

## Case projekt 2 – Audio filter

### Baggrund

Man kan ofte have et ønske om at fjerne en enkelt tone/frekvens. Det kan fx. være 50 Hz fra lysnettet, som forstyrrer et målesignal eller det kan være en kraftig forvrængnings-komponent pga. en ikke-linearitet i ens system.

### Opgaven

I skal designe, analysere og implementere et 2. ordens IIR filter til at fjerne en enkelt, forstyrrende tone i et audio signal. Filtret er et notch-filter, dvs. det har et hak (notch) ved en enkelt frekvens. Højere ordens IIR filtre vil man ofte implementere som fx. en kaskade af 2.ordens filtre, så det er godt at have styr på 2.ordens filtre.

Proceduren vil altså være som følger :

- Analysere det givne lydsignal og undersøge hvilken frekvens, som ønskes fjernet.
- Designe IIR filtret vha. placering af poler/nuller
- Implementere IIR filtret på Blackfin processor
- Teste funktionen af filtret vha. både tonegenerator/oscilloskop og den givne lydfil

## Kom-godt-igang software

I har fået lydfilen med den forstyrrende rentone ("tale\_tone\_48000.wav"). Desuden ligger der et template projekt til CrossCore miljøet til programmering af Blackfin processoren, som I kan bruge i opgave 3.

### Opgave 1 – Analyse af inputsignal

I skal finde frekvensen af den tone, som I ønsker at fjerne. Indlæs lydfilen i fx. Matlab og analyser signalet her - med analyse menes at se på tids-signalet, amplitude-spektrum, spektrogram, ..

### Opgave 2 – IIR notch filter design

I skal designe et 2. ordens IIR notch filter vha. pol/nul-punkts placering i z-domænet. Nuller/poler har en radius og vinkel, da de er komplekse tal - fx.  $z_0 = r \exp(j\omega)$  hvor  $r$  er nul-radius og  $\omega$  er nul-vinkel.

Fra opgave 1 kender I vinkel af de to nuller og poler - husk at nuller/poler altid kommer i par af to, dvs. for hver nul/pol vil der være en tilsvarende med samme radius, men med negativ fortegn på vinklen (den kompleks konjugerede). Nul-radius kan I sætte til 1 og pol-radius kan I sætte til fx. 0,9.

Start med at eksperimentere i Matlab ved at kalde funktionen "fdatool". I Fdatool (=filter design and analysis tool) skal I vælge "pol-nulpunkts-analyse" (ikon i nedre venstre bjælke). Her kan I indsætte poler/nuller og se effekten i fx. frekvens-responset - det kan give en god intuitiv forståelse. Prøv at flytte pol-radius til andet end 0,9 - fx. 0,7 eller 0,999. Hvad er betydningen af pol-radius ?

1. Skitser pol-nul-punkts diagram for filtret
2. Udregn systemfunktionen  $H(z)$  og opskriv differensligningen herudfra. OBS: DETTE SKAL I GØRE MANUELT, DVS. "I HÅNDEN" OG BESKRIVES I RAPPORTEN.. I MÅ IKKE BLOT AFLÆSE FRA MATLAB !
3. Tegn signalgraf for filtret på direkte form I.
4. Bestem og plot frekvens-responset  $|H(ej\omega)|$  for filtret ud fra  $H(z)$  - Er det et ideelt filter ? (hvorfor/hvorfor ikke)
5. Test at filtret virker ved at filtrere lydfilen i Matlab (I kan benytte "filter" kommando i Matlab)

### Opgave 3 – Simulering af fixed-point implementation

Opgaven består i at få omsat differensligningen fra opgave 2 til et fungerende filter i Matlab-kode – OBS: Uden brug af indbyggede funktioner, dvs. vha. for-løkke.

#### Del A.

I første omgang skal I blot implementere filtret (med Matlabs "normale" floating-point datatype) og teste at I får fjernet tonen.

Plot evt. amplitudespektret for at verificere at tonen er væk.

#### Del B.

Bagefter skal I undersøge effekten af kvantisering på jeres filter i relation til en fixed-point implementation. Antag at man ønsker et meget skarpt filter og derfor vælger en pol-radius på 0,999.

I skal undersøge påvirkningen på filterkoefficienterne, dvs. såkaldt koefficient-kvantisering. I kan benytte følgende formel til at kvantisere et tal  $x$ , som ligger i intervallet -1 til 1, til  $b$  bits præcision :

$$x_{\text{kvant}} = 2^{(1-b)} * \text{round}(x * 2^{(b-1)})$$

Overordnet skal I følge, med udgangspunkt i jeres IIR filter

- Kvantisere jeres filterkoefficienter til et antal betydende bits i Matlab. Prøv med 4, 8 og 16 bits.
- Finde ændringen i pol-nulpunkts diagrammet og frekvensresponset - fjernes den rette tone?
- Afprøve filtret med de kvantiserede filterkoefficienter og afgøre om tonen fjernes.

### Opgave 4 – Test af filtret

Lav en slut-test af filtret. I kan teste filtret på flere forskellige måder :

- Afspilning af rentoner i Matlab
- Afspilning af frekvens-sweep ("chirp"-kommando) i Matlab
- Med den "forurenede" lydfil.