UNIT 4:

Infinite Impulse Response (IIR): Signal Processing & Digital Control System

A. Dasar Teori

A.1. ***Digital Low Pass Filter (LPF)***

IIR merupakan keluaran dari suatu system dinamis diskret saat diberikan input berupa isyarat impulse. Secara teori memiliki runtun impulse yag tak hingga, karena berbasis feedback yaitu menggunakan data pada waktu sebelumnya untuk menghitung data saat ini. Secara singkat pembahasa IIR didasarkan pada response frekuensi dari system dinamis diskret berikut:

$$y[n] = \sum_{k_1=0}^{N_1} a_{k_1} y[n-k] + \sum_{k_2=0}^{N_2} b_{k_2} x[n-k_2]$$

Atau versi paling sederhana:

$$y[n] = ay[n-1] + bx[n-1]$$

Dengan n adalah indeks waktu diskret, a dan b adalah konstanta, sedangkan $x[n]$ adalah isyarat diskret yang berlaku sebagai input. Untuk kasus signal processing, maka $x[n]$ adalah isyarat yang ingin diproses, sedangkan tugas nya adalah merancang system dinamis nya. Adapun dalam ranah kendali maka $x[n]$ adalah keluaran actuator yang berlaku sebagai input bagi system dinamis tersebut.

Dengan asumsi system kausal maka $y[n < 0] = 0$, maka diperoleh $y[n]$ untuk $n \geq 0$ berikut:

$$y\,[0] = 0$$

$$y[1] = ay[0] + bx[0] = bx[0]$$

$$y[2] = ay[1] + bx[1] = b(ax[0] + x[1])$$

$$y[3] = ay[2] + bx[2] = b(a^2x[0] + ax[1] + x[2])$$

$$y[n] = b\sum_{k=1}^{n} a^{k-1} x[n-k]\,, \text{for } n \geq 0$$

Untuk mengamati impulse response maka input diganti isyarat impulse $x[n] = \delta[n]$ sehingga diperoleh:

$$y[n] = b\sum_{k=1}^{n} a^{k-1}\delta[n-k] = ba^{n-1}\delta[0] = ba^{n-1}$$

Secara keseluruhan diperoleh:

$$y[n] = \begin{cases} 0, & \text{for } n \leq 0 \\ ba^{n-1}, & \text{for } n > 0 \end{cases} \qquad \text{atau} \qquad y[n] = ba^{n-1}u[n-1]$$

Dengan $u[n]$ adalah isyarat unit step dengan $u[n < 0] = 0$ dan $u[n \geq 0] = 1$ sedangkan isyarat unit impulse $\delta[n]$ dengan $\delta[0] = 1$, sedangkan $\delta[n \neq 0] = 1$.

Kemudian jika dilakukan transformasi DTFT, akan diperoleh:

$$Y_d(j\Omega) = \sum_{n=-\infty}^{+\infty} y[n]e^{-j\Omega n} = b\sum_{n=1}^{+\infty} a^{n-1}e^{-j\Omega n} = \frac{b}{a}\sum_{n=1}^{+\infty}\left(ae^{-j\Omega}\right)^n$$

Berupa penjumlahan deret geometry tak hingga, sehingga dapat dituliskan sebagai:

$$Y_d(j\Omega) = \frac{b}{a}\frac{\left(ae^{-j\Omega}\right)^1}{1-(ae^{-j\Omega})} = \frac{b}{e^{j\Omega} - a}$$

Magnitude nya bernilai:

$$|Y_d(j\Omega)| = \frac{b}{\sqrt{1 + a^2 - 2a\cos(\Omega)}}$$

Namun, pemahaman mengenai filtering langsung pada analisa diskret cukup sulit memaknai nilai frekuensi yang ingin difilter mengingat pada kondisi diskret, frekuensi Ω berkisar pada $-2\pi \leq \Omega \leq 2\pi$, sehingga perlu dianalisa dari perancangan filter analog.

Berdasarkan pembahasan pada sub bab CTFT pada UNIT 2 FFT, diperoleh bahwa struktur filter analog untuk LPF adalah berikut:

$$\frac{Y(j\omega)}{X(j\omega)} = \frac{\omega_c}{j\omega + \omega_c}$$

Jika dilihat dari persamaan di domain waktunya diperoleh:

$$j\omega\, Y(j\omega) = -\omega_c Y(j\omega) + \omega_c X(j\omega)$$

Melalui inverse Fourier Transform diperoleh:

$$\frac{dy(t)}{dt} = -\omega_c\, y(t) + \omega_c x(t)$$

$$e^{\omega_c t}\frac{dy(t)}{dt} + \omega_c e^{\omega_c t} y(t) = \omega_c e^{\omega_c t} x(t)$$

$$\frac{d\{e^{\omega_c t}y(t)\}}{dt} = \omega_c e^{\omega_c t}x(t) \rightarrow\ e^{\omega_c t}y(t) - y(0) = \int_0^t \omega_c e^{\omega_c t}x(t)dt$$

$$y(t) = y(0)e^{-\omega_c t} + e^{-\omega_c t}\int_0^t \omega_c e^{\omega_c t}x(t)dt$$

Dengan mengamati step response nya yaitu saat $x(t) = u(t)$ akan diperoleh:

$$y(t) = y(0)e^{-\omega_c t} + \omega_c e^{-\omega_c t} \int_0^t e^{\omega_c t}\, dt$$

$$y(t) = y(0)e^{-\omega_c t} + (1 - e^{-\omega_c t}), \quad \text{untuk } t > 0$$

Adapun saat $t < 0$ maka $y(t) = 0$, sehingga dapat dituliskan:

$$y(t) = \{1 + (y(0) - 1)e^{-\omega_c t}\}u(t)$$

Kemudian untuk zero initial condition akan diperoleh:

$$y(t) = (1 - e^{-\omega_c t})u(t)$$

Jika output $y(t)$ kemudian dicuplik dengan periode cuplikan T_s, maka dapat dituliskan hasil cuplikannya sebagai:

$$y[k] = y(kT_s) = (1 - e^{-\omega_c k T_s})u(kT_s) = (1 - e^{-k\omega_c T_s})u[k]$$

Menggunakan DTFT diperoleh spectrum di domain frekuensi nya sebagai:

$$Y(j\Omega) \triangleq \mathcal{F}\{y[k]\} = \sum_{k=-\infty}^{\infty} (1 - e^{-k\omega_c T_s})u[k]e^{-j\Omega k}$$

$$Y(j\Omega) = \sum_{k=0}^{\infty} \left(e^{-j\Omega k} - e^{-k\{\omega_c T_s + j\Omega\}}\right)$$

Yang membentuk penjumlahan deret geometri tak hingga, sehingga dapat dituliskan sebagai:

$$Y(j\Omega) = \frac{1}{1 - e^{-j\Omega}} - \frac{1}{1 - e^{-\omega_c T_s} e^{-j\Omega}}$$

Perlu diingat bahwa transfer function di atas adalah transfer function hasil step response di system kontinyu. Padahal perancangan filter harus focus pada struktur intrinsic dari system filter tersebut tidak boleh ada pengaruh external (yang pada kasus ini berupa input step yang sudah tergabung ke dalam system). Sehingga diperlukan proses inversion dengan mengalikan transfer function di atas dengan inverse dari system yang mimick pada isyarat step. Untuk mengetahuinya, perlu diingat bahwa isyarat step adalah **impulse** response dari system diskret berikut:

$$w[n] = \sum_{k=-\infty}^{n} x[k] \rightarrow u[n] = \sum_{k=-\infty}^{n} \delta[k]$$

Dengan $w[n]$ adalah output sedangkan $x[n]$ adalah input, sehingga untuk mencari system inverse nya adalah dengan mencari fungsi dari $w[n]$ yang menghasilkan $x[n]$. Jika ditulsikan:

$$w[n] = x[n] + \sum_{k=-\infty}^{n-1} x[k] \rightarrow w[n] = x[n] + w[n-1] \rightarrow x[n] = w[n] - w[n-1]$$

Jika system inverse dinotasikan untuk inputnya adalah $v_i[n]$ sedangkan outputnya adalah $v_o[n]$, maka system yang memiliki **impulse** response berupa unit step memiliki system inverse berikut:

$$v_o[n] = v_i[n] - v_i[n-1]$$

Sehingga impulse response dari system inverse tersebut adalah:

$$h_o[n] = \delta[n] - \delta[n-1]$$

Jika dilakukan DTFT akan diberikan:

$$H_o(j\Omega) = \sum_{n=-\infty}^{+\infty} (\delta[n] - \delta[n-1])e^{-j\Omega n} = e^{-j\Omega.0} - e^{-j\Omega.1} = 1 - e^{-j\Omega}$$

Sehingga transfer function filter nya menjadi:

$$H_f(j\Omega) = Y(j\Omega)H_o(j\Omega)$$

$$H_f(j\Omega) = \left\{\frac{1}{1-e^{-j\Omega}} - \frac{1}{1-e^{-\omega_c T_s}e^{-j\Omega}}\right\}\left\{1-e^{-j\Omega}\right\} = 1 - \frac{1-e^{-j\Omega}}{1-e^{-\omega_c T_s}e^{-j\Omega}}$$

$$H_f(j\Omega) = \frac{1-e^{-\omega_c T_s}}{e^{j\Omega} - e^{-\omega_c T_s}}$$

Kesemua proses tersebut kemudian dapat dengan sederhana dituliskan sebagai:

$$H_f(j\Omega) = \left[(1-z^{-1})\,\mathcal{Z}\left\{\mathcal{L}^{-1}\left\{\frac{H(s)}{s}\right\}_{t=kT_s}\right\}\right]_{z=e^{j\Omega}}$$

Yang dikenal sebagai **Zero-Order-Hold (ZOH)**.

Kembali ke bentuk $H_f(j\Omega)$, untuk memudahkan definisikan $z \triangleq e^{j\Omega}$ maka dapat ditulis:

$$H_f(z) = \frac{1-e^{-\omega_c T_s}}{z-e^{-\omega_c T_s}}$$

Yang kemudian disebut sebagai **Transformasi Z**. Sebenarnya untuk transformasi Z, secara general $z \triangleq r\,e^{j\Omega}$ tidak hanya untuk radius satu namun berapapun radiusnya di bidang complex. Namun seperti hal nya Fourier Transform yang hanya focus pada sumbu imaginary di bidang complex, yang merupakan versi khusus dari Laplace Transform, DTFT juga merupakan versi khusus dari transformasi Z. CTFT dan DTFT digunakan dalam bidang signal processing yang berfokus pada sinyal, sedangkan Laplace dan Z digunakan dalam bidang control engineering yang berfokus pada system.

Sehingga berdasarkan formula DTFT, dapat juga dirumuskan transformasi Z sebagai:

$$W(z) = \sum_{k=-\infty}^{+\infty} w[k]z^{-k}$$

Berdasarkan $Y_d(j\Omega)$ jika dituliskan ke dalam notasi z dan diequivalentkan dengan $H_f(z)$ maka:

$$Y_d(z) = \frac{b}{z-a} \equiv H_f(z) = \frac{1-e^{-\omega_c T_s}}{z-e^{-\omega_c T_s}} = \frac{(1-e^{-\omega_c T_s})z^{-1}}{1-e^{-\omega_c T_s}z^{-1}} \equiv \frac{Y(z)}{X(z)}$$

Maka diperoleh $a = e^{-\omega_c T_s}$ sedangkan $b = 1 - e^{-\omega_c T_s}$ sehingga system digital yang bekerja sebagai LPF dengan frekuensi cut off ω_c rad/s adalah:

$$y[n] = e^{-\omega_c T_s} y[n-1] + (1 - e^{-\omega_c T_s})x[n-1]$$

A.2. Digital High Pass Filter (HPF)

Transfer function kontinyu untuk HPF diberikan oleh:

$$\frac{Y(j\omega)}{X(j\omega)} = \frac{\frac{j\omega}{\omega_c}}{\frac{j\omega}{\omega_c}+1} = \frac{j\omega}{j\omega+\omega_c} \to \ H(s) = \frac{s}{s+\omega_c}$$

Maka menggunakan ZOH diperoleh:

$$H_f(z) = (1-z^{-1})\ \mathcal{Z}\left\{\mathcal{L}^{-1}\left\{\frac{H(s)}{s}\right\}_{t=kT_s}\right\}$$

$$H_f(z) = (1-z^{-1})\mathcal{Z}\left\{\mathcal{L}^{-1}\left\{\frac{1}{s+\omega_c}\right\}_{t=kT_s}\right\} = (1-z^{-1})\mathcal{Z}\left\{\{e^{-\omega_c t}u(t)\}_{t=kT_s}\right\}$$

$$H_f(z) = \ (1-z^{-1})\mathcal{Z}\{e^{-\omega_c kT_s}u(kT_s)\} = (1-z^{-1})\mathcal{Z}\{e^{-\omega_c kT_s}u[k]\} = \left(\frac{z-1}{z}\right)\left(\frac{z}{z-e^{-\omega_c T_s}}\right)$$

$$H_f(z) = \frac{1-z^{-1}}{1-e^{-\omega_c T_s}z^{-1}} \equiv \frac{Y(z)}{X(z)}$$

Sehingga digital filter untuk HPF diberikan sebagai:

$$y[n] = e^{-\omega_c T_s}y[n-1] - x[n-1] + x[n]$$

A.3. Digital Band Stop Filter (BSF)

Transfer function kontinyu untuk BSF diberikan oleh:

$$H(s) \triangleq \frac{Y(s)}{X(s)} = K\frac{(s+\omega_2)(s+\omega_3)}{(s+\omega_1)(s+\omega_4)}$$

Dengan $\omega_1 < \omega_2 < \omega_3 < \omega_4$, dengan $K = \frac{\omega_1\omega_4}{\omega_2\omega_3}$

$$H(s) = K\frac{s^2 + (\omega_2 + \omega_3)s + \omega_2\omega_3}{s^2 + (\omega_1 + \omega_4)s + \omega_1\omega_4} = K + K\frac{(\omega_2 + \omega_3 - \omega_1 - \omega_4)s + (\omega_2\omega_3 - \omega_1\omega_4)}{s^2 + (\omega_1 + \omega_4)s + \omega_1\omega_4}$$

untuk menyederhanakan notasi dengan mendefinisikan $\eta_1 = (\omega_2 + \omega_3 - \omega_1 - \omega_4)$ dan $\eta_2 = (\omega_2\omega_3 - \omega_1\omega_4)$, maka:

$$\frac{H(s)}{s} = \frac{K}{s} + \frac{K\eta_1 s + K\eta_2}{s(s + \omega_1)(s + \omega_4)} \equiv \frac{A}{s} + \frac{B}{(s + \omega_1)} + \frac{C}{(s + \omega_4)}$$

Maka menggunakan Partial Fraction Expansion (PFE), saat $s = 0$ maka $A = K\left(1 + \frac{\eta_2}{\omega_1\omega_4}\right)$, saat $s = -\omega_1$ maka $B = K\frac{\omega_1\eta_1 - \eta_2}{\omega_1(\omega_4 - \omega_1)}$, saat $s = -\omega_4$ maka $C = K\frac{(-\omega_4\eta_1 + \eta_2)}{\omega_4(\omega_4 - \omega_1)}$, maka diperoleh:

$$\mathcal{L}^{-1}\left\{\frac{H(s)}{s}\right\} = \{A + Be^{-\omega_1 t} + Ce^{-\omega_4 t}\}u(t)$$

$$\mathcal{L}^{-1}\left\{\frac{H(s)}{s}\right\}_{t = kT_s} = \{A + Be^{-\omega_1 kT_s} + Ce^{-\omega_4 kT_s}\}u(kT_s)$$

$$H(z) = (1 - z^{-1})\mathcal{Z}\left\{\{A + Be^{-\omega_1 kT_s} + Ce^{-\omega_4 kT_s}\}u[k]\right\}$$

$$H(z) = (1 - z^{-1})\left\{\frac{A}{1 - z^{-1}} + \frac{B}{1 - e^{-\omega_1 T_s}z^{-1}} + \frac{C}{1 - e^{-\omega_4 T_s}z^{-1}}\right\}$$

$$H(z) = (1 - z^{-1})\left\{\frac{A + B - z^{-1}(B + Ae^{-\omega_1 T_s})}{(1 - z^{-1})(1 - e^{-\omega_1 T_s}z^{-1})} + \frac{C}{1 - e^{-\omega_4 T_s}z^{-1}}\right\}$$

$$\frac{H(z)}{(1 - z^{-1})} = \frac{(A + B + C) - \{(B + Ae^{-\omega_1 T_s}) + (A + B)e^{-\omega_4 T_s} + C(1 + e^{-\omega_1 T_s})\}z^{-1} + \{Ce^{-\omega_1 T_s} + (B + Ae^{-\omega_1 T_s})e^{-\omega_4 T_s}}{(1 - z^{-1})(1 - e^{-\omega_1 T_s}z^{-1})(1 - e^{-\omega_4 T_s}z^{-1})}$$

$$H(z) = \frac{(A + B + C) - \{(B + Ae^{-\omega_1 T_s}) + (A + B)e^{-\omega_4 T_s} + C(1 + e^{-\omega_1 T_s})\}z^{-1} + \{Ce^{-\omega_1 T_s} + (B + Ae^{-\omega_1 T_s})e^{-\omega_4 T_s}}{1 - (e^{-\omega_1 T_s} + e^{-\omega_4 T_s})z^{-1} + e^{-(\omega_1 + \omega_4)T_s}z^{-2}}$$

Untuk simplifikasi definisikan

$$\omega_1 < \omega_2 < \omega_3 < \omega_4$$

$$\eta_1 = (\omega_2 + \omega_3 - \omega_1 - \omega_4), \qquad \eta_2 = (\omega_2\omega_3 - \omega_1\omega_4), K = 1$$

$$A = K\left(1 + \frac{\eta_2}{\omega_1\omega_4}\right), B = K\frac{\omega_1\eta_1 - \eta_2}{\omega_1(\omega_4 - \omega_1)}, C = K\frac{(-\omega_4\eta_1 + \eta_2)}{\omega_4(\omega_4 - \omega_1)}$$

$\gamma_0 = (A + B + C), \gamma_1 = -\{(B + Ae^{-\omega_1 T_s}) + (A + B)e^{-\omega_4 T_s} + C(1 + e^{-\omega_1 T_s})\}$,

$\gamma_2 = \{Ce^{-\omega_1 T_s} + (B + Ae^{-\omega_1 T_s})e^{-\omega_4 T_s}\}$, $\alpha_1 = -(e^{-\omega_1 T_s} + e^{-\omega_4 T_s})$ dan $\alpha_2 = e^{-(\omega_1 + \omega_4)T_s}$

maka dapat dituliskan:

$$\frac{Y(z)}{X(z)} \equiv H(z) = \frac{\gamma_0 + \gamma_1 z^{-1} + \gamma_2 z^{-2}}{1 + \alpha_1 z^{-1} + \alpha_2 z^{-2}}$$

Sehingga diperoleh struktur dinamika BSF sebagai:

$$y[n] = -\sum_{m=1}^{2} \alpha_m y[n-m] + \sum_{p=0}^{2} \gamma_p x[n-p]$$

A.4. Digital Band Pass Filter (BPF)

Struktur transfer function BPF sama dengan BSF, namun untuk bentuk:

$$H(s) \triangleq \frac{Y(s)}{X(s)} = K \frac{(s+\omega_2)(s+\omega_3)}{(s+\omega_1)(s+\omega_4)}$$

Maka harus dengan urutan berikut $\omega_2 < \omega_1 < \omega_4 < \omega_3$.

A.5. Digital PID Control Design in First Order System

Kembali ke bagian A.1. mengingat system berikut:

$$y[n] = ay[n-1] + bx[n-1]$$

Memberikan impulse response berupa:

$$y[n] = ba^{n-1}u[n-1]$$

Untuk $n \to \infty$, nilai $y[n] \to \infty$ juga kecuali saat $|a| \leq 1$, sehingga untuk $|a| > 1$ sistem akan tidak stabil, sedangkan untuk $|a| \leq 1$ sistem akan menuju nol (stabil).

Jika dilihat dari domain Z, maka system tersebut memiliki transfer function berikut:

$$H(z) = \frac{b}{z-a}$$

Sehingga nilai a di sini adalah pole dari system tersebut. Menariknya pada kasus system digital ini, nilai pole positive tidak mengakibatkan ketidakstabilan asalkan tidak lebih dari 1. Sehingga berbeda dengan system kontinyu (kausal) yang kestabilan nya ditentukan oleh pole yang harus berada di LHP, untuk system digital agar stabil, pole harus berada di **unit circle**.

Sehingga misalkan dirancang system yang tidak stabil dengan transfer function berikut:

$$H(z) = \frac{b}{z-2}$$

Agar stabil, dapat diberikan kendali closed loop PID. Namun terlebih dahulu perlu melakukan operasi ZOH pada PID kontinyu yang memiliki struktr berikut, dengan $C(s)$ adalah keluaran actuator yang menjadi masukan ke system, sedangkan $E(s)$ adalah error $E(s) = R(s) - Y(s)$ antara reference $R(s)$ dengan keluaran system (sensor) $Y(s)$.

$$\frac{C(s)}{E(s)} = K_p\left(1 + \frac{1}{T_i s} + T_d s\right)$$

$$\frac{C(z)}{E(z)} = (1 - z^{-1})\mathcal{Z}\left\{\mathcal{L}^{-1}\left\{\frac{U(s)}{s}\right\}_{t=kT_s}\right\}$$

$$\frac{C(z)}{E(z)} = (1 - z^{-1})\mathcal{Z}\left\{\mathcal{L}^{-1}\left\{K_p\left(\frac{1}{s} + \frac{1}{T_i s^2} + T_d\right)\right\}_{t=kT_s}\right\}$$

$$\frac{C(z)}{E(z)} = (1 - z^{-1})\mathcal{Z}\left\{K_p\left(u[k] + \frac{1}{T_i}kT_s u[k] + T_d\delta[k]\right)\right\}$$

Perlu catatan untuk $\mathcal{Z}\{ku[k]\}$ berikut:

$$\mathcal{Z}\{ku[k]\} = \sum_{k=-\infty}^{+\infty} ku[k]z^{-k} = \sum_{k=0}^{+\infty} ku[k]z^{-k}$$

$$\mathcal{Z}\{ku[k]\} = 0 + z^{-1} + 2z^{-2} + 3z^{-3} + 4z^{-4} + 5z^{-5} + \cdots$$

$$\mathcal{Z}\{ku[k]\} = (z^{-1} + z^{-2} + z^{-3} + \cdots) + (z^{-2} + z^{-3} + \cdots) + (z^{-3} + z^{-4} + z^{-5} + \cdots) + \cdots$$

$$\mathcal{Z}\{ku[k]\} = \frac{z^{-1}}{1 - z^{-1}} + \frac{z^{-2}}{1 - z^{-1}} + \frac{z^{-3}}{1 - z^{-1}} + \cdots$$

$$\mathcal{Z}\{ku[k]\} = \frac{1}{1 - z^{-1}}\frac{z^{-1}}{1 - z^{-1}} = \frac{z^{-1}}{(1 - z^{-1})^2}$$

$$\frac{C(z)}{E(z)} = (1 - z^{-1})K_p\left\{\frac{1}{(1 - z^{-1})} + \frac{z^{-1}}{(1 - z^{-1})^2}\frac{T_s}{T_i} + T_d\right\}$$

$$C(z) = K_p\left\{1 + \frac{1}{z - 1}\frac{T_s}{T_i} + T_d(1 - z^{-1})\right\}E(z)$$

$$c[n] = K_p\left\{e[n] + \frac{T_s}{T_i}\sum_{k=1}^{\infty} e[n-k] + T_d(e[n] - e[n-1])\right\}$$

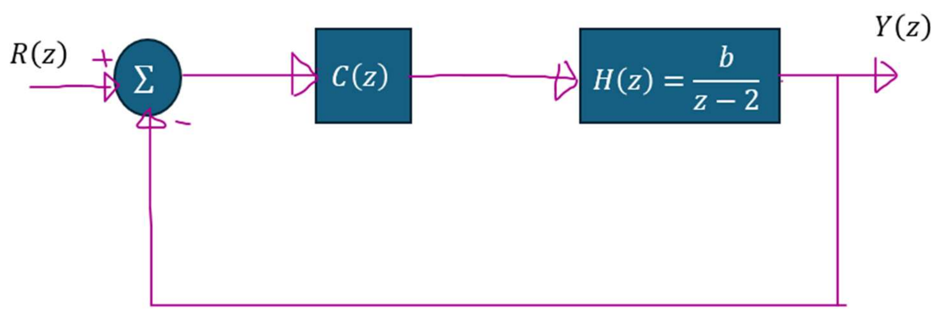
$$\frac{C(z)}{E(z)} = K_p\left\{\frac{1 - z^{-1} + \frac{z^{-1}T_s}{T_i} + T_d - 2T_d z^{-1} + T_d z^{-2}}{(1 - z^{-1})}\right\}$$

$$\frac{C(z)}{E(z)} = K_p \left\{ \frac{(1+T_d) + \left(\frac{T_s}{T_i} - 1 - 2T_d\right) z^{-1} + T_d z^{-2}}{(1-z^{-1})} \right\}$$

Sehingga dalam implementasi pada time domain diberikan:

$$c[n] = c[n-1] + K_p(1+T_d)e[n] + K_p\left(\frac{T_s}{T_i} - 1 - 2T_d\right)e[n-1] + K_pT_de[n-2]$$

Jika system feedback dirancang dengan struktur berikut:



Kemudian untuk simplifikasi dituliskan $\beta_0 = K_p(1+T_d)$, $\beta_1 = K_p\left(\frac{T_s}{T_i} - 1 - 2T_d\right)$, dan $\beta_2 = K_pT_d$ maka:

$$C(z) = \frac{\beta_0 + \beta_1 z^{-1} + \beta_2 z^{-2}}{1 - z^{-1}}$$

sehingga closed loop transfer function nya diberikan:

$$Y = CH(R - Y) \rightarrow \frac{Y}{R} = \frac{CH}{1 + CH}$$

$$\frac{Y}{R} = \frac{\left(\frac{\beta_0 + \beta_1 z^{-1} + \beta_2 z^{-2}}{1 - z^{-1}}\right)\left(\frac{bz^{-1}}{1 - az^{-1}}\right)}{1 + \left(\frac{\beta_0 + \beta_1 z^{-1} + \beta_2 z^{-2}}{1 - z^{-1}}\right)\left(\frac{bz^{-1}}{1 - az^{-1}}\right)}$$

$$\frac{Y}{R} = \frac{(\beta_0 + \beta_1 z^{-1} + \beta_2 z^{-2})(bz^{-1})}{(1 - az^{-1})(1 - z^{-1}) + (\beta_0 + \beta_1 z^{-1} + \beta_2 z^{-2})(bz^{-1})}$$

$$\frac{Y}{R} = \frac{\beta_0 bz^{-1} + \beta_1 bz^{-2} + \beta_2 bz^{-3}}{1 + z^{-1}(-a - 1 + \beta_0 b) + z^{-2}(a + \beta_1 b) + \beta_2 bz^{-3}}$$

$$\frac{Y}{R} = \frac{\beta_0 bz^2 + \beta_1 bz + \beta_2 b}{z^3 + z^2(-a - 1 + \beta_0 b) + z(a + \beta_1 b) + \beta_2 b}$$

$$\frac{Y}{R} = \frac{b\beta_2 + b\beta_1 z + b\beta_0 z^2}{b\beta_2 + (b\beta_1 + a)z + z^2\big(b\beta_0 - (a+1)\big) + z^3}$$

$$\frac{Y}{R} = \frac{b\beta_2 + b\beta_1 z + b\beta_0 z^2}{b\beta_2 + (b\beta_1 + a)z + z^2\big(b\beta_0 - (a+1)\big) + z^3}$$

Menggunakan metode pole placement, misalkan diinginkan closed loop system memiliki pole:

$$z_{1,2} = -0.3 \pm j0.4 \text{ and } z_3 = 0.4$$

Maka characteristic equation nya menjadi:

$$\prod_{k=m}^{3}(z - z_m) = \{z^2 + 0.6z + 0.25\}\{z - 0.4\} = z^3 + 0.2z^2 + 0.01z - 0.1$$

Sehingga diperoleh dengan kesamaan posisi:

$$\beta_{2des} = -\frac{0.1}{b}, \beta_{1des} = \frac{0.01 - a}{b}, \beta_{0des} = \frac{0.2 + a + 1}{b}$$

Ingat bahwa $\beta_{0des} = K_{pdes} + \beta_{2des} \rightarrow K_{pdes} = \beta_{0des} - \beta_{2des}$ sedangkan $T_{ddes} = \frac{\beta_{2des}}{K_{pdes}}$ dan

$$T_i = \frac{T_s}{\beta_{1des} + K_{pdes} + 2\beta_{2des}}$$

A.6. Vieta Formula

Pada pole placement dibutuhkan *desired characteristic equation* yang dari desired poles. Untukmenghindari kalkulasi secara manual, berikut algoritma Vieta Formula untuk implementasi dalam pemprograman untuk konversi desired poles menjadi charactersitice equation. Untuk desired poles nya adalah p_1, p_2, p_3 maka diperoleh desired characteristic equation:

$$z^3 + q_2 z^2 + q_1 z + q_0 = 0$$

Dengan

$$\begin{bmatrix} 1 \\ q_2 \\ q_1 \\ q_0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -p_3 & 1 & 0 \\ 0 & -p_3 & 1 \\ 0 & 0 & -p_3 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -p_2 & 1 \\ 0 & -p_2 \end{bmatrix} \begin{bmatrix} 1 \\ -p_1 \end{bmatrix}$$

Begitu selanjutnya untuk orde yang lebih besar. Sehingga gain pegendali dapat langsung diperoleh dengan:

$$\beta_{2des} = \frac{q_0}{b}, \beta_{1des} = \frac{q_1 - a}{b}, \beta_{0des} = \frac{q_2 + a + 1}{b}$$

Ingat bahwa $\beta_{0des} = K_{pdes} + \beta_{2des} \rightarrow K_{pdes} = \beta_{0des} - \beta_{2des}$ sedangkan $T_{ddes} = \frac{\beta_{2des}}{K_{pdes}}$ dan

$$T_i = \frac{T_s}{\beta_{1des} + K_{pdes} + 2\beta_{2des}}$$

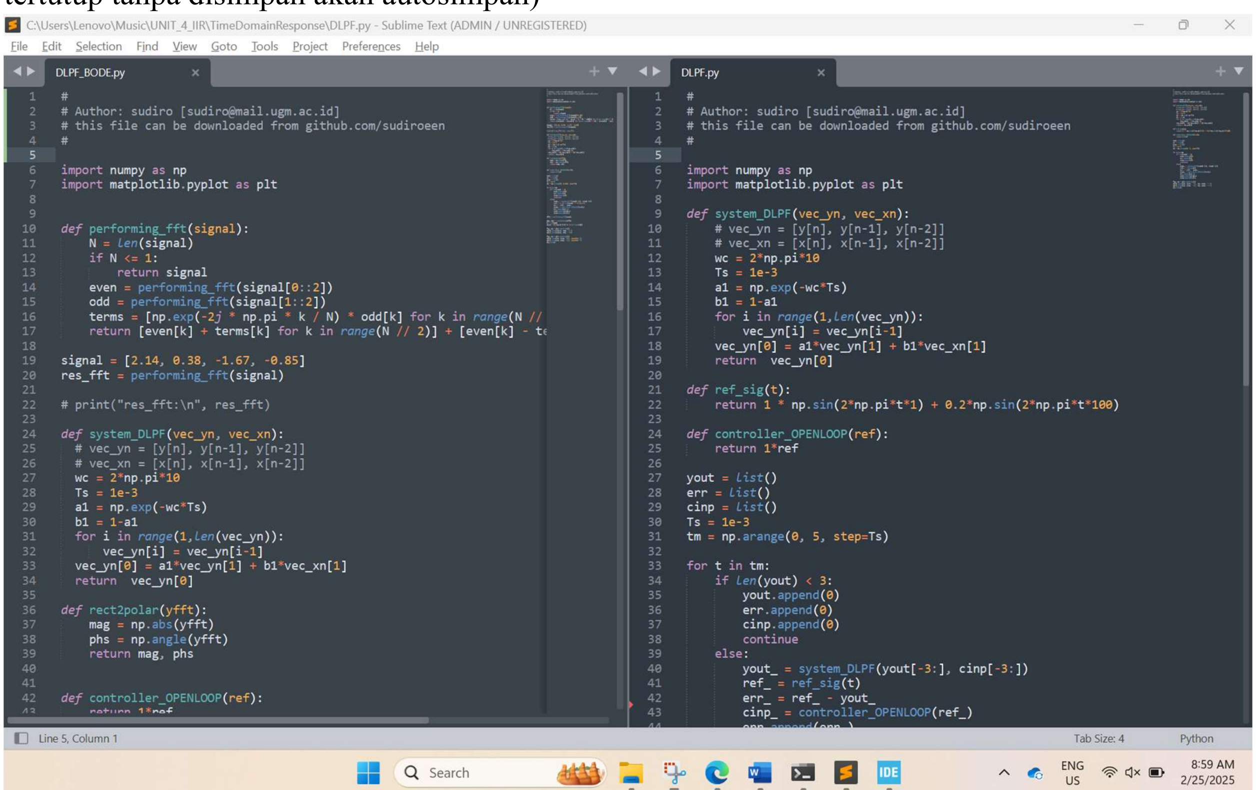
B. Percobaan

- Percobaan dilakukan dalam dua kondisi yaitu mengamati bode plot dari magnitude dan phase response dari FFT dari impulse response dari transfer function masing – masing filter.
- Semua code python yang dibutuhkan dapat diunduh dari https://github.com/sudiroeen/InfiniteImpulseResponse_DTETIsenpro
- File dengan nama berakhiran _BODE.py digunakan untuk mengamati impulse response dan frequency response dari masing – masing filter
- File dengan nama DPLF.py dan seterusnya tanpa imbuhan lain di belakang Namanya digunakan untuk mengamati time response dari masing – masing filter nya untuk memfiltering isyarat sesuai bandwidth frekuensi yang diinginkan

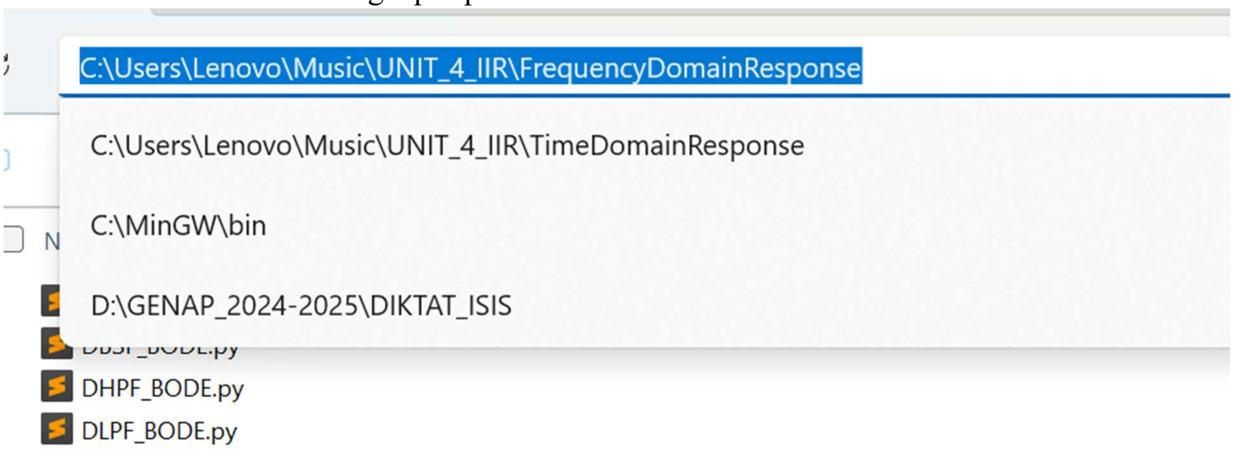
B.1. Impulse Response, Bode Diagram,

Karena ketiga pengujian ini menggunakan code Python, maka berikut Langkah – Langkah running program python:

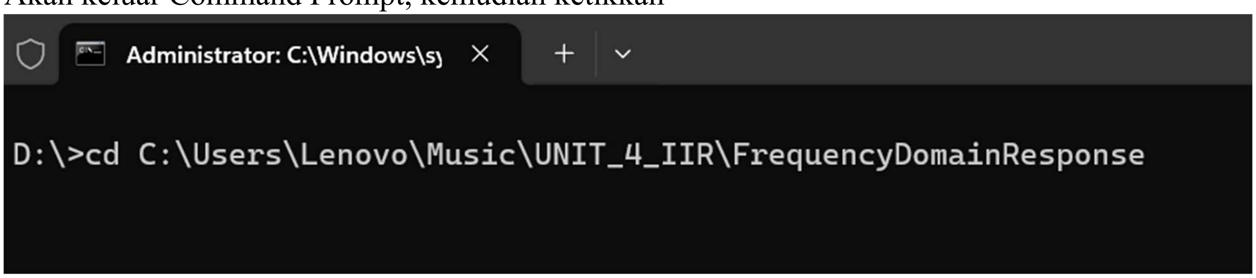
1. Download semua file yang ada di link github di atas.
2. Untuk mengedit file dapat menggunakan Sublime (disarankan karena saat tidak sengaja tertutup tanpa disimpan akan autosimpan)



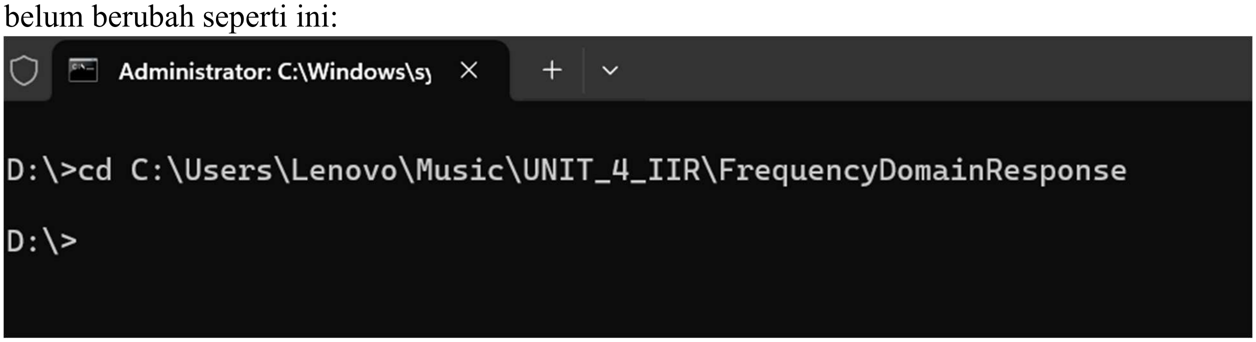
3. Buka folder “FrequencyDomainResponse/” untuk running program Impulse Response dan Bode Diagram
4. Klik nama Lokasi secara lengkap seperti berikut:



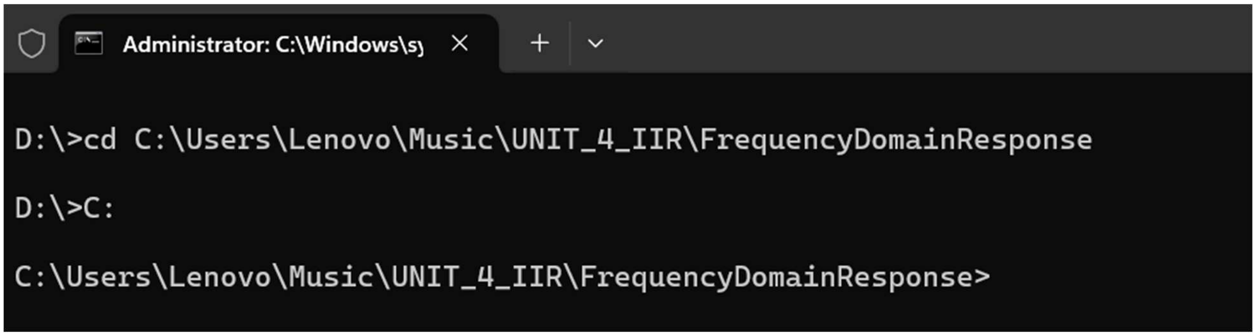
5. Klik tombol Windows + R secara bebarengan, kemudian ketikkan “cmd” kemudian klik tombol “OK”
6. Akan keluar Command Prompt, kemudian ketikkan



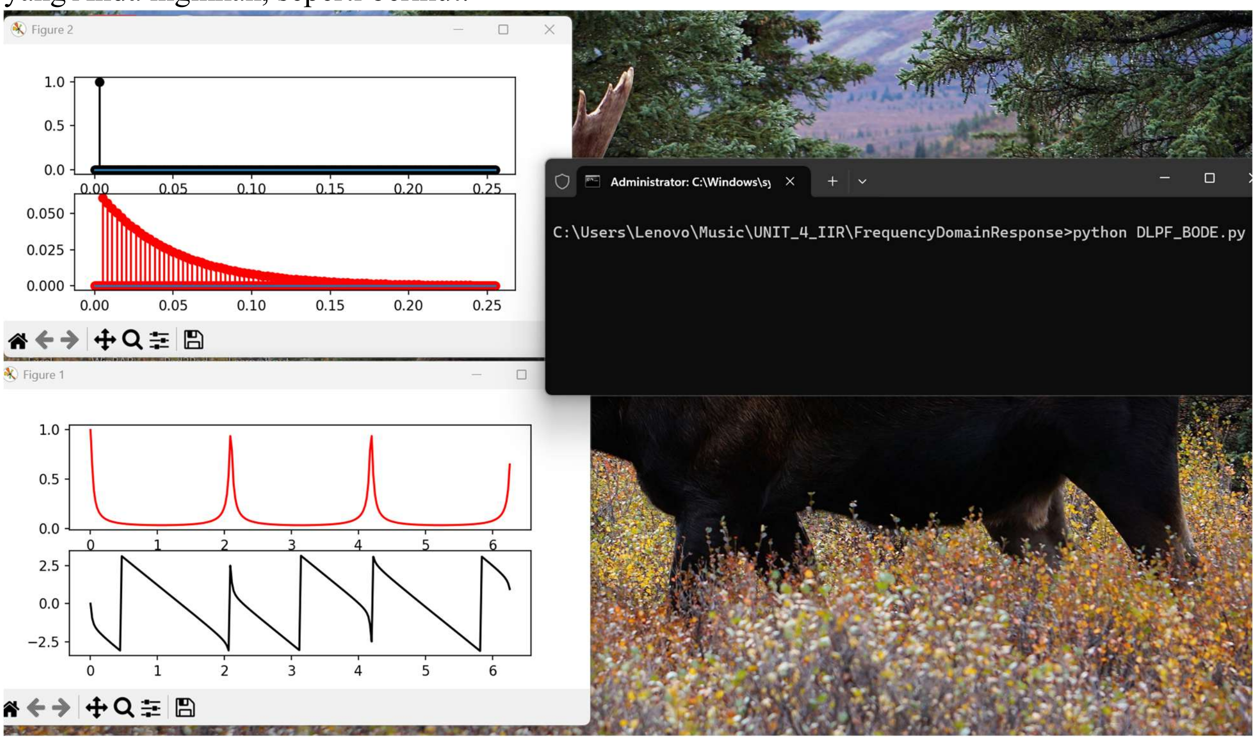
7. Kemudian klik tombol Enter di keyboard Anda. Jika Lokasi di Command Prompt Anda belum berubah seperti ini:



Maka ketik “C:” kemudian enter seperti ini:



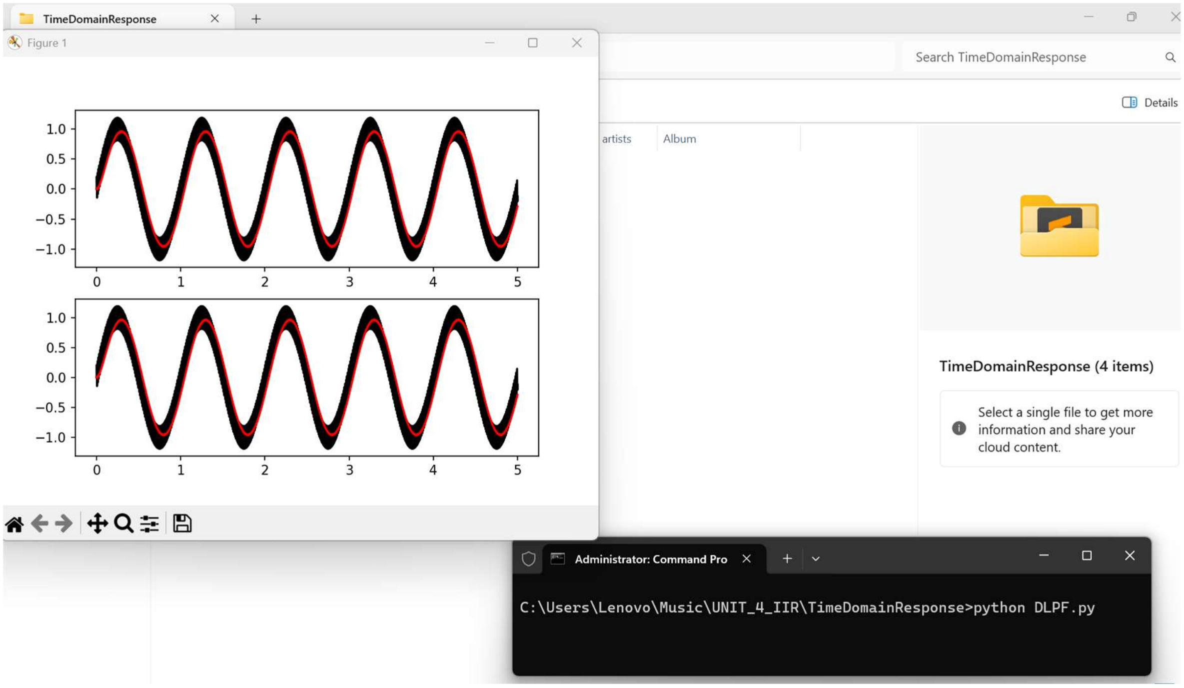
8. Setelah berhasil, kemudian ketik “python<spasi> namaFile.py” untuk running program yang Anda inginkan, seperti berikut:



Proses running berhasil jika tertampil dua figure dengan masing – masing figure ada dua frame plot seperti di atas.

B.2. Time Response dari Filter

1. Dengan cara setup Python yang sama seperti pada sub-section B.1, kemudian ke Lokasi berikut
2. Jika running DLPF sebagai contoh, hasilnya akan berikut:



Dengan sinyal berwarna hitam adalah sinyal sinusoidal dengan noise, sedangkan sinyal merah adalah sinyal sinusoidal tanpa noise.

B.3. Pengujian dengan STM32CubeIDE

1. Gunakan DMA Interrupt sesuai di Modul Unit 1 Percobaan 3 Tugas 1.

2. Untuk implementasi di STM32CubeIDE gunakan code C bantuan pada file berikut: