

## SIGBE/17/4

### a) Syntes av lågpasfilter.

(i) *Syntes av FIR-filter med fönsterfunktion.*

Talsekvensen i filen <http://www.users.abo.fi/~htoivone/courses/sigbe/signal3.dat> representerar en audiosignal  $\{x(nT_s)\}$  som diskretiserats med samplingsfrekvensen  $f_s = 11025$  Hz. Signalen är densamma som signalen i uppgift SIGBE/15/3 (`signal.dat`) men är korrumpierad med högfrekvent brus i frekvensområdet över ca 2750 Hz. (Lyssna!)

För att eliminera det oönskade högfrekventa bruset bör signalen filtreras med ett lämpligt filter. Undersök först om filtreringen kan göras med ett enkelt medeltalsfilter enligt

$$y(n) = \frac{1}{N} (x(n) + x(n-1) + \dots + x(n-N+1))$$

För att avgöra om acceptabelt filtreringsresultat kan uppnås på detta sätt, bestäm filtrets frekvenssvar för olika värden på  $N$  samt beräkna, och lyssna på, den filtrerade signalen.

Bestäm sedan ett faslinjärt FIR filter för ändamålet genom att tillämpa syntes med hjälp av lämplig fönsterfunktion. För att uppnå en lämplig kompromiss mellan prestanda och komplexitet hos filtret kan man t.ex. kräva att filtret uppfyller följande specifikationer:

- spärbandets hörnfrekvens: 2750 Hz
- dämpning i spärbandet:  $> 50$  dB
- övergångsbandets bredd: 250 Hz
- maximal avvikelse i passbandet: 0.05 dB

Plotta filtrets koefficienter samt upprita filtrets frekvenssvar (förstärkning och fasförskjutning). MATLAB-rutinen `fir1` eller det grafiska verktyget `fdatool` kan användas för filtersyntesen.

(ii) Bestäm den filtrerade signalen  $\{y(nT_s)\}$  som fås genom att filtrera signalen  $\{x(nT_s)\}$  med filtret enligt fall (i). MATLAB-rutinen `filter` kan användas.

- Bestäm spektret hos signalen  $y$  och verifiera att frekvenser över 2750 Hz faktiskt har eliminerats.
- Lyssna på signalen  $y$ ; kan man höra att det högfrekventa bruset har eliminerats?
- Den ursprungliga signalen  $x$  och den filtrerade signalen  $y$  kan jämföras i tidsplanet genom att observera att det faslinjära FIR-filtret medför en fasförskjutning som motsvarar en tidsfördröjning på  $L = (N-1)/2$  sampel (jmf ekvation (8.55) eller (8.84) i kompendiet). Jämför således signalerna genom att plotta  $y(nT_s)$  och  $x((n-L)T_s)$  i ett lämpligt tidsintervall (t.ex. för  $1001 \leq n \leq 1100$ ).

(iii) *Syntes av FIR-filter med Parks-McClellan metoden.*

Upprepa filtersyntesen i fall (i) genom att använda Parks-McClellan metoden.

Lämplig filterlängd kan bestämmas med hjälp av MATLAB-rutinen `firpmord`, och filtersyntesen kan utföras med programmet `firpm` eller med hjälp av det grafiska användargränssnittet `fdatool`.

**b) Syntes av notch-filter.**

Filen <http://www.users.abo.fi/~htoivone/courses/sigbe/x.dat> innehåller en audiosignal med samplingsfrekvensen 8192 Hz.

(i) Signalen är korrumpierad av en störande ton med okänd frekvens, som kan höras om man spelar upp signalen. Bestäm tonens frekvens.

(ii) Konstruera ett digitalt notch-filter för att eliminera störningen. Använd t.ex. 3dB bandbredden 10 Hz.

Upprita filtrets förstärkning  $|H(e^{j\omega})|$  som funktion av frekvensen i ett diagram med linjär (ej logaritmisk) skalning.

(iii) Filtrera signalen med filtret i (ii). Verifiera att den störande tonen eliminerats genom att jämföra den filtrerade signalen med den ursprungliga signalen

(se <http://www.users.abo.fi/~htoivone/courses/sigbe/y.dat>), samt genom att spela upp signalen.