Name: Rency Ajit Kansagra

ID: 1001844931

CSE 3313-001

**Report**

* **Problem description**

The objective of this project is to create and execute a digital filter using a microcontroller to efficiently eliminate noise from a provided signal. This requires the calculation of the appropriate filter coefficients A and B and integrating them into the difference equation function of the microcontroller, which has been done in the previous assignment. The filter design should meet the specified design criteria, which include a maximum filter order of N=10 and certain passband (𝛿𝑝= −1 dB) and stopband (𝛿𝑠= −25dB). The project uses MATLAB to calculate these coefficients and evaluate the filter's performance in achieving the desired signal clarity by removing noise at a normalized frequency.

* **Filter Design calculations for finding N and Ω𝑐 from design criteria including justification for the values you chose for 𝜔𝑝 and 𝜔𝑝 (show all calculations)**

Chosen:

𝜔p = 0.09π rad/sec

𝜔s = 0.38 π rad/sec

Given:  
Passband 𝛿𝑝= −1 dB =>Converted to Linear =>10 -1/20 = 0.891251

Stopband 𝛿𝑠=−25dB =>Converted to Linear =>10 -25/20 = 0.0562

Kp = = 0.2596 ≈ 0.260

Ks = = 317.88

Ωp = = = 0.2827

Ωs = = = 1.1938

N = = = 3.13 ≈ 3

**Formulae:**

= Kp => = 0.260 => Ωc = 0.3959

= Ks => = 317.88 => Ωc = 0.2827

𝛿𝑠 = 0.0562 < Ωc = 0.3959 < 𝛿𝑝 = 0.891251

𝛿𝑠 = 0.0562 < Ωc = 0.2827 < 𝛿𝑝 = 0.891251

We can use **Ωc = 0.3959 and N = 3**.

**Reasons for choosing :**

In the digital filter design for this project, 𝜔𝑝=0.09𝜋 rad/sec and 𝜔𝑠=0.38𝜋 rad/sec were selected to optimize filter performance within specific constraints. The passband frequency *ωp*​ includes the desired signal, ensuring it passes through with minimal attenuation, while the stopband frequency *ωs*​ encompasses the noise, effectively filtering it out. These choices also respect the digital frequency limit of *π* rad/sec to prevent issues like aliasing. By setting *ωp*​ and *ωs*​ close enough maintains a filter order less than 10 (𝑁≤10).

* **Your A and B filter coefficients**

A=[1 , -2.2184 , 1.7132 , -0.4530];

B=[0 , 0.0085 , 0.0276 , 0.0057 ];

* **Plot of the magnitude of your digital filter frequency response abs(H) vs digital frequency**

A screenshot of a computer

Description automatically generated

* **main.c -**Coefficients A and B were added along with N.
* **serialfft.m-** butter() was replaced with calculated Coefficients A and B
* **Plot of the microcontroller output with and without filtering applied • Push the blue button to switch between filtering and no filtering**

**Before:**

A screenshot of a computer

Description automatically generated

**After:**

A screenshot of a computer

Description automatically generated

**MATLAB script**

clear all

N = 3;

Oc = 0.3959;

for k = 0:N-1

poles(k+1) = Oc\*exp(1i\*(pi+2\*k\*pi)/(2\*N))\*exp(1i\*(pi/2));

end

B = [0.3959^3];

A = poly(poles);

fprintf('Analog B = %.4f\n', B);

fprintf('Analog A = ');

fprintf('%.4f ', A);

fprintf('\n');

[H,W] = freqs(B,A,1001);

figure(1);

plot(W, abs(H));

title('Transfer Function of Analog Filter H(s)');

xlabel('\Omega');

grid on;

[R, P, K] = residue(B, A);

Hc = tf(B,A);

Hd = c2d(Hc,1);

[num\_d, den\_d] = tfdata(Hd, 'v');

disp('Digital B coefficients:');

disp(num\_d);

disp('Digital A coefficients:');

disp(den\_d);

z = exp(1i\*2\*pi\*(1:101)/101);

Hz = freqresp(Hd,z);

figure(2),plot(2\*pi\*(1:101)/101,abs(squeeze(Hz)))