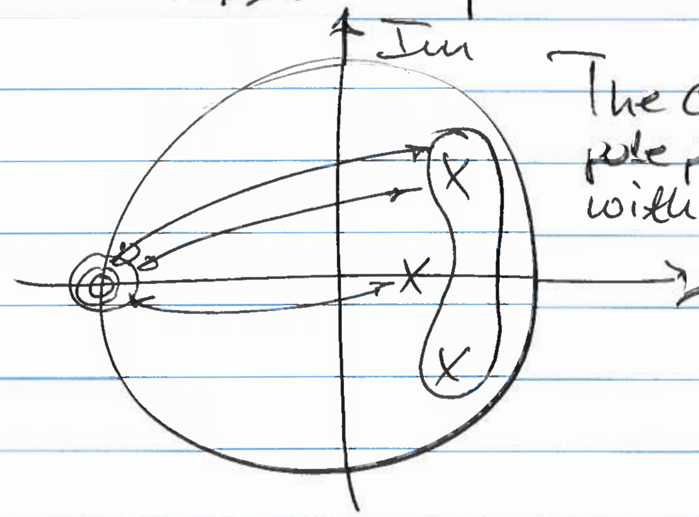
SIGNAL PROCESSING

— SUGGESTED SOLUTIONS

$$H(z) = g \cdot \frac{1+3z^{-1}+3z^{-2}+z^{-3}}{1-1.4590z^{-1}+0.8104z^{-2}-0.1878z^{-3}}$$

Restructure $H(z)$ into one 2^{nd} order section and one 1^{st} order section via factorization.

Pole/Zero - plot



The complex conjugated pole pair is merged with two zeros, and the real pole is merged with the remaining zero.

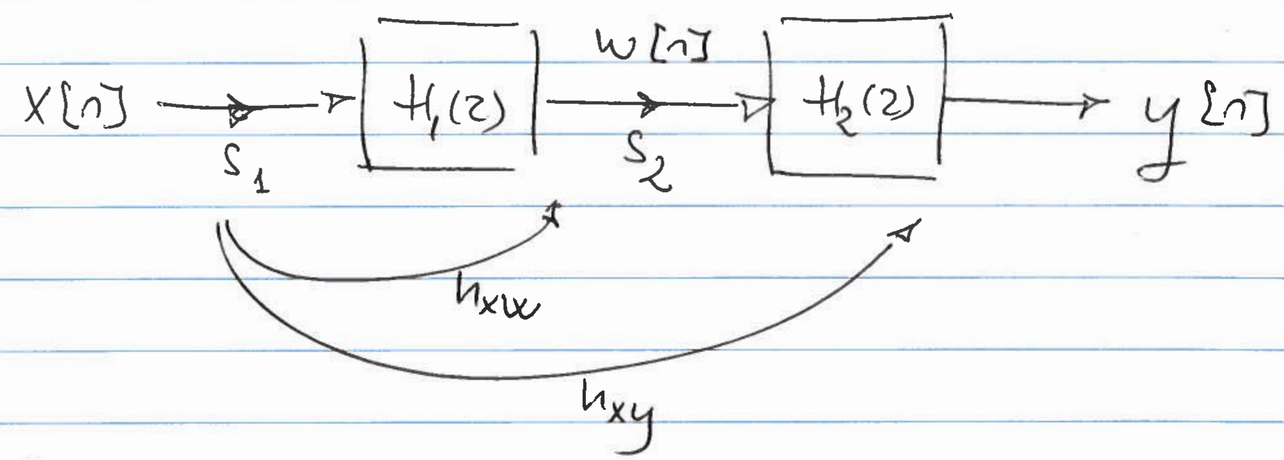
So, we may re-write $H(z)$ as follows;

$$H(z) = \left\{ s_1 \cdot \frac{z+1}{z-0.4142} \right\} \cdot \left\{ s_2 \cdot \frac{(z+1)^2}{(z-(0.5224+j0.4524))(z-(0.5224-j0.4524))} \right\}$$

$\Downarrow$

$$H(z) = \left\{ s_1 \cdot \frac{1+z^{-1}}{1-0.4142z^{-1}} \right\} \cdot \left\{ s_2 \cdot \frac{1+2z^{-1}+z^{-2}}{1-1.0448z^{-1}+0.4776z^{-2}} \right\}$$

Next, lets find s_1 and s_2 using VARIANCE SCALING

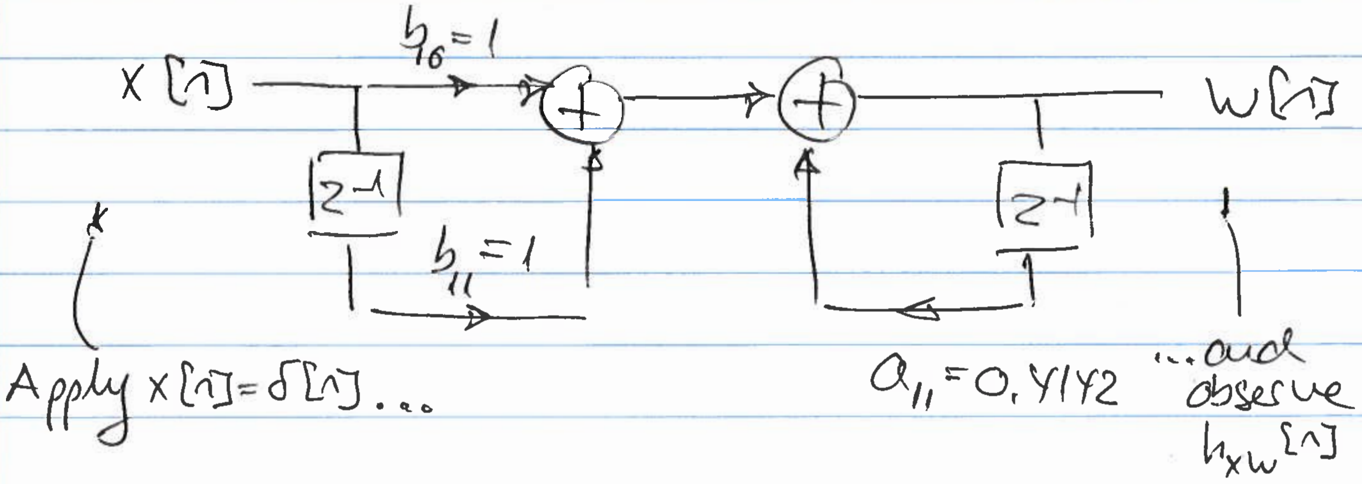


Since $H_1(z)$ and $H_2(z)$ are implemented using DF-I, there is only two variables which are not allowed to overflow; w[n] and y[n]

The impulse responses from the input to w and y, respectively, are denoted $h_{xw}[n]$ and $h_{xy}[n]$

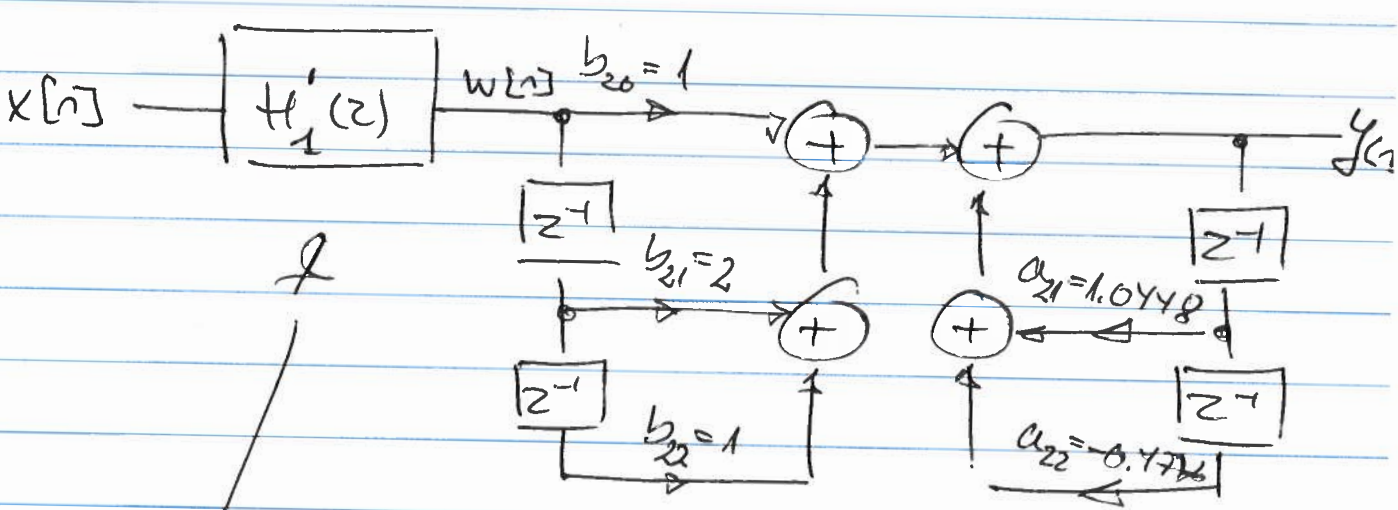
In order to find s_1 we must calculate the squared and summed impulse response h_{xw}

$$s_1 = \frac{1}{\sqrt{\sum_{n=-\infty}^{\infty} |h_{xw}[n]|^2}} \cong \frac{1}{\sqrt{\sum_{n=0}^{K} |h_{xw}[n]|^2}}$$



$$s_1 = \frac{1}{\sqrt{\sum_{0}^{100} h_{xw}^2}} = 0.5412$$ (See Matlab - program)

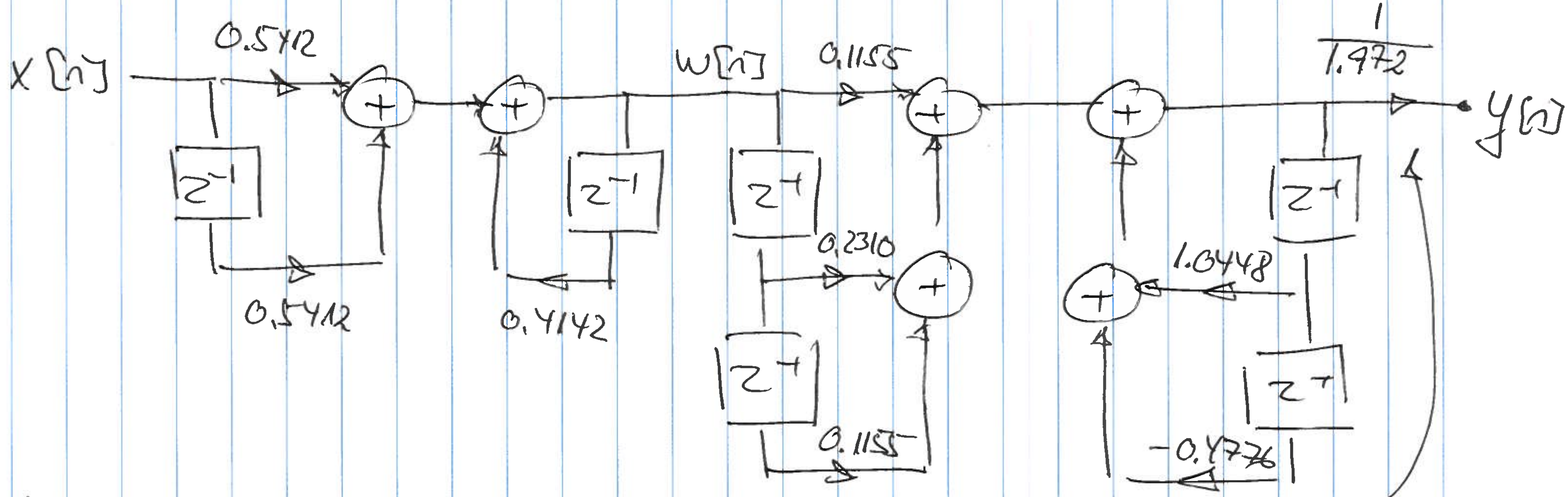
Next we can find s_2 by calculating the squared and summed impulse response from x' to y.



OBS! It is important that you apply the scaled 1st order section here — otherwise the energy in w[n] will not equal 1.

$$s_2 = \frac{1}{\sqrt{\sum_{0}^{100} h_{xy}^2}} = 0.1155$$ (See Matlab. program)

THE COMPLETELY SCALED FILTER STRUCTURE



After scaling, the filter has a DC-gain equal to 1.972 (5,9 dB)
That can be compensated for at the output.

4

```
% Forslag til MATLAB-program som beregner skaleringsfaktoren %
% for 1. ordens sektionen på baggrund af den kvadrerede og   %
% summerede impulsrespons til registeret w.                  %

clear;

% Initialisering af filterkoefficienter %
b10 = 1;
b11 = 1;
a11 = 0.414213;

% Initialiser de interne variable %
w1 = 0;
hw = 0;

% Generer impuls
x(1)=1;
x(2:100)=zeros;

% Beregn kvadreret og summeret impulsrespons. %
% Der medtages ialt 100 samples.              %
for n=1:100,
 if n == 1
   w0 = b10*x(n) + a11*w1;
 else
   w0 = b10*x(n) + b11*x(n-1) + a11*w1;
 end;

 % Impulsresponsen kvadreres og summeres %
 hw = hw + w0^2;

 % Registeret opdateres %
 w1 = w0;
end;

% Beregn og udskriv skaleringsfaktoren %
sw = 1/sqrt(hw)
```

```
% Forslag til MATLAB-program beregner skaleringsfaktoren %
% for 2. ordens sektionen på baggrund af de kvadrerede   %
% og summerede impuls responser til output-registeret y. %
% Bemærk, at vi i denne beregning bliver nødt til at     %
% anvende den skalerede 1. ordens sektion FORAN 2.       %
% ordens sektionen..!!                                   %

clear;

% Initialisering af filterkoefficienter %
b10 = 0.5412;
b11 = 0.5412;
a11 = 0.4142;
b20 = 1;
b21 = 2;
b22 = 1;
a21 = 1.044816;
a22 = -0.477593;

% Initialiser de interne variable %
w1 = 0;
w2 = 0;
y1 = 0;
y2 = 0;
hw = 0;
hy = 0;

% Generer impuls
x(1)=1;
x(2:100)=zeros;

% Beregn kvadreret og summeret impulsrespons. I alt 100 samples. %
for n=1:100,
 if n == 1
   w0 = b10*x(n) + a11*w1;
   y0 = b20*w0 + b21*w1 + b22*w2 + a21*y1 + a22*y2;
 else
   w0 = b10*x(n) + b11*x(n-1) + a11*w1;
   y0 = b20*w0 + b21*w1 + b22*w2 + a21*y1 + a22*y2;
 end;

 % Impulsresponsen kvadreres og summeres %
 hy = hy + y0^2;

 % Registrene opdateres %
 w2 = w1;
 w1 = w0;

 y2 = y1;
 y1 = y0;
end;

% Beregn og udskriv skaleringsfaktoren %
sy = 1/sqrt(hy)
```