Z Transform Basics Design and analysis of control systems are usually performed in the frequency domain; where the time domain process of convolution is replaced by a simple process of multiplication of complex polynomials in the frequency domain. Sampled data systems use a similar concept using a unit delay as the basic building block. The analog s-plane maps into the sampled data z-plane by substitution of variables where $z=e^{sT}$ or more importantly by

$$z^{-1}=e^{-sT}$$

The later representation is seen to be identical to a delay line, with z^{-n} representing a delay of nT seconds. Transfer functions, including impedance and admittance functions are described as polynomial ratios of the form G=N/D, where $N=a_0 + a_1z^{-1} + \ldots a_nz^{-n}$ and $D = 1 + b_1z^{-1} + \ldots + b_nz^{-n}$ are the numerator and denominator polynomials respectively. Notice that $b_0 = 1$. Then rearranging the following equation with D' = D-1

$$Vo/Vi = N(1+D1)$$
$$Vo(1+D') = ViN$$
$$Vo = ViN-VoD'$$

which is the "Direct" programming method that is more rigorously derived in [2] pp 284,285. This equation can also be implemented in the s-plane using the following block diagram. This implementation allows for SPICE analysis of the time domain difference equations, including both transient and ac analysis.

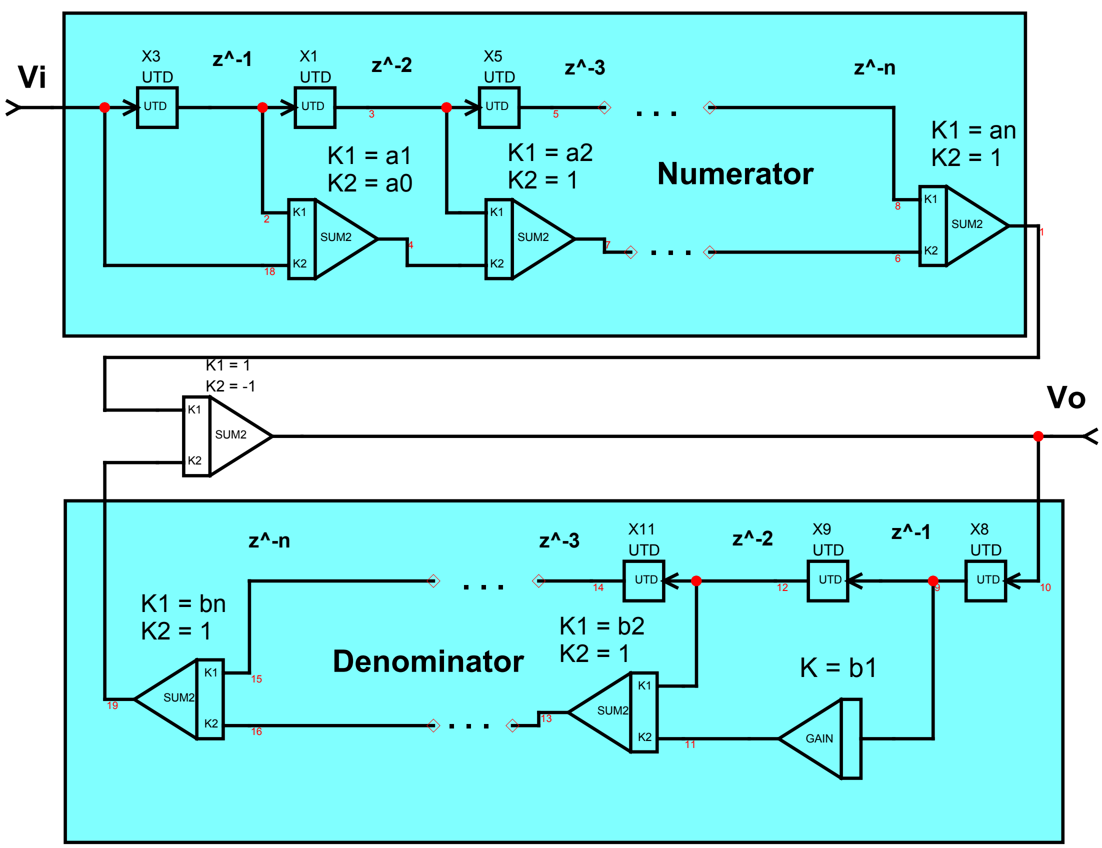


Figure z1, Direct Programming Method

Bilinear Transform Solving for s as a function of z yields

$$s=(1/T)\ln(z)$$

The ln(z) function can be broken down into 2 common approximations. Lets first do this by using the first term of the series expansion where $\ln(z) = 2(z-1)/(z+1)$. Then let $z+1 = 2z$ to further simplify to $\ln(x)=(z-1)/z$. So that

$$s=(1/T)(z-1)/z$$

$$s=(2/T)(z-1)/(z+1)$$

The first representation is the one commonly used [2] pg 60 in the z-transform tables. Mathematically it is common to let T = 1 and omit it from the tables, leaving it to the user to scale the result for other sample frequencies. This scaling is quite valuable for evaluating high order polynomials where preventing numerical overflow is important; but the work presented here will never go beyond first order polynomials so the value for T will be retained. Restating the above equations to represent integration and delays yields:

$1/s=T(1/(1- z^{-1}))$	Rectangular integration (z Transform)
$1/s=T/2(1+ z^{-1})/(1- z^{-1})$	Trapezoidal integration (Bilinear transform)

There are 2 interpretations to these equations in terms of integration method, although they were derived here from a series expansion; they could have also been derived in time domain using rectangular and trapezoidal integration methods. Figure z2 shows the results for a continuous time integration of current through a 1 uHy inductor vs. the z transform method. The z transform uses a 100kHz sample rate.

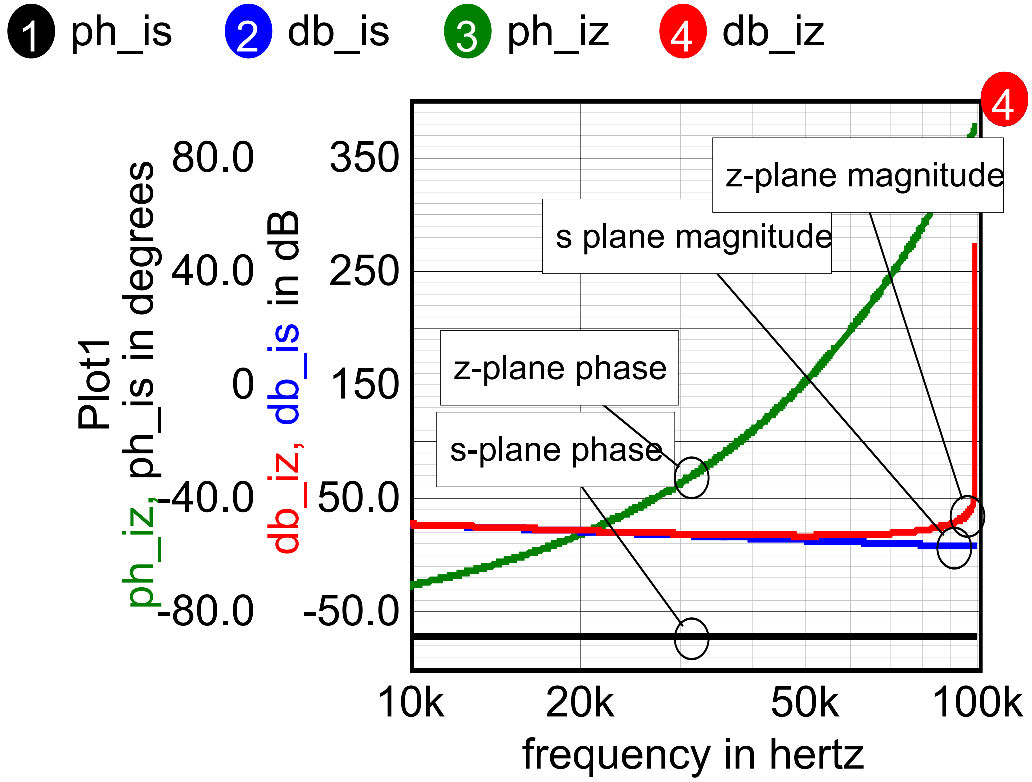


Figure z2, Z Transform of an integrator compared with continuous time integration

Figure z3 shows the bilinear transform method in which the phase lags 90 degrees up to ½ the sample frequency. Its magnitude function goes to zero at ½ the sample frequency.

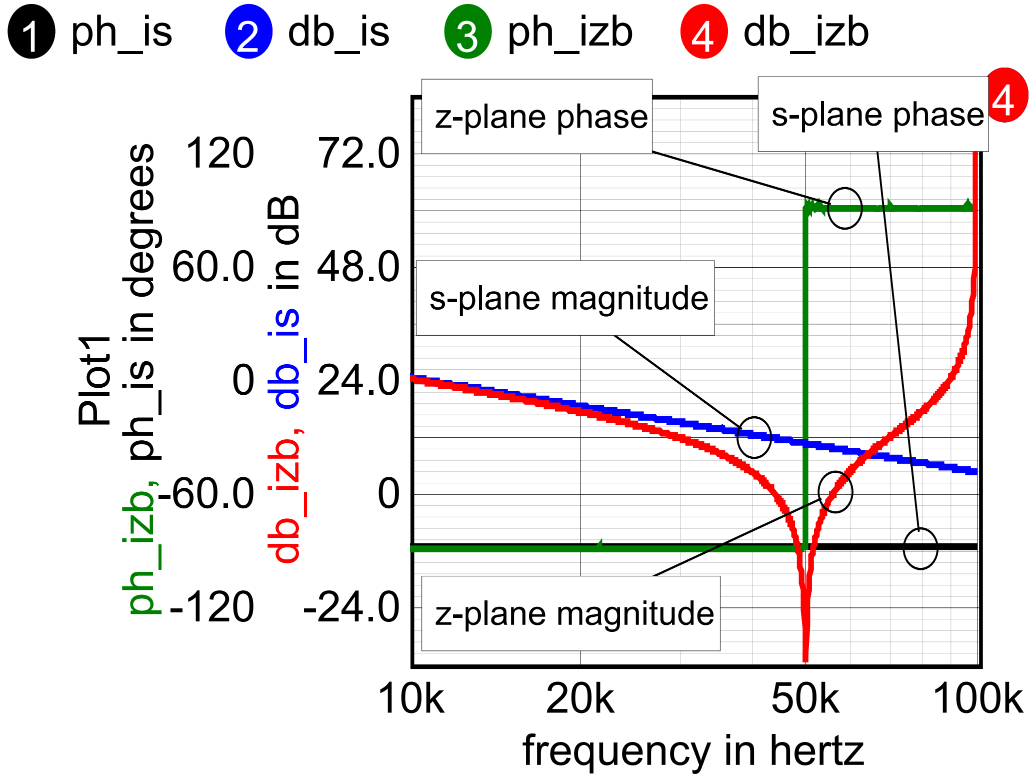


Figure z3, Bilinear transform of integrator compared with continuous time integration

These figures illustrate why most designers favor the bilinear transform for low pass filters. The filter attenuation actually increases when compared with the same linear design and out of band signals near ½ the sampling frequency are attenuated. That makes the anti-aliasing filter easier to design. For control systems, the gain margin increases, in some cases improving response time.

As frequency increases past ½ the sampling frequency, aliasing causes the results repeat as shown in figure z4.

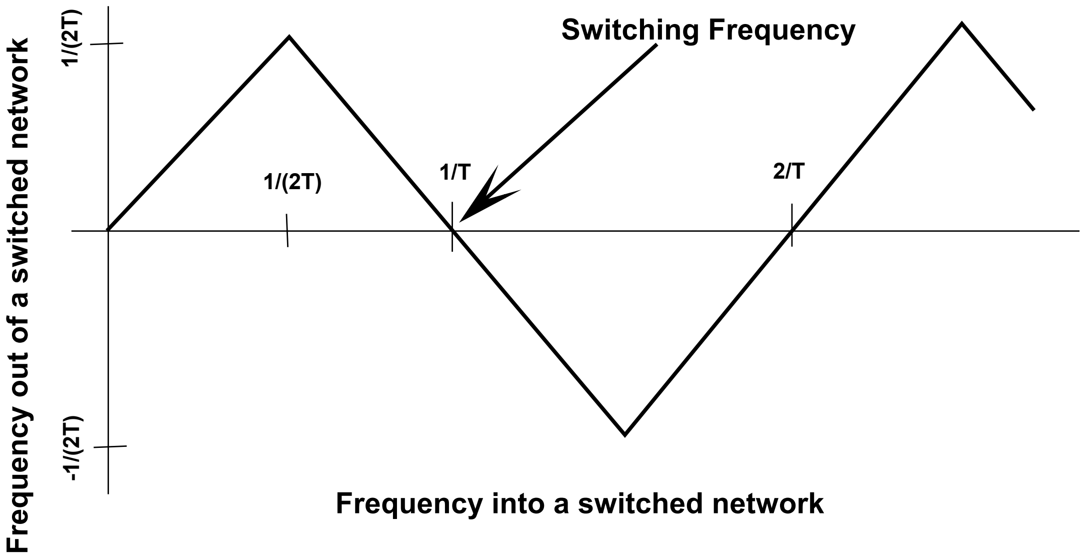


Figure z4, Time domain frequency out vs. frequency in for sample data systems.

While the information bandwidth doesn't exceed ½ the switching frequency, there is indeed information contained above the sampling frequency. Z-transforms can be used to described heterodyned signal detection by placing an analog bandpass filter about the center frequency of interest followed by a digital lowpass filter. Moreover, the samples can be separated by 90 deg (in time), with the in phase component representing real numbers and the delayed sample data being imaginary numbers. A Fourier transform converts the complex time data to the frequency domain where it can be filtered. Then an inverse fourier transform recovers the filtered time dependant data. If certain rules are followed, there will be no imaginary data in the time domain.

z-plane frequency warping As shown previously in Figures z2 and z3, s-plane poles and zeros ranging to infinity are warped into the z plane. Mathematically, the warping is described by evaluating the s-plane frequency for s = jw and the z-plane frequency = jwz.

jw=(2/T)*(e^(jwz*T-1)/(e^(jwz*T-1)

Multiplying numerator and denominator by 1/2*e^(-jwz*T/2) gives

w=2/T*sin(wz*T/2)/cos(wz*T/2)

w= c *tan(wz*T/2) or wz=2/T*atan(w/c), c=2/T

Now, the z-plane argument is phase going from –pi/2 to pi/2 as s-plane frequency goes from – infinity to + infinity.

Figure z5 illustrates this warping. Importantly the warping maps each s-plane frequency to a unique z-plane frequency. Filters such as Chebyshev, Butterworth and Elliptical can be mapped into the z-plane such that filter cutoff frequencies are the same by adjusting c. The filter will then have a somewhat sharper cutoff than its corresponding s-plane filter because frequencies approaching infinity are compressed to ws/2.

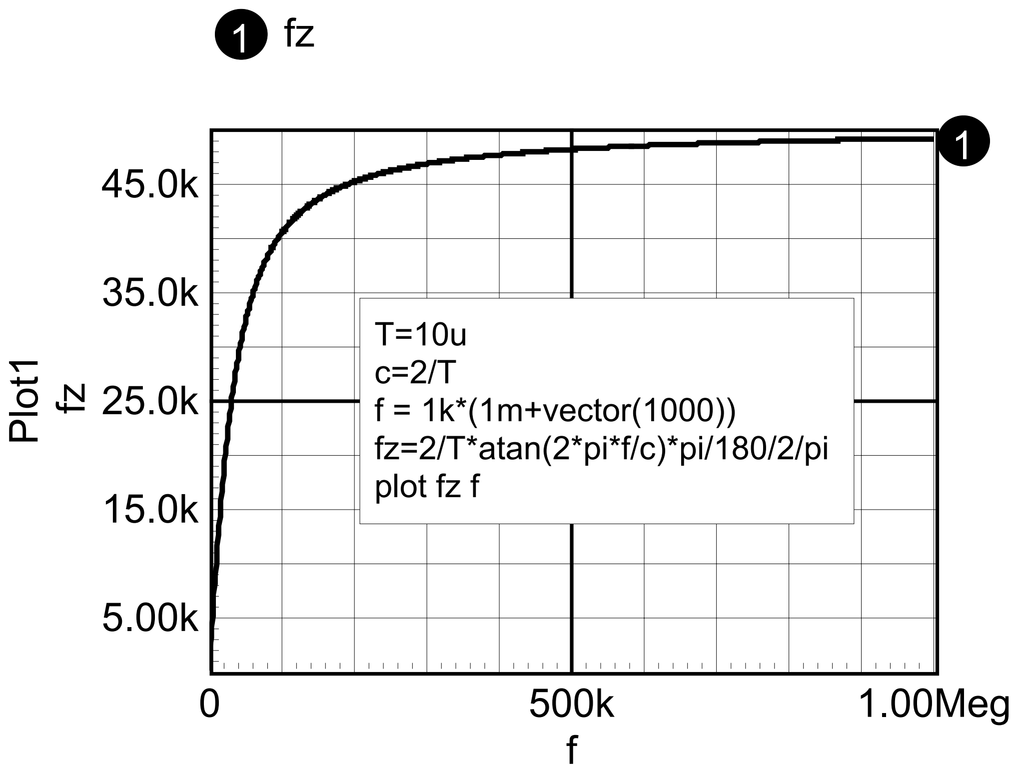
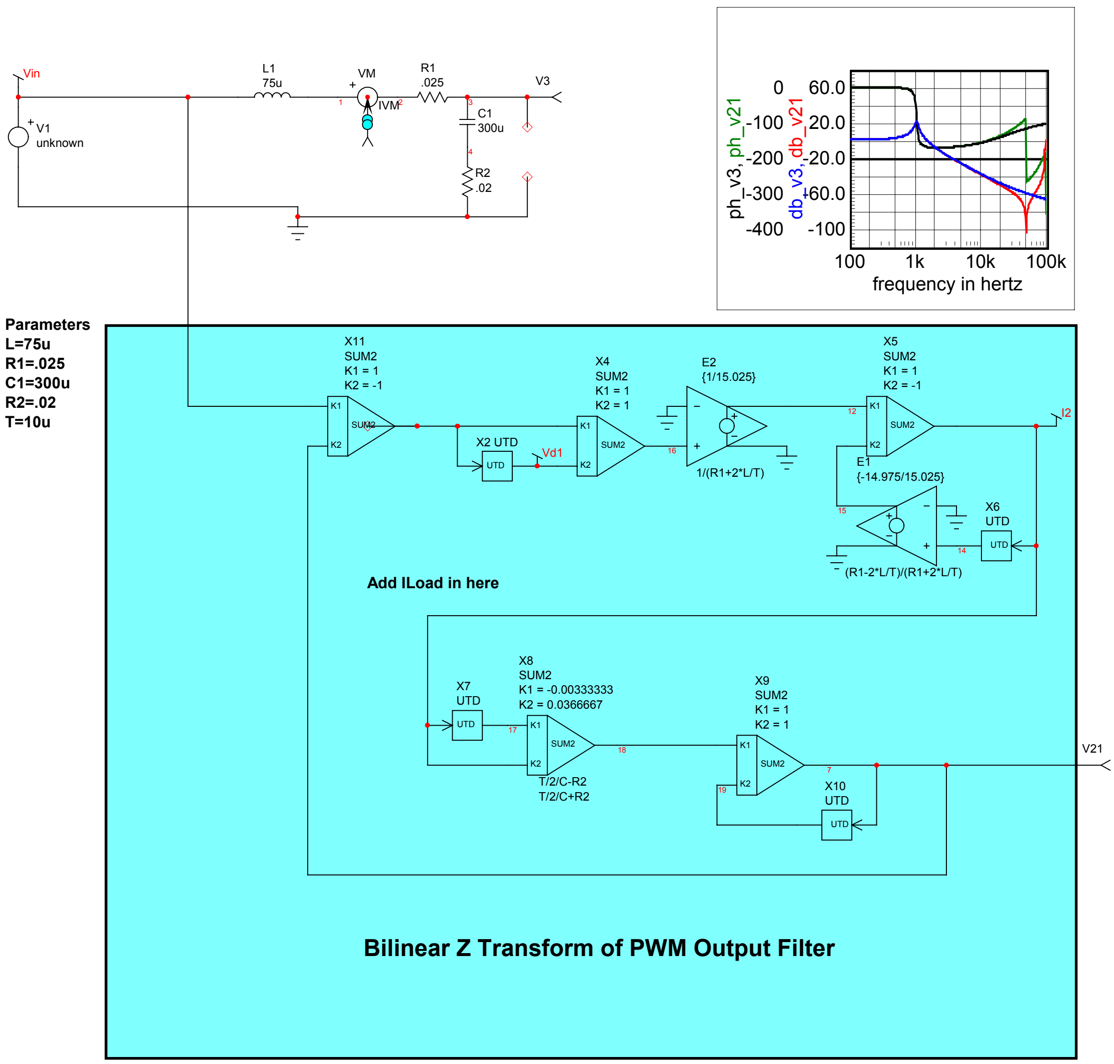


Figure z5, Bilinear transform maps s-plane frequency, f to sampled frequency fz for c=2/T.

The script shown in figure z5 was used to plot the graph in IntuScope. Notice that angles are in degrees, the pi/180 correct this and frequency is converted from 1/sec to Hertz by scaling w = 2*pi*f.

To recap; when transforming from continuous to sampled systems, the poles or zeros at infinity move to the nyquist frequency (1/2 the sampling frequency) in the z-plane. For low-pass filters, there are zeros at infinity so that the signals near the nyquist frequency go to zero. The frequency warping between z-plane and s-plane is approximately linear for low frequencies; but s-plane frequencies get compressed near the nyquist frequency and show different behavior depending on the approximation used for ln(z).The constant c can be adjusted to make fz = f at a single frequency. Analog filters are needed to select the appropriate frequency range and are usually low pass, rejecting signals > 1/(2*T)



Vin
V1
unknown
L1
75u
VM
IVM
R1
.025
C1
300u
R2
.02
V3
0
-100
-200
-300
-400
60.0
20.0
-20.0
-60.0
-100
ph_v3, ph_v21
db_v3, db_v21
100
1k
10k
100k
frequency in hertz
Parameters
L=75u
R1=.025
C1=300u
R2=.02
T=10u
X11
SUM2
K1 = 1
K2 = -1
X2 UTD
Vd1
X4
SUM2
K1 = 1
K2 = 1
E2
{1/15.025}
1/(R1+2*L/T)
X5
SUM2
K1 = 1
K2 = -1
I2
E1
{-14.975/15.025}
X6
UTD
(R1-2*L/T)/(R1+2*L/T)
Add ILoad in here
X7
UTD
X8
SUM2
K1 = -0.00333333
K2 = 0.0366667
T/2/C-R2
T/2/C+R2
X9
SUM2
K1 = 1
K2 = 1
X10
UTD
V21
Bilinear Z Transform of PWM Output Filter