hz)
7  title("sinus-300Hz-op-3000Hz")
8  ylabel("spanning (V)")
9  xlabel("tijd (ms)")
10
11 t1k5Hz=0.0:(1/1500):(1/300)*4
12 sig300Hz=sin(2*pi*300*t1k5Hz)
13 subplot(412)
14 plot(t1k5Hz*1000,sig300Hz)
15 title("sinus-300Hz-op-1500Hz")
16 ylabel("spanningswaarde")
17 xlabel("tijd (ms)")
18
19 t750Hz=0.0:(1/750):(1/300)*4
20 sig300Hz=sin(2*pi*300*t750Hz)
21 subplot(413)
22 plot(t750Hz*1000,sig300Hz)
23 title("sinus-300Hz-op-750Hz")
24 ylabel("spanningswaarde")
25 xlabel("tijd (ms)")
26
27 t325kHz=0.0:(1/325):(1/300)*4
28 sig300Hz=sin(2*pi*300*t325kHz)
29 subplot(414)
30 plot(t325kHz*1000,sig300Hz)
31 title("sinus-300Hz-op-325Hz")
32 ylabel("spanningswaarde")
33 xlabel("tijd (ms)")
```



Oplossing B:

Bij samplefrequentie 325 Hz. De samplefrequentie moet minstens 2 keer groter zijn dan de hoogste frequentie in het signaal. 325 Hz is de enige samplefrequentie in deze oefening die niet twee keer groter is dan de 300 Hz. Het aliassignaal is gelijk aan 35 Hz.

3. A) Maak een testsignaal dat 24 ms lang duurt bestaande uit een signaal van 250 Hz en 2 V amplitude, een signaal van 2500 Hz en 0.5 V amplitude en een signaal van 7000 Hz en 0.8 V amplitude. Breng hierin een storing aan die telkens twee sampletijden duurt en gelijk is aan waarde -3. Laat de storing terugkomen om de 6 milliseconden. De samplefrequentie bedraagt 16 kHz.

Geef het signaal weer met een tijdsas.

Oplossing:

```
clf
clc

t = [0.0 : 625D-4 : 24]

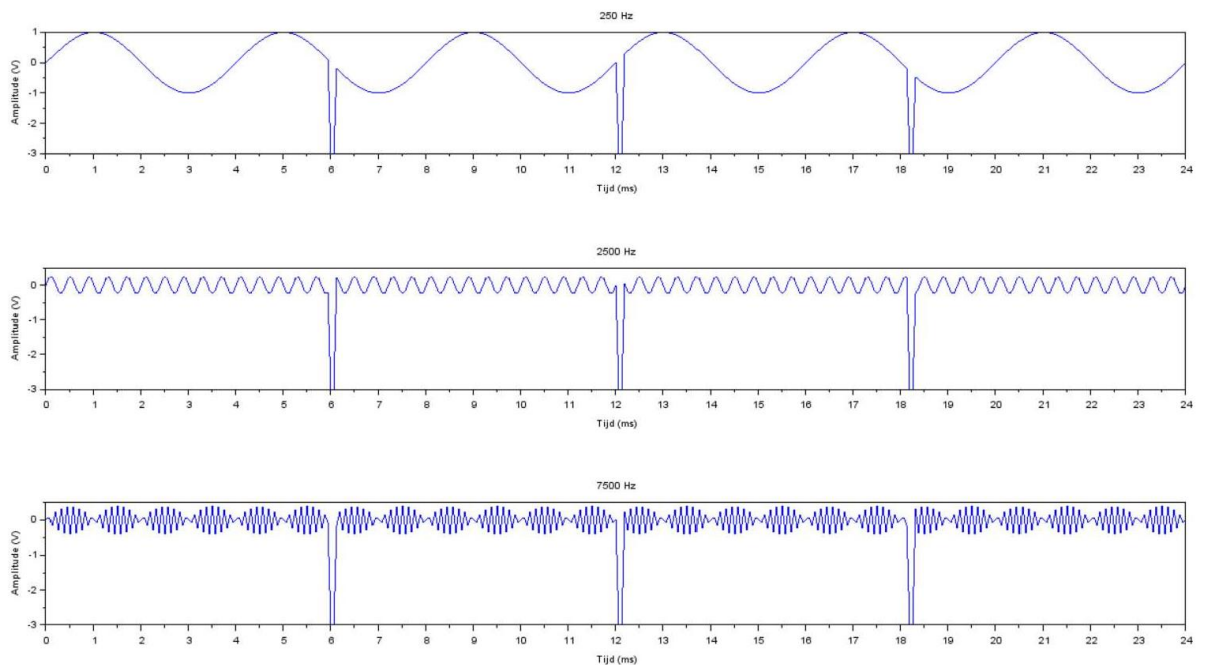
// 1 Periode 16kHz = 0.0000625 = 625D-7
StoorEind = 0;

for i = 1 : length(t)
    if t(i) >= StoorEind + 6 then
        disp(i) // Index waar de waarde verandert moet
        worden
        disp(i+1)
        StoorEind = t(i+1)
        // 97, 98, 194, 195, 292, 293
    end
end

sin250Hz = sin(2 * %pi * 0.25 * t)
sin250Hz(97) = -3
sin250Hz(98) = -3
sin250Hz(194) = -3
sin250Hz(195) = -3
sin250Hz(292) = -3
sin250Hz(293) = -3
subplot(3, 1, 1)
xtitle("250 Hz", "Tijd (ms)", "Amplitude (V)")
plot(t, sin250Hz)
```

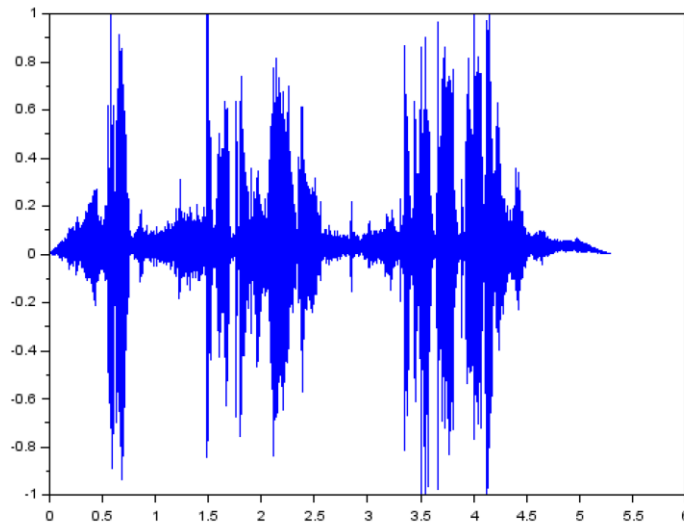
```
sin2500Hz = 0.25 * sin(2 * %pi * 2.5 * t)
sin2500Hz(97) = -3
sin2500Hz(98) = -3
sin2500Hz(194) = -3
sin2500Hz(195) = -3
sin2500Hz(292) = -3
sin2500Hz(293) = -3
subplot(3, 1, 2)
xtitle("2500 Hz", "Tijd (ms)", "Amplitude (V)")
plot(t, sin2500Hz)
```

```
sin7500Hz = 0.4 * sin(2 * %pi * 7.5 * t)
sin7500Hz(97) = -3
sin7500Hz(98) = -3
sin7500Hz(194) = -3
sin7500Hz(195) = -3
sin7500Hz(292) = -3
sin7500Hz(293) = -3
subplot(3, 1, 3)
xtitle("7500 Hz", "Tijd (ms)", "Amplitude (V)")
plot(t, sin7500Hz)
```



4. Ga naar volgende website : <http://www.wavsource.com/> en download een waf-fille. Tracht deze file in te lezen in scilab en weer te geven in een plotfunctie.

```
1 [y,f,bit]=wavread('D:\sciDocuments\opgave2\anything.wav');  
2 t=[1:1:length(y)]*1/f;  
3 plot(t,y)
```



5. Probeer het geluid van oefening 4 af te spelen, bekijk hiervoor volgende instructie https://help.scilab.org/docs/5.3.1/en_US/sound.html

Oplossing :

```
1 [y,Fs,bits]=wavread('D:\sciDocuments\opgave2\anything.wav');  
2 sound(wav, Fs, -bits) //speel het wav-bestand op een normale frequentie  
3 sound(wav, Fs*2, -bits) //speel het wav-bestand op een dubbele frequentie  
4 sound(wav, Fs/2, -bits) //speel het wav-bestand op een halve frequentie
```

- B) Verdubbel de samplefrequentie en speel het geluid opnieuw af.
c) Halveer de samplefrequentie en speel het geluid opnieuw af.
D) Verklaar hetgeen je hoort.

Oplossing :

Halveren van de samplefrequentie vertraagt het geluid, Verdubbelen van de samplefrequentie versnelt het geluid