

Tools to Analyze and Design Discrete Time Linear Filters

1. [One pole at \$a\$ and one zero at zero](#) (set $a=0$) $y[n] = x[n] + by[n-1]$ $H(z) = \frac{z}{z-b}$

2. [One pole at \$b\$ and one zero at \$a\$](#) $y[n] = -by[n-1] + x[n] + ax[n-1]$ $H(z) = \frac{z-a}{z-b}$

3. [Pair of complex poles and two zeros at zero](#)

$$y[n] = x[n] + 2b \cos(\phi)y[n-1] - b^2y[n-2] \quad H(z) = \frac{z^2}{z^2 - 2b \cos(\phi)z + b^2}$$

4. [Pair of complex zeros and two poles at zero](#)

$$y[n] = x[n] + 2a \cos(\theta)x[n-1] - a^2x[n-2] \quad H(z) = \frac{z^2 - 2a \cos(\theta)z + a^2}{z^2}$$

5. [Pair of complex zeros and pair of complex poles](#)

$$y[n] = x[n] + 2a \cos(\theta)x[n-1] - a^2x[n-2] + 2b \cos(\phi)y[n-1] - b^2y[n-2] \quad H(z) = \frac{z^2 - 2a \cos(\theta)z + a^2}{z^2 - 2b \cos(\phi)z + b^2}$$

6. [Design of First and Second Order Digital Filters](#) (Combines tools in 1-5 above)

7. [MATLAB filter design tool equivalent to the LabVIEW module 8.1 can be found at 8.1: Discrete-Time Frequency Response from Poles and Zeros](#). This tool supports 3 poles and 3 zeros. (Also see code provided with the on-line version of the text book at: [MATLAB versions of some LabVIEW modules](#).) In this tool the angle of the poles and zeros is normalized by pi. Example for zeros at $\pm\pi/4$ (45 degrees) inter 0.25.

8. Mathematica tool for [Transfer Function Analysis by Manipulation of Poles and Zeros](#). This tool supports 4 poles and 4 zeros.

9. [MATLAB live script LS101PoleZero_z.mlx: Digital Filter Performance by specifying the gain constant and up to six poles and/or zeros via radius \$r\$ and angle](#) From: Figure 6.13.3: MATLAB live script for discrete-time frequency response from poles and zeros.

$$\mathbf{H}(e^{j\Omega}) = \frac{C(e^{j\Omega} - \mathbf{z}_1)(e^{j\Omega} - \mathbf{z}_2) \cdots (e^{j\Omega} - \mathbf{z}_M)}{(e^{j\Omega} - \mathbf{p}_1)(e^{j\Omega} - \mathbf{p}_2) \cdots (e^{j\Omega} - \mathbf{p}_N)}$$