

**COMP.SGN.100 Signaalinkäsittelyn perusteet,
Harjoitus 10, 21.-23.4.2021**

1. (*Kynä & paperi*) Digitaalisen signaalin näytteenottotaajuus on 48 kHz ja se halutaan muuntaa näytteenottotaajuuteen 16 kHz. Signaalista tiedetään, että taajuuksien 0 – 6 kHz tulee säilyä tulossignaalin. Miten määritellään antialias-suotimen päästökaista ja estokaista?
2. (*Kynä & paperi*) Signaali halutaan muuntaa näytteenottotaajuudelta 48 kHz näytteenottotaajuuteen 32 kHz. Piirrä muunnoksen vaiheet lohkokaaavana käyttäen uudelleennäytteistystä ($\uparrow L$) ja ($\downarrow M$) ja alipäästösuodatusta ($H(z)$). Esitä tarvittavien alipäästösuodinten päästö- ja estokaistojen sijainti, kun taajuudet väliltä 0 – 14 kHz halutaan säilyttää.
3. (*Matlab*) Tutkitaan tässä tehtävässä kvantisointikohinaa Matlabilla. Huom. Jos Matlab ei tunne quant-funktiota (eli sinulla ei ole Deep Learning Toolbox asennettuna), lataa vastaava kvant-funktio kurssin Moodlesta (Harj_10.zip).
 - (a) Avaa ensin Matlabin testisignaali komennolla `load gong.mat`, joka lataa sen muuttujaan `y`. Kvantisoi testisignaali 1 + 7 bittiin `quant`-komennolla. Tulosta virhesignaali (kvantisoidun ja kvantisoimattoman erotus) ruudulle ja laske sen varianssi komennolla `var`. Vertaa tätä teoreettiseen tulokseen (kvantisointikohinan varianssin kaavaan).
 - (b) Toista testi kurssin Moodlesta (Harj_10.zip) löytyvälle testisignaalille `seiska.wav`. Mistä erot kvantisointivirheen varianssissa eri testisignaaleille johtuvat?
4. (*Matlab*) Käytettäessä IIR-suotimia äärellisellä laskentatarkkuudella, voi syntyä ilmiö nimeltä *rajasykli*. Tällöin järjestelmän vaste ei herätteen loputtua vaimene vaan jää ikuisesti värähtelemään jollain taajuudella nollan ympärille.

Lataa testiskriptin pohja `rajasykli.m` kurssin Moodlesta (Harj_10.zip). Kirjoita kommentoidun for-silmukan tilalle koodi, jossa muuttujaan `y(n)` sijoitetaan suodatuksen tulos kvantisoinnin jälkeen. Kvantisointi tehdään viiteen bittiin (+merkkibitti) `quant`-funktiolla. Toteutettavan suotimen siirtofunktio on

$$H(z) = \frac{1}{1 + 0.5z^{-1}}.$$

Aja skripti, jolloin suotimen ulostulo tulostuu ruudulle. Vaikka heräte on nollassa heti kahden ensimmäisen näytteen jälkeen, tulos jää värähtelemään ikuisesti (tämän suotimen tapauksessa Nyquistin taajuudella).

5. (*Matlab*) Toteutetaan Matlabilla näytteenottotaajuuden muunnos $8192 \text{ Hz} \rightarrow 6144 \text{ Hz}$.

Muunnos vaatii seuraavat vaiheet:

- (a) Lataa ensin Matlabin testisignaali `handel`
- (b) Muodosta nollasignaali z , jonka pituus on kolminkertainen `handel`iin nähden.
- (c) Sijoita nollasignaalin joka kolmanteen paikkaan alkuperäisen signaalin arvot komennolla `z(1:3:end) = y;` Kuuntele tulos (`soundsc(z, 3*8192)`).
- (d) Suunnittele asteen 100 FIR-suodin (komento `fir1`) ja suodata syntyneet häiriöt pois.
- (e) Kuuntele tulos ja totea, että se kuulostaa samalta kuin alkuperäinen.
- (f) Suunnittele antialias-suodin, jonka aste on myös 100.
- (g) Suodata kohdan (d) tulossignaali antialias-suotimella.
- (h) Ota tuloksesta talteen joka neljäs näyte (indeksoimalla `s(1:4:end)`).
- (i) Kuuntele tulos näytteenottotaajuudella 6144 Hz, joka on $3/4$ alkuperäisestä.
- (j) Tulosta kohtien (a), (c), (d), (g) ja (h) signaalien spektrogrammit komennolla `spectrogram`.