Converting Feedback Systems from Analog to Digital Control©

By

Raymond L. Barrett, Jr., PhD, PE
CEO, American Research and Development, LLC

1.0 Introduction -

The availability of inexpensive microprocessor-based embedded controllers has allowed the use of digital control techniques in modern systems. Digital controllers do not exhibit the drift and temperature dependence of their analog counterparts.

Despite certain advantages, there are still very high speed applications that cannot yet employ digital techniques. However, because the digital processing speed keeps improving, more applications become amenable to digital control.

For systems that utilize analog control and are to be converted to digital control, the insertion of an Analog-to-Digital Converter (***A2D***) in the feedback loop can cause loop stability issues if the properties of the signal representation and conversion are not considered.

2.0 Signal/System Classification-

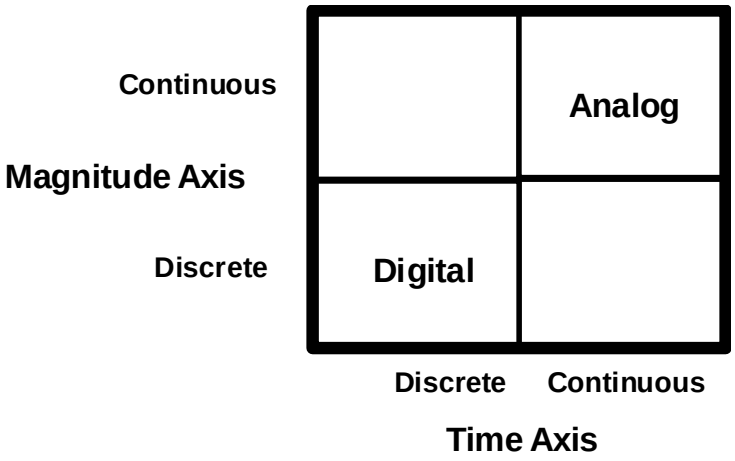

Figure 2.0 - Signal/System Classification Matrix

Analog signals exist at all instances of time and can take on a continuous range of magnitude values over some range of interest. Digital signals, however, have a finite set of discrete values that are used to describe the magnitudes and can only change those values at discrete time intervals. Analog systems that are converted to digital utilize the Analog-to-Digital Converter (***A2D***) with a clocking scheme to provide a digital approximation to the analog signal magnitude as it exists at a sampling instant defined by the sampling clock.

Relative to any signal that may exist as an analog value, the Analog-to-Digital Converter (***A2D***) can introduce errors in the conversion process in both the magnitude representation, as well as the timing of the resulting conversion. For control purposes, the magnitude errors are generally less of a problem than the timing errors.

Another form of control that utilizes a combinations of continuous time with discrete magnitudes such as Pulse-Width-Modulation (***PWM***) control is employed in some systems and faces similar loop stability issues that are more complex than well-defined digital control. Likewise a control scheme with discrete times but continuous magnitudes is possible but usually involves inconvenient implementation issues and is seldom used. Digital control is becoming dominant for those applications that are suitable.

3.0 Analog Signal Representation using Sinusoids-

Classical Analog systems represent the signals as sinusoids for periodic steady-state analysis. The choice of representation of signals is associated with the use of Fourier and Laplace Transform techniques for System descriptions.

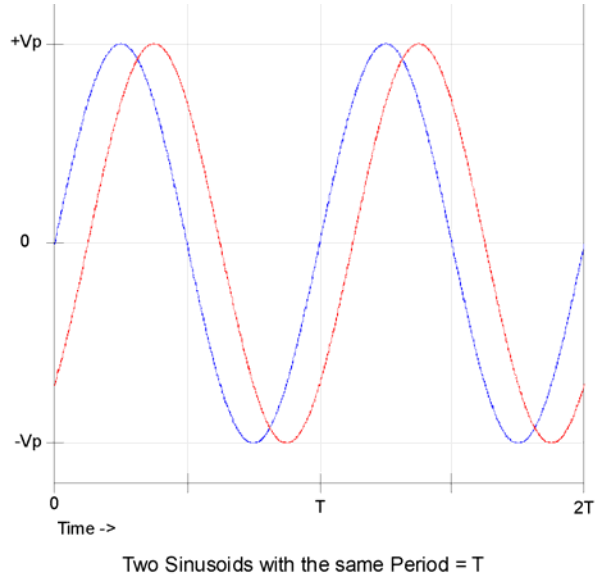

Figure 3.0 – Two Sinusoid with the same Period and a Phase Difference

In Figure 3.0, each sinusoid necessarily is at the same frequency (***F***) expressed in Hertz units and related to the Period (***T***) by ***F*** = ***1/T***.

In the illustration, each sinusoid is essentially identical except for the difference in the time when each waveform rises through a value of zero magnitude. That difference may be expressed as a Delay or time difference ΔT, or more commonly as a proportional phase (ϕ) difference in radians given by $\phi_{Radians} = 2\pi(\Delta T/T)$. The phase difference in radians can be converted to degrees by $\phi_{Degrees} = (180/\pi) \; x \; \phi_{Radians}$

Choosing one sinusoid as a reference for timing purposes allows the second sinusoid to be expressed as:

$$V_{Sine} = V_p \sin\left(2\pi Ft + \phi_{Radians}\right) \qquad [3.0]$$

The peak level is used to denote the magnitude of the waveform and is the same for both sinusoids shown in the illustration, but may be different distinct values for each sinusoid in other cases. The sinusoid that is chosen as the "$T = 0$" timing reference may be expressed as:

$$V_{Sine-0} = V_{p-Sine-0} \sin\left(2\pi F_{Sine-0} t\right) \qquad [3.1]$$

Note the distinction that the value for $\phi_{Radians}$ is zero for the (0) reference signal. Similarly, the value of V_{Sine-0} is often chosen to be some convenient value such as one unit ($V_{Sine-0} = 1$) so that the effects of signal processing on magnitude and phase shifts is contained in the value of V_p and $\phi_{Radians}$ of the sinusoid response.

4.0 Analog System Response-

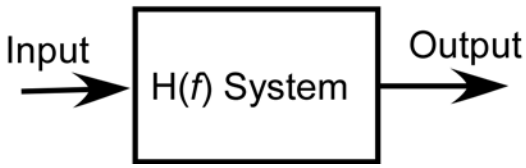

Figure 4.0 – A System Diagram with Input and Output Denoted

For the System in Figure 4.0 to be amenable to Classical Analog Control, that system must be causal and Linear, Time-Invariant (***LTI***) to a close approximation. Physical systems are causal in that the inputs always occur before the expected output results occur (i.e., there are no "predictors" in the system). Linearity is assured if the output is proportional to the input and a zero input implies a zero output (at least incrementally). Time-Invariance is assured if the

relationship between the output and input changes slowly enough to be inconsequential, such as is caused by drift and aging effects.

Such a System is described by a "transfer function" ***H(f)*** that expresses the Output response as a function of the Input stimulus. Other necessary inputs, such as a source of power, are often ignored in simple models.

Several means are commonly used to describe the transfer function, including the "Impulse or Step Response" to describe a response to an impulse or step input respectively. It is not common to design an analog system directly from such a time-domain response, but rather to design with steady-state sinusoidal excitation. To determine the analog system behavior one tool commonly used in the frequency-domain is the Bode plot: it shows the magnitude and phase of the output response with a sinusoidal excitation with a unit magnitude and a zero phase.

5.0 An Analog System Response-

In the following Bode plot, the horizontal frequency axis has the units of Hertz or cycles-per-second, but could have easily been radians per second as well. The presentation utilizes a logarithmic scaling both for the horizontal frequency axis as well as the vertical response magnitude axis. The vertical axis for the Phase response, however, remains linear in presentation. The two plots are placed in relation to each other so that they can be thought to have a common horizontal axis and each magnitude and phase are related by the corresponding frequency location.

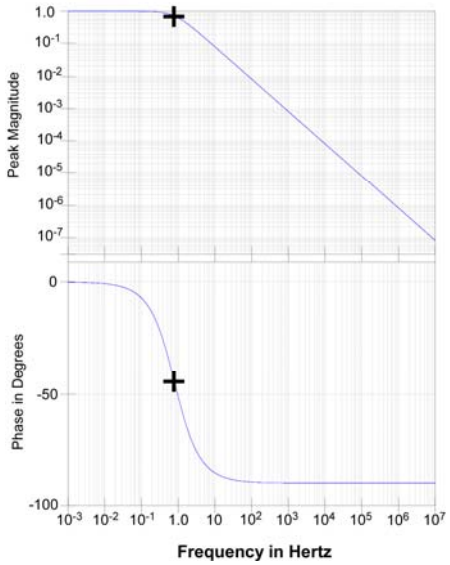

Figure 5.0 – Bode Plot of *H(f)* System Response

A system is chosen for discussion that is represented by the Bode plot above in Figure 5.0, and has a salient point marked by the bold "+" symbols. For a steady-state sinusoidal stimulus, the system shows a response pair of magnitude and phase effects plotted versus the excitation frequency. The salient point marked by the bold "+" symbols indicates a "pole" in the response. Behaviors are significantly different at frequencies above and below the pole frequency.

The frequency axis is plotted logarithmically (as is the Peak Magnitude) because the relationships appear as line segments asymptotically above and below the pole frequency. This system transfer function ***H(f)*** can be represented by the equation:

$$H(f) = \frac{1}{1 + j\left(\frac{f}{f_{pole-H}}\right)} = \frac{1}{1 + jF_H} \qquad [5.0]$$

In equation 5.0 the behavior is described as a complex number with the frequency dependence ratio of $\boldsymbol{F_H = f/f_{pole\text{-}H}}$ providing the imaginary part. From the single complex number that provides the transfer function at each frequency, we derive the magnitude and phase as:

$$|H(f)| = \frac{1}{\sqrt{1 + F_H^2}} \qquad [5.1]$$

$$\theta_H(f) = -ATAN(F_H) \qquad [5.2]$$

We evaluate transfer function asymptote responses as the frequency approaches zero ($\boldsymbol{F_H} << 1$), infinity ($\boldsymbol{F_H} >> 1$), and also at the pole frequency itself ($\boldsymbol{F_H} = 1$).

For the case of the stimulus frequency significantly less than the pole frequency:

$$|H(f)|_{F_H \to 0} = \frac{1}{\sqrt{1 + (0)^2}} = 1 \qquad [5.3]$$

$$\theta_H(f)|_{F_H \to 0} = -ATAN(0) = 0 \qquad [5.4]$$

The transfer function asymptotes agree with the Bode plot of figure 5.0 as the frequency approaches zero, or is significantly less that the pole frequency of 1Hertz, as marked by the bold "+" symbols on the Bode plot.

For the case of the stimulus frequency significantly greater than the pole frequency:

$$\left|H(f)\right|_{F_H \to \infty} = \frac{1}{\sqrt{1+(\infty)^2}} = 0 \qquad [5.5]$$

$$\theta_H(f)\big|_{F_H \to \infty} = -ATAN(\infty) = -90^o \qquad [5.6]$$

The transfer function asymptotes agree with the Bode plot of figure 5.0 as the frequency approaches infinity, or is significantly greater that the pole frequency of 1Hertz, as marked by the bold "+" symbols on the Bode plot.

For the case of the stimulus frequency equal to the pole frequency:

$$\left|H(f)\right|_{F_H=1} = \frac{1}{\sqrt{1+(F_H)^2}} = \frac{1}{\sqrt{1+(1)^2}} = \frac{1}{\sqrt{2}} \approx .707 \qquad [5.7]$$

$$\theta_H(f)\big|_{F_H=1} = -ATAN(F_H) = -ATAN(1) = -45^o \qquad [5.8]$$

Because the magnitude response depends on the ratio of $\boldsymbol{F_H = f/f_{pole\text{-}H}}$ and for $\boldsymbol{F_H} >> 1$, we can make the approximation:

$$\left|H(f)\right|_{F_H >> 1} \approx \frac{1}{\sqrt{(F_H)^2}} = \frac{1}{F_H} \qquad [5.9]$$

The magnitude is inversely proportional to the frequency ratio $\boldsymbol{F_H = f/f_{pole\text{-}H}}$ and for every time the frequency increases by a decade (factor of 10), the magnitude decreases by a decade. On the Bode plot with logarithmic scales, this proportion is shown as the asymptotic straight line above the pole frequency, with a decrease of a decade in magnitude per decade of frequency increase, hence the reason for using the logarithmic axes for the Bode plot magnitude.

The phase response is somewhat different, with symmetrical results around the pole frequency. A series expansion of the ***ATAN*** function indicates that it is proportional to the ratio $\boldsymbol{F_H = f/f_{pole\text{-}H}}$ near the pole frequency value, so we simply evaluate the ***ATAN*** at frequencies a decade above and a decade below the pole frequency with the expectation that the errors will be about 10%:

$$\theta_H(f)\big|_{F_H \to 10} = -ATAN(F_H) = -ATAN(10) = -5.7^o \qquad [5.10]$$

$$\theta_H(f)\big|_{F_H \to 1/10} = -ATAN(F_H) = -ATAN(1/10) = -84.3^o \qquad [5.11]$$

The numerical result shows that the ATAN function has nearly reached its asymptotic limits of 0^o and -90^o within the first decade above and below the pole frequency, with a nearly constant slope of -45^o per decade around the pole frequency.

6.0 Analog Feedback Control of the System Response-

The pole frequency of the system discussed in Section 5.0 appeared at 1 Hertz, but we show that we can use an analog feedback control system to induce that system to produce a behavior with a pole frequency much higher and therefore provide faster response. Such might be the case if the system discussed in Section 5.0 was a mechanical system and faster response were required.

We modify the system by the addition of two new blocks as follows:

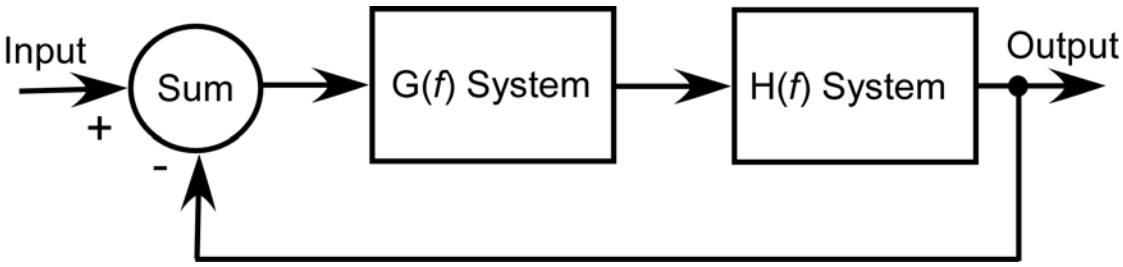

Figure 6.0 – Feedback System Diagram with Input and Output Denoted

We have added the "***G(f)***" block and the "Sum" block with a feedback of the output signal into the "Sum" block "-" input.

We write:

$$[Output] = H(f) \bullet G(f)[Input - Output]$$

$$[Output] = H(f) \bullet G(f)[Input] - H(f) \bullet G(f)[Output]$$

$$[Output] + H(f) \bullet G(f)[Output] = H(f) \bullet G(f)[Input]$$

$$T(f) = \frac{[Output]}{[Input]} = \frac{H(f) \bullet G(f)}{1 + H(f) \bullet G(f)} \qquad [6.0]$$

Addition of the ***G(f)*** block with feedback has allowed the freedom to make the resultant transfer function ***T(f)*** nearly unity for all cases with the ***H(f)*G(f)*** >>1. We choose a typical "Operational Amplifier" as a candidate for the ***G(f)*** requirement and show its Bode plot as Figure 6.1 below.

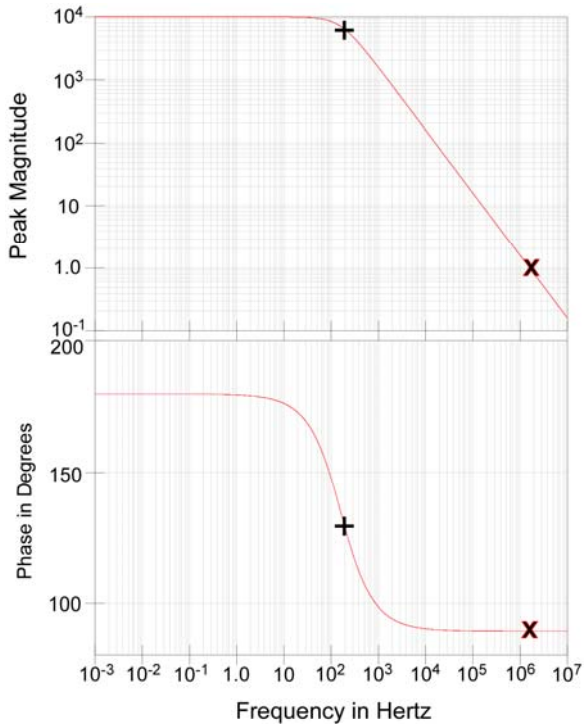

Figure 6.1 – Bode Plot of *G(f)* of the Operational Amplifier Response

The Operational Amplifier has its own pole frequency $f_{pole\text{-}G}$, near 200 Hertz and considerable "Gain" of ~10,000 (10^4) for low frequencies. It is chosen for this high value so that ***G(f)*H(f)*** >> 1, at least for lower frequencies. The Operational Amplifier does not reach unity gain until the frequency approaches ~2,000,000 Hertz (2MHz), as denoted by the bold "**x**" marks on the Bode plot. The algebraic relationship, for $F_G = f/f_{pole\text{-}G}$ that defines the "Operational amplifier Bode plot is given by:

$$G(f) = 10^4 \bullet \frac{1}{1 + j(F_G)} \qquad [6.1]$$

The Operational Amplifier has two differential input terminals and also serves also as the "Sum" block in the feedback system diagram. The negative terminal is used as the feedback input so that the ***G(f)*H(f)*** product can be evaluated with the difference ("-" sign), already in the feedback loop. The magnitude and phase of ***G(f)*** are respectively given by:

$$|G(f)| = \frac{10^4}{\sqrt{1 + (F_G)^2}} \qquad [6.2]$$

$$\theta_{(G)}(f) = -ATAN(F_G) \qquad [6.3]$$

7.0 The Open Loop System -

Each block in the path around the feedback loop contributes to the response as a product of the transfer functions ***G(f)*H(f)***, and the product is known as the "Open Loop" gain:

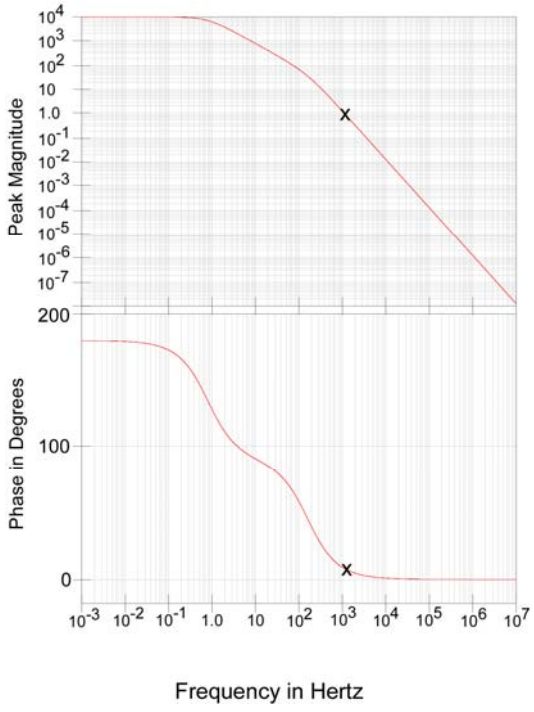

Figure 7.0 – Bode Plot of the "Open Loop" or *G(f)*H(f)* Response

The "Open Loop" combination has its own combined characteristic as shown in Bode plot of Figure 7.0 above, and given by equation [7.0], below:

$$G(f) \bullet H(f) = \frac{10^4}{[1 + j(F_G)] \bullet [1 + j(F_H)]} \qquad [7.0]$$

The "Open Loop" produces a composite magnitude from that product:

$$|G(f) \bullet H(f)| = |G(f)| \bullet |H(f)| = \frac{10^4}{\sqrt{1 + (F_G)^2}} \bullet \frac{1}{\sqrt{1 + (F_H)^2}} \qquad [7.1]$$

The asymptotic low frequency magnitude is obtained from the product of magnitudes with the value 10^4 contributed mostly by the $|\boldsymbol{G(f)}|$ "Operational Amplifier, " and only unity from the $|\boldsymbol{H(f)}|$ of the original system. As the frequency increases to exceed the pole frequency $\boldsymbol{f_{pole\text{-}H}}$ = 1 Hertz, the magnitude decreases at the rate of one decade decrease per decade of frequency increase. However, with further frequency increases to exceed the pole frequency $\boldsymbol{f_{pole\text{-}G}}$ = 100 Hertz, the magnitude decreases at the rate of two decades decrease per decade of frequency increase.

Note that the greater rate of decrease of Open Loop magnitude reduces the unity-gain of the combination to near 1000 Hertz as denoted by the bold "**x**" marks on the Bode plot; much less than that of the Operational Amplifier unity-gain frequency.

The "Open Loop" also produces a composite phase from the product of transfer functions that is the sum of angles:

$$\theta_{(G \bullet H)}(f) = \theta_{(G)}(f) + \theta_{(H)}(f) = -ATAN(F_G) - ATAN(F_H) \qquad [7.2]$$

The asymptote low frequency phase is obtained from the sum of phase angles $\boldsymbol{0_G} + \boldsymbol{0_H}$ and is dominated by the 180° $\boldsymbol{0_G}$ phase shift of the Operational Amplifier. As the frequency increases to 1/10 of the pole frequency $\boldsymbol{f_{pole\text{-}H}}$ = 1/10 Hertz, the phase decreases at approximately –45° per decade up to 10 times the pole frequency $\boldsymbol{f_{pole\text{-}H}}$ = 10 Hertz, bringing the phase from 180° to 90° and also the beginning of the effect of 1/10 frequency of pole at $\boldsymbol{f_{pole\text{-}G}}$ = 10 Hertz. The phase continues to decrease at approximately –45° per decade up to 10 times the pole frequency $\boldsymbol{f_{pole\text{-}G}}$ = 100 Hertz, and approaches the asymptotic value of zero.

8.0 The Open Loop System and Closed Loop Stability -

Because the "Closed Loop" response with feedback was shown previously in equation [6.0], and repeated here for convenience, to be:

$$T(f) = \frac{H(f) \bullet G(f)}{1 + H(f) \bullet G(f)} \qquad [6.0]$$

We note that the "Closed Loop" response is the ratio of the "Open Loop" to the augmented (1 + "Open Loop") characteristics, and used the justification for the approach based on the idea that the "Open Loop" characteristic is much greater than unity. However, we have also shown that the "Open Loop" unity-gain frequency is much less than the "Operational Amplifier" and denoted the location on the Bode plot with the "**x**" character. Comparing the "Open Loop" low frequency asymptote with its gain of 10^4 and phase of 180^o to the unity-gain frequency with a magnitude of 1 and a phase nearly zero, we conclude that the feedback has undergone a phase reversal as the frequency increases. What started as a positive sign has become a difference and the denominator of T(f) approaches a magnitude of zero resulting in infinite gain attributed to the second-order pole nature of the open-loop. An uncompensated second-order system is only conditionally stable and results in unfavorable behavior in the closed-loop. We seek to avoid the case of phase reversal at the "Open Loop" unity-gain frequency.

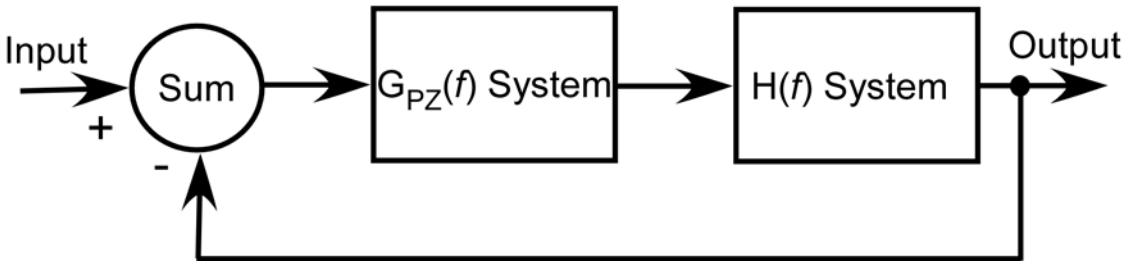

Figure 8.0 – Feedback System Diagram with PZ Compensation Denoted

To avoid the phase reversal of the "Open Loop" characteristic as we approach the unity-gain frequency, we modify the "Operational Amplifier" behavior by introducing a feedback around that stage alone. We intentionally introduce another denominator pole as well as a numerator "zero" into the "Operational Amplifier" transfer function to perform a PZ (pole-zero) compensation:

$$PZ(f) = \frac{1 + j(F_Z)}{1 + j(F_P)} \qquad [8.0]$$

As shown in Figure 8.0 below, we place the pole $f_{pole\text{-}P}$ at 0.1 Hertz and the denominator zero $f_{zero\text{-}Z}$ at ~2 Hertz. The pole behavior is similar to those encountered before and the higher frequency zero behavior is likewise similar, but the magnitude increases at a one decade per decade frequency increase, cancels the pole effects after a little more than one decade of influence. The phase effect is likewise similar, but with a reversal of sign to provide the composite compensator transfer function Bode plot.

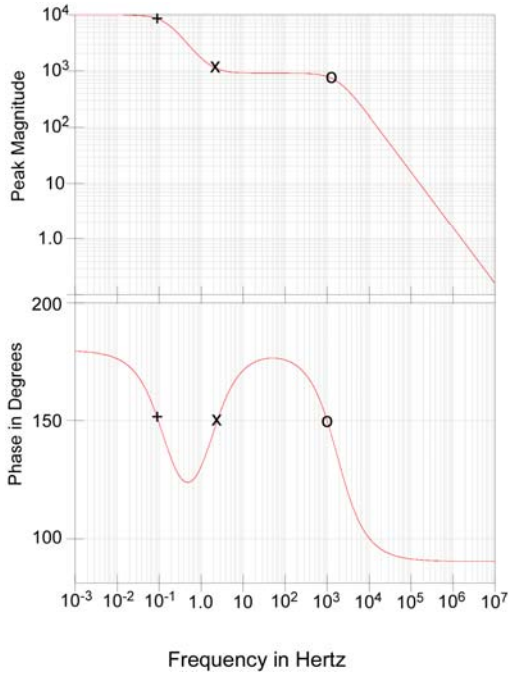

Figure 8.0 – Bode Plot of the Operational Amplifier Response with PZ Compensator

The PZ compensator initiates a magnitude decrease at the pole frequency $f_{pole\text{-}P}$ of 0.1 Hz (as indicated by the "+" symbol in the Figure 8.0 Bode plot), for little more than one decade until the zero frequency $f_{zero\text{-}Z}$ of ~2 Hz (as indicated by the "**x**" symbol in the Figure 8.0 Bode plot), decreasing the effective gain at the rate of one decade decrease per decade of frequency over the one decade (a net effect before the zero of one decade gain decrease), and a return to a resultant constant gain of $\sim 10^3$ as frequency increases.

From the zero frequency $f_{zero\text{-}Z}$ of ~2 Hz the gain remains $\sim 10^3$ as frequency increases until the asymptote intercepts the original behavior of the "Operational Amplifier" at about 2,000 Hertz., as shown in the prior Figure 6.1 of the "Operational Amplifier" itself.

The intercept of asymptotes is illustrated in Figure 8.0 by the "**o**" symbol on the Bode plot.

The phase response of the compensated "Operational Amplifier" is also altered by the PZ structure, with a phase at $f_{pole\text{-}P}$ of 0.1 Hz near 150° and decreasing, but returning to near 150° at $f_{zero\text{-}Z}$ of ~2 Hz and increasing. The 150° phase shift is not encountered again until the frequency rises to near the 2,000 Hertz intercept point.

Contrast the range of phase values introduced by the compensated "Operational Amplifier" of Figure 8.0 with the uncompensated "Operational Amplifier" of Figure 6.1 over the range of ~2 Hertz to 2,000 Hertz.

Application of the PZ compensation to the "Operational Amplifier" has modified the transfer function $G_{PZ}(f)$ to become:

$$G_{PZ}(f) = 10^4 \bullet \frac{1+j(F_Z)}{1+j(F_P)} \bullet \frac{1}{1+j(F_G)} \qquad [8.1]$$

The magnitude of $G_{PZ}(f)$ is given by:

$$|G_{PZ}(f)| = 10^4 \bullet \frac{\sqrt{1+(F_Z)^2}}{\sqrt{1+(F_P)^2}} \frac{1}{\sqrt{1+(F_G)^2}} \qquad [8.2]$$

The phase of $G_{PZ}(f)$ is given by:

$$\theta_{(G-PZ)}(f) = ATAN(F_Z) - ATAN(F_P) - ATAN(F_G) \qquad [8.3]$$

9.0 The PZ Compensated Open Loop System-

Application of the PZ compensation to the "Operational Amplifier" has modified the "Open Loop" transfer function $G_{PZ}(f)*H(f)$ to become:

$$G_{PZ}(f) \bullet H(f) = 10^4 \bullet \frac{1+j(F_Z)}{1+j(F_P)} \bullet \frac{1}{1+j(F_G)} \bullet \frac{1}{1+j(F_H)} \qquad [9.1]$$

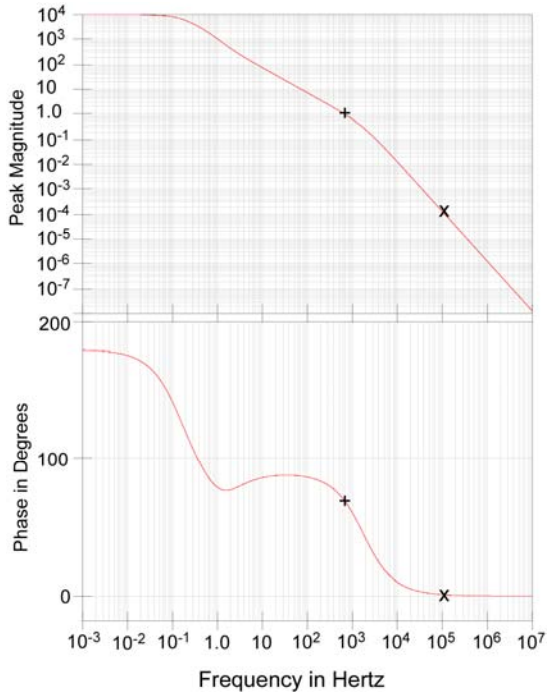

Figure 9.0 – Bode Plot of the PZ Compensated "Open Loop" or $G_{PZ}(f)*H(f)$ Response

The magnitude of $G_{PZ}(f)*H(f)$ is given by:

$$\left|G_{PZ}(f)H(f)\right| = 10^4 \bullet \frac{\sqrt{1+(F_Z)^2}}{\sqrt{1+(F_P)^2}} \bullet \frac{1}{\sqrt{1+(F_G)^2}} \bullet \frac{1}{\sqrt{1+(F_H)^2}} \qquad [9.2]$$

The phase of $G_{PZ}(f)*H(f)$ is given by:

$$\theta_{(G-PZ\bullet H)}(f) = ATAN(F_Z) - ATAN(F_P) - ATAN(F_G) - ATAN(F_H) \qquad [9.3]$$

From the Bode plot, we note that the phase associated with the unity-gain frequency (denoted by the "+" character) is ~75°, and indicates that the "Closed Loop" system will be stable with this "Phase Margin" of ~75°.

10.0 The PZ Compensated Closed Loop System-

The PZ Compensated "Open Loop" system has sufficient phase margin to connect in a "Closed Loop" without oscillation and the Bode plot for the "Closed Loop" ***T(f)*** is shown in Figure 10.0 as follows.

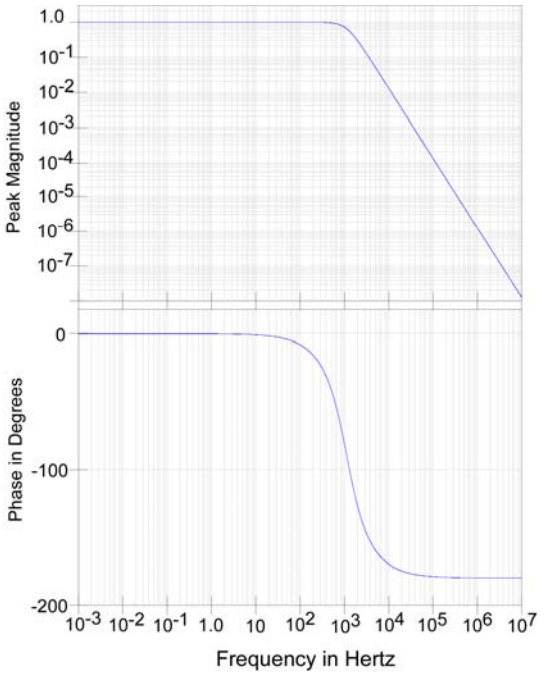

Figure 10.0 – Bode Plot of the PZ Compensated "Closed Loop" *T(f)* Response

We note that "Closed Loop" behavior ***T(f)*** produces a similar response to the original ***H(f)*** both in asymptotic low-frequency magnitude and phase. As the frequency increases, however, there is no substantial magnitude decrease until the frequency approaches the "Open Loop" $G_{PZ}(f)*H(f)$ unity-gain frequency at ~1000 Hertz. Above that frequency, the magnitude drops by two decades for every decade of frequency increase indicating that two poles are involved in setting the asymptote.

The phase change asymptote is ~90° per decade and also indicates that the "Closed Loop" has a double pole. The high-frequency asymptote is 180° further indicating that two poles are involved in setting that asymptote. We know:

$$T(f) = \frac{H(f) \bullet G_{PZ}(f)}{1 + H(f) \bullet G_{PZ}(f)} \qquad [6.0]$$

We expand:

$$T(f)=\frac{10^4 \bullet \frac{1+j(F_Z)}{1+j(F_P)} \bullet \frac{1}{1+j(F_G)} \bullet \frac{1}{1+j(F_H)}}{1+10^4 \bullet \frac{1+j(F_Z)}{1+j(F_P)} \bullet \frac{1}{1+j(F_G)} \bullet \frac{1}{1+j(F_H)}} \qquad [10.0]$$

$$T(f)=\frac{\frac{[1+j(F_Z)]}{[1+j(F_P)][1+j(F_G)][1+j(F_H)]}}{10^{-4}+\frac{[1+j(F_Z)]}{[1+j(F_P)][1+j(F_G)][1+j(F_H)]}} \qquad [10.1]$$

$$T(f)=\frac{[1+j(F_Z)]}{10^{-4}[1+j(F_P)][1+j(F_G)][1+j(F_H)]+[1+j(F_Z)]} \qquad [10.2]$$

We have expanded the expression for the "Closed Loop" ***T(f)*** to show that the denominator has a much more complex function than the "Open Loop" product of $G_{PZ}(f)*H(f)$ and that while it is possible to solve for the roots algebraically, it is a daunting task left better to graphical or numerical methods.

11.0 The Justification for the Detailed Analog System Development-

We have shown that there is much careful consideration of the behaviors of the system components in a compensated Analog Control System. We will now proceed to convert components of the Analog System to a Digital representation. We will assume that the initial Analog System described by ***H(f)*** remains and digital components will be utilized to replace ***G(f)*** and the PZ compensator, as well as any other components.

12.0 Sampling Analog Signals to Begin Digital Representation-

Analog signals are continuous in both time and magnitude, whereas Digital signals are discrete in both time and magnitude. We first convert the Analog signals to a discrete-time representation by sampling and retain the magnitudes as continuous. We then convert the continuous magnitudes to discrete values in a subsequent process.

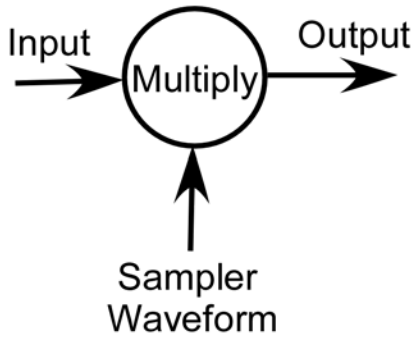

Figure 12.0 – Sampling a Continuous-Time Signal

We perform sampling using a stream of unit magnitude pulses in the time domain in Figure 12.0, as follows:

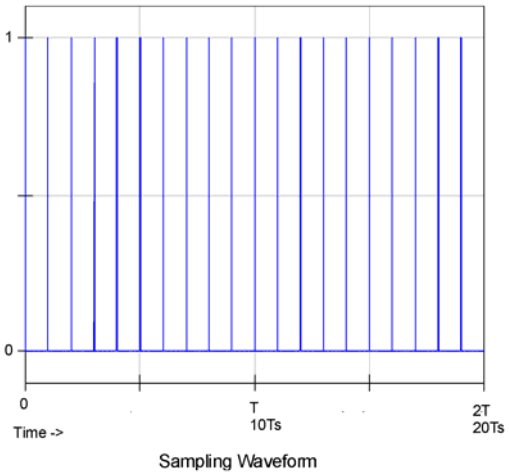

Figure 12.1 – Unit Magnitude Periodic Sampling Waveform

As a "practical" discussion matter, we choose a sampling period $\boldsymbol{T_S}$ to be 10x smaller than the period $\boldsymbol{T}$ of the highest frequency "SineWave" of interest (i.e., the sampling rate is 10x greater than the highest frequency of interest).

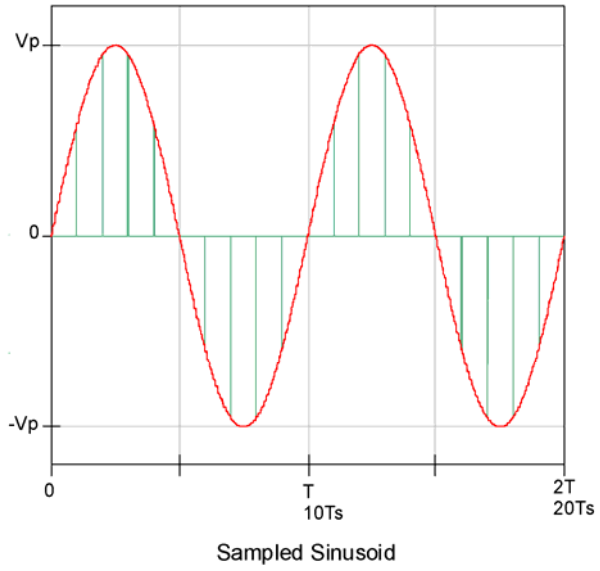

Figure 12.2 – Example "SineWave" and Sampled Output

Each pulse in the sequence is of a finite duration, but sufficiently long enough duration to obtain the sample with accuracy.

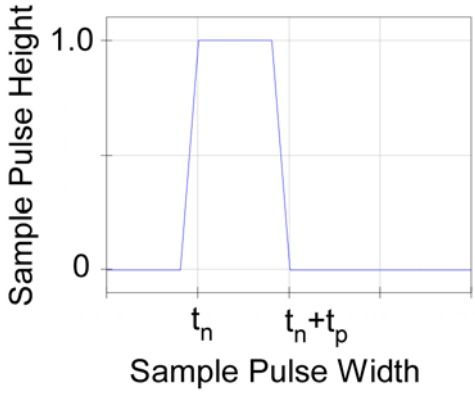

Figure 12.3 – A Single Pulse from the Periodic Sampling Waveform

We show in Figure 12.3 a single representative pulse with width t_p taken from the sequence of Sampler pulses. The pulse should be as short a duration as can be obtained without sacrificing

accuracy. The pulse is shown with a finite rise and fall time to be consistent with a practical sampler device.

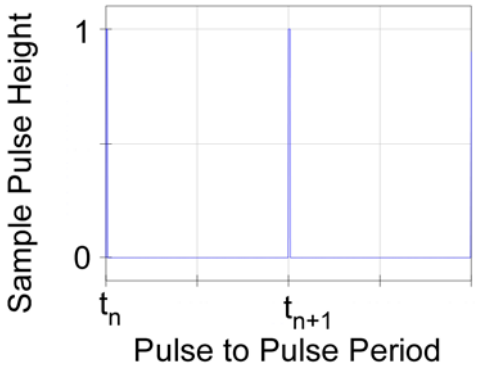

Figure 12.4 – A Pair of Adjacent Pulses from the Periodic Sample Waveform

Two adjacent pulses from the periodic Sampler Waveform are shown in Figure 12.4 to illustrate the uniform spacing such that $\boldsymbol{T_S} = (\boldsymbol{t_{n+1}} - \boldsymbol{t_n})$ for every adjacent interval in the Sampler Waveform.

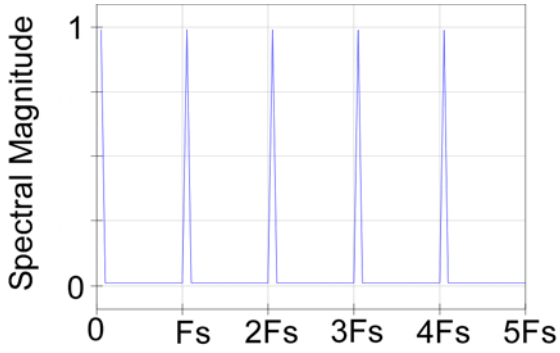

Figure 12.5 – A Pair of Adjacent Spectral Lines from the Periodic Sampler Spectrum

As shown in Figure 12.5, the periodic Sampler Waveform exhibits adjacent spectral lines with a frequency spacing $\boldsymbol{F_S}$ related to the sampler period $\boldsymbol{T_S}$ by the relationship $\boldsymbol{F_S} = 1/\boldsymbol{T_S}$. The initial spectral line at $\boldsymbol{F_S} = 0$ indicates that the Periodic Sampler can represent DC values correctly.

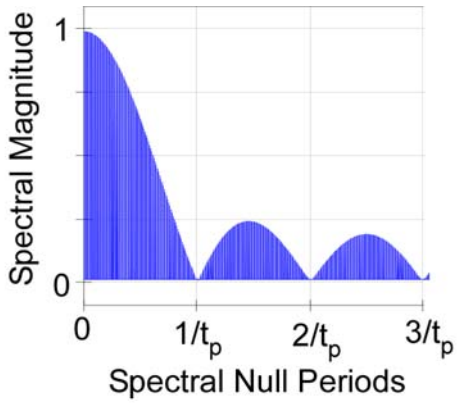

Figure 12.5 – Broad Spectral Lines from the Periodic Sampler Spectrum

As illustrated in Figure 12.5, the periodic Sampler Waveform exhibits spectral null behaviors in the frequency domain with spacing F_{Null} as relates to the sampler pulse width and is defined by $F_{Null} = 1/ t_p$. The spectral line envelope follows a sinx/x form with the peak of unity at DC. In other applications, sampling can be used to modulate information to multiple different spectral lines, but the behavior is more of a nuisance for control applications.

13.0 Sampled Analog "SineWave" Signals-

The frequency spectrum of the "SineWave" input shows a single spectral line in Figure 13.0 at the frequency of $F_{SineWave}$:

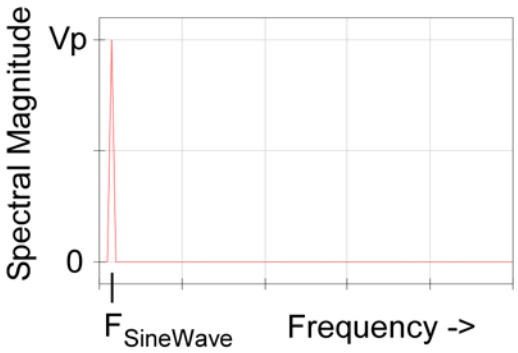

Figure 13.0 – Single Spectral Line from the "SineWave" Source

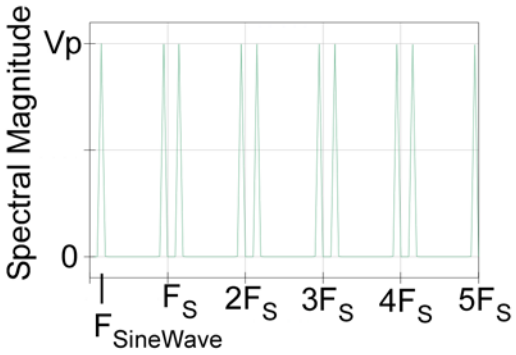

Figure 13.1 – Multiple Spectral Lines from the Sampled "SineWave" Source

The sampled "SineWave" signal consisting of the pulse Output sequence shown in Figure 12.2 with the timing of the sampler but the amplitudes of the "SineWave" at the sampling instants is shown in the frequency domain in Figure 13.1; it contrasts with the single "SineWave" source spectrum in Figure 13.0 previously shown as the single line. The Sampling waveform "convolves" the "SineWave" single line spectrum into spectral line pairs around each multiple of the sampling frequency $\boldsymbol{F_S}$.

The convolution occurs for every spectral line at all multiples of $\boldsymbol{F_S}$ and maintains the sinx/x envelope characteristic of the Sampler including the higher frequency parts of the spectrum as Figure 13.2 illustrates:

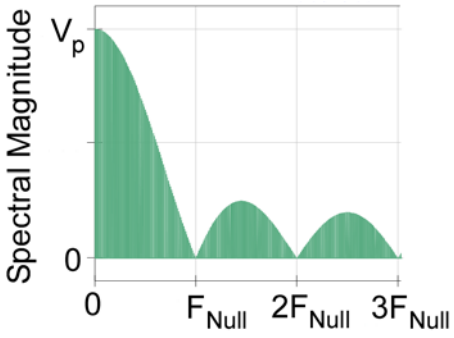

Figure 13.2 – Broad Spectral Lines from the Sampled "SineWave" Spectrum

An exact reconstruction of the original "SineWave" signal is theoretically possible from the sequence of samples, but that possibility is of little practical use for control purposes because the filter required for reconstruction requires a sinx/x impulse response which is non-causal and cannot be realized using an LTI filter.

14.0 Minimum Sampling Frequency for Analog "SineWave" Signals-

To accurately reproduce a "SineWave" Signal, each period of the "SineWave" Waveform must be represented by at least two samples. That means the sampling rate must be higher than twice the highest "SineWave" Signal frequency. One simple way to justify this requirement is to revisit Figure 13.1 and note the spectral line pair around F_S in relation to the spectral line $F_{SineWave}$ from the "SineWave" Waveform itself.

In that part of the spectrum between DC and F_S, two spectral lines appear one at $F_{SineWave}$, and one at (F_S - $F_{SineWave}$). We see that as $F_{SineWave}$ approaches 1/2F_S, the two spectral lines meet midway and are indistinguishable from behaviors with $F_{SineWave}$ above 1/2F_S. The frequency of 1/2F_S is called the Nyquist frequency and the effect of sampling at less than twice that frequency is called "Aliasing." F_S is usually chosen to be significantly higher than the Nyquist rate for any frequencies of interest to avoid Aliasing effects.

15.0 Approximate Sampled Analog "SineWave" Signals-

If the value of each sample is maintained until the next sample is taken, a "stairstep" approximation of the "SineWave" signal results as shown in Figure 15.0 and is labeled as a Zero-Order-Hold (ZOH) response:

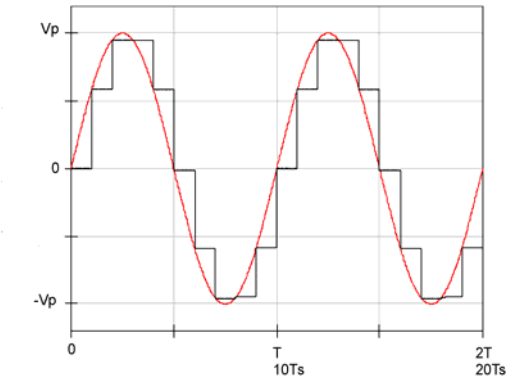

Figure 15.0 – Sampled "SineWave" with a Zero-Order-Hold Filter Waveform

The Zero-Order-Hold (ZOH) is equivalent to a filter with a pulse response of unit height and pulse width of one sample period T_S, resulting in a causal filter. Unfortunately the causality also introduces a delay of ½ the pulse width or T_S /2 for the entire spectrum.

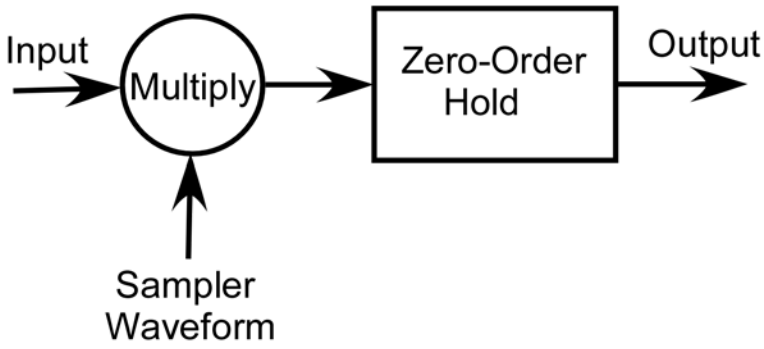

Figure 15.1 – Sampler with Zero-Order-Hold Filter System

We show the spectrum in Figure 15.2 of the discrete-time sampled "SineWave" signal of Figure 15.01 with the higher-order spectral lines greatly reduced by the ZOH filter.

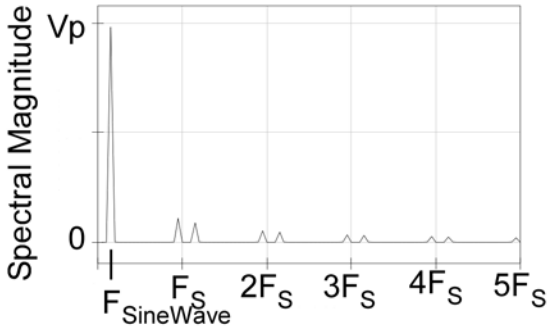

Figure 15.2 – Spectrum of the Sampled "SineWave" with a Zero-Order-Hold Filter

For the system described by ***H(f)***, the single pole asymptotic high frequency behavior is likely to reduce response near F_S caused by the ZOH at the H(f) input to negligible levels.

16.0 Excess Phase introduced in by ZOH action-

Delay of a signal by a half the T_S period by the pulse response necessarily introduces that delay into the "Open Loop" response. We saw in Figure 3.0 that delay can be expressed as phase. A full cycle of delay is 360°, and a half-cycle delay is equivalent to 180° phase shift. The ZOH delay produces a 180° phase shift at the Nyquist frequency of F_S /2. The delay produces a proportional phase shift at lower frequencies as given by:

$$\theta_{ZOH}(f) = 180^o \frac{f}{f_{Nyquist}} = 360^o \frac{f}{F_s} \qquad [16.0]$$

We see that choosing $\boldsymbol{F_S = 10 f_{Max}}$ still implies that 36° phase shift will be introduced into a carefully designed Analog Control Loop by the ZOH in the Analog to Digital conversion process.

For the PZ Compensated "Open Loop" of Figure 9.0, we had a unity-gain frequency of ~1,000 Hertz and ~75° phase margin. If we allow a ZOH to introduce no more than 10° phase shift, we must ensure that the sampling frequency exceeds a minimum:

$$\theta_{ZOH}(1{,}000) \le 10^{o} \ge 360^{o}\frac{1000}{F_s} \Rightarrow F_S \ge 36{,}000 \qquad [16.1]$$

Clearly, then, we need more than 36 samples per cycle of the highest frequency just to maintain reasonable stability. A higher sampling rate would be preferred if possible.

17.0 D2A Phase introduced in the $G_{PZ}(f)*H(f)$ "Open Loop" by ZOH action-

For the PZ Compensated "Open Loop," if we also employ a Digital to Analog (***D2A***) converter at the input of the ***H(f)*** Analog System, we will encounter a second ZOH to convert the digital control signal back into an Analog Signal. That ZOH, too, will introduce a phase shift of its own and require consideration of its effect on stability.

18.0 Digital Control of *H(f)* -

Armed with the well-understood Analog Control System shown in Figure 8.0, we introduce A2D and D2A System Blocks and proceed with converting the design.

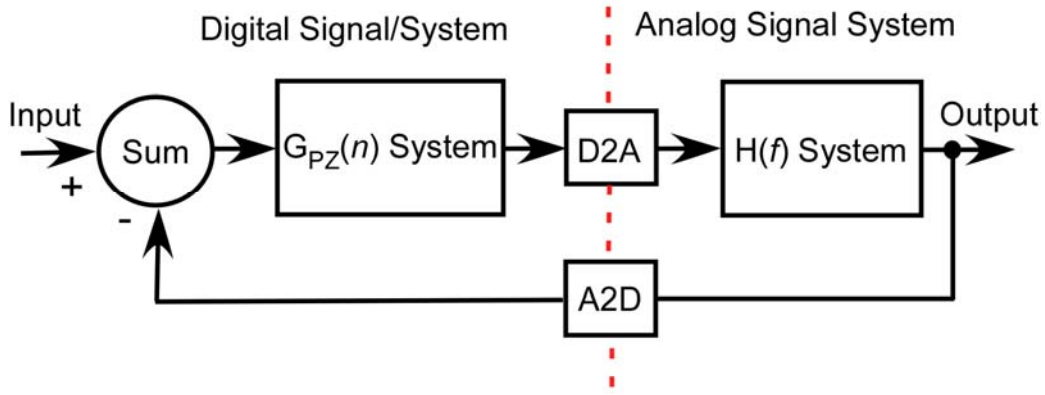

Figure 18.0 – Digital Control System for the *H(f)* Analog Block

The insertion of the Analog to Digital (A2D) Block and the Digital to Analog (D2A) Block has been discussed in the context of the loop stability issues caused by the delays of those new

blocks in the signal paths. Delay introduced in a feedback control system is the most important issue, but it is not the only issue for conversion of functionality.

It is not common to design digital signal processing systems in the frequency domain, but rather in the time domain. We will employ the "Bilinear Transform" to convert the $G_{PZ}(f)$ to an equivalent $G_{PZ}(n)$ for digital implementation. We will also show that noise introduction into the loop is the major consideration associated with converting the magnitudes to discrete values.

19.0 Delay Introduced in the *Input* Path by an additional ZOH action-

For the PZ Compensated "Closed Loop" of Figure 18.0, if we also employ a Digital to Analog (***D2A***) converter at the input of the Digital System, we will encounter another ZOH to convert an analog control signal into a digital counterpart. That ZOH, too, will introduce a delay of its own but will have no direct effect on stability of this system.

If, however, this system is employed in the context of other "outer" control loops, the delay may contribute to stability issues elsewhere.

20.0 Approach to Convert $G_{PZ}(f)$ to $G_{PZ}(n)$-

There exists a direct, formal, algebraic method of converting a system such as that described by equation 8.1 (repeated here for convenience) to a digital form using the "Bilinear Transform," as well as a clear indirect method. We explore the indirect method but retain the favorable stability behaviors guaranteed by the use of the "Bilinear Transform."

$$G_{PZ}(f) = 10^4 \bullet \frac{1 + j(F_Z)}{1 + j(F_P)} \bullet \frac{1}{1 + j(F_G)} \qquad [8.1]$$

21.0 System Diagram Forms-

We show a methodology for representing systems that is simple to understand and utilize. Refer to Figure 21.0 for a "First-Order System" Diagram.

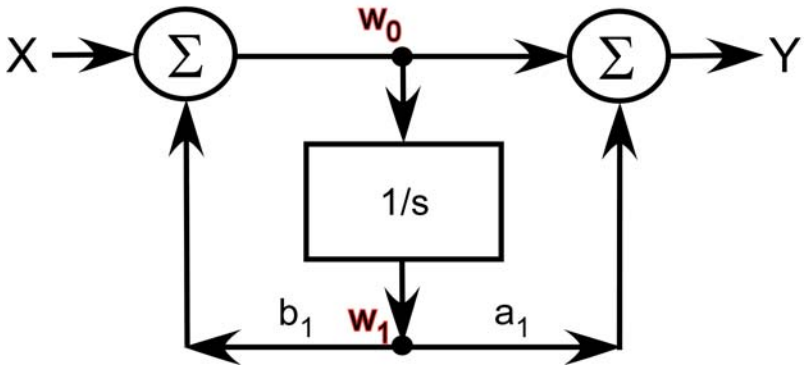

Figure 21.0 – First-Order PZ System Diagram

We construct a set of describing equations as follows:

$$w_0 = X + b_{1PZ} w_1 \qquad [21.0]$$

$$Y = w_0 + a_{1PZ} w_1 \qquad [21.1]$$

The block labeled ***1/s*** is an integrator in analog systems, or a unit delay in discrete-time systems.

$$w_1 = \frac{1}{s} w_0 \qquad [21.2]$$

$$w_0 = s w_1 \qquad [21.3]$$

We substitute equation 21.3 into equation 21.0:

$$s w_1 = X + b_{1PZ} w_1 \qquad [21.4]$$

$$\left[s - b_{1PZ} \right] w_1 = X \qquad [21.5]$$

$$\frac{w_1}{X} = \frac{1}{[s - b_{1PZ}]} \quad [21.6]$$

$$\frac{w_0}{X} = \frac{s}{[s - b_{1PZ}]} \quad [21.7]$$

$$Y = w_0 + a_{1PZ} w_1 = \frac{s}{[s - b_{1PZ}]} X + \frac{a_{1PZ}}{[s - b_{1PZ}]} X \quad [21.8]$$

$$Y = \frac{[s + a_{1PZ}]}{[s - b_{1PZ}]} X \quad [21.9]$$

$$PZ[s] = \frac{Y}{X} = \frac{[s + a_{1PZ}]}{[s - b_{1PZ}]} \quad [21.10]$$

$$PZ[s]_{s \to j2\pi f} = \frac{[j2\pi f + a_{1PZ}]}{[j2\pi f - b_{1PZ}]} \quad [21.11]$$

We note that this is the required form for the PZ compensator, knowing that the ***1/s*** block describes a realizable LTI Integrator.

We rearrange the equation, add a scaling constant $\boldsymbol{K_{PZ}}$, set the coefficients, and solve:

$$K_{PZ} \frac{[a_{1PZ} + j2\pi f]}{[-b_{1PZ} + j2\pi f]} = \frac{[1 + jF_Z]}{[1 + jF_P]} \quad [21.12]$$

We justify the "Real" coefficients in the numerator and denominator:

$$K_{PZ} \frac{a_{1PZ}\left[1 + \frac{j2\pi f}{a_{1PZ}}\right]}{-b_{1PZ}\left[1 + \frac{j2\pi f}{-b_{1PZ}}\right]} = \frac{[1 + jF_Z]}{[1 + jF_P]} \quad [21.13]$$

$$\frac{2\pi f}{a_{1PZ}} = F_Z = \frac{f}{f_{zero-Z}} \Rightarrow a_{1PZ} = 2\pi f_{zero-Z} \quad [21.14]$$

$$\frac{2\pi f}{-b_{1PZ}} = F_P = \frac{f}{f_{pole-P}} \Rightarrow b_{1PZ} = -2\pi f_{pole-P} \quad [21.15]$$

Because the PZ compensator has unity gain in $\boldsymbol{G_{PZ}(f)}$, we know:

$$K_{PZ}\frac{a_{1PZ}}{-b_{1PZ}} = 1 \Rightarrow K_{PZ} = -\frac{b_{1PZ}}{a_{1PZ}} = -\frac{-2\pi f_{pole-P}}{2\pi f_{zero-Z}} = \frac{f_{pole-P}}{f_{zero-Z}} \quad [21.16]$$

22.0 Operational Amplifier System Diagram Coefficients-

In the same way that we constructed the "Analog Open Loop" using a cascade of G(f)*H(f), we construct another first-order model for the Operational Amplifier, but we will simply solve again for a new set of coefficients as follows:

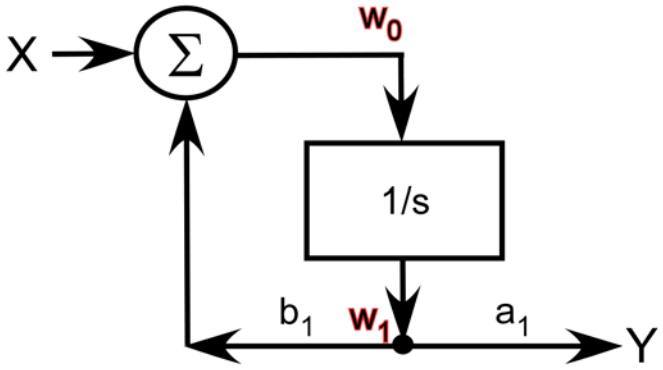

Figure 22.0 – First-Order Single-Pole System Diagram

As before, we construct a set of describing equations as follows:

$$w_0 = X + b_{1G}w_1 \quad [22.0]$$

$$Y = a_{1G}w_1 \quad [22.1]$$

$$w_1 = \frac{1}{s} w_0 \qquad [22.2]$$

$$w_0 = s w_1 \qquad [22.3]$$

We substitute equation 22.3 into equation 22.0:

$$s w_1 = X + b_{1G} w_1 \qquad [22.4]$$

$$\left[s - b_{1G}\right] w_1 = X \qquad [22.5]$$

$$\frac{w_1}{X} = \frac{1}{\left[s - b_{1G}\right]} \qquad [22.6]$$

$$Y = \frac{a_{1G}}{\left[s - b_{1G}\right]} X \qquad [22.7]$$

$$G[s] = \frac{Y}{X} = \frac{a_{1G}}{\left[s - b_{1G}\right]} \qquad [22.8]$$

$$G[s]_{s \to j2\pi f} = \frac{a_{1G}}{\left[j2\pi f - b_{1G}\right]} \qquad [22.9]$$

$$K_G \frac{a_{1G}}{-b_{1G}\left[1 + \frac{j2\pi f}{-b_{1G}}\right]} = 10^4 \bullet \frac{1}{\left[1 + jF_G\right]} \qquad [22.10]$$

$$\frac{2\pi f}{-b_{1G}} = F_G = \frac{f}{f_{pole-G}} \Rightarrow b_{1G} = -2\pi f_{pole-G} \qquad [22.11]$$

Because the Operational Amplifier gain in ***G(f)*** is 10,000 (10^4), we know:

$$K_G \frac{a_{1G}}{-b_{1G}} = 10^4 \Rightarrow K_G = -10^4 \bullet \frac{b_{1G}}{a_{1G}} = -10^4 \bullet \left(-2\pi f_{pole-G}\right) = 2\pi \bullet 10^4 \bullet f_{pole-G} \qquad [22.12]$$

23.0 PZ Compensated Operational Amplifier System Diagram –

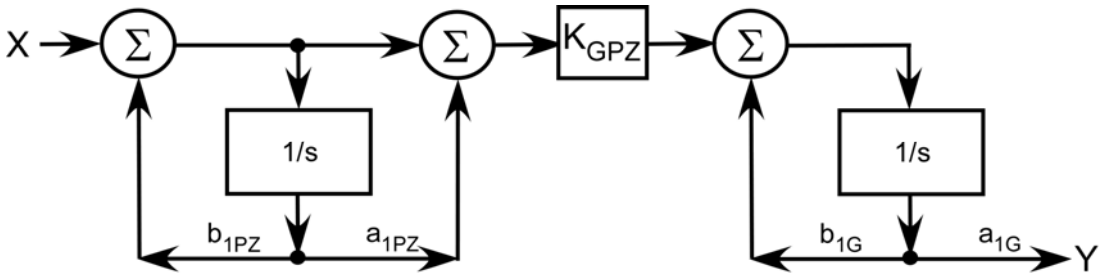

Figure 23.0 – $G_{PZ}(f)$ First-Order Single-Pole System Cascade

We have developed a System-Level Model for our Analog Control System, but it is still an analog version, primarily because we employ the continuous-time integrators denoted as ***1/s*** System Blocks.

24.0 Discrete-Time Integrator using the Bilinear Transform –

We will now replace analog integrator Blocks with discrete-time Bilinear-Transform equivalent Integrators. We employ the Bilinear Transform Blocks because the result of transforming a stable Analog System will always produce a stable Digital System.

The same general first-order Block Diagram applies to achieve the Bilinear Transform, with the notable exception that the Laplace ***1/s*** integrator Block is replaced by a z^{-1} Delay Block, as illustrated in figure 24.0 as follows:

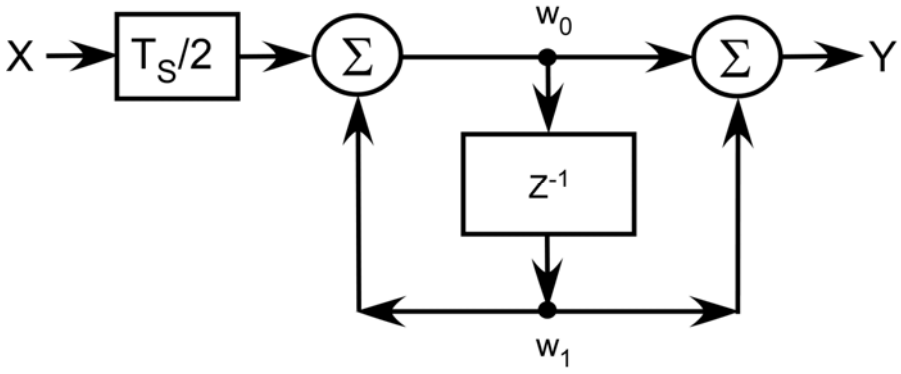

Figure 24.0 – Bilinear Transform Integrator System Block Diagram

The Bilinear Transform is a relatively simple mapping from the ***s***-domain to the ***z***-domain. It is invertible and defined by the following substitution relationships:

$$s = \left(\frac{2}{T_S}\right)\frac{1 - z^{-1}}{1 + z^{-1}} \qquad [24.0]$$

$$z = \frac{1 + \left(T_S/2\right)s}{1 - \left(T_S/2\right)s} \qquad [24.1]$$

The Bilinear Transform is based on a trapezoidal integration and is calculated by the System represented by the Block Diagram of figure 24.0 as follows:

$$w_0 = \left(\frac{T_S}{2}\right)X + w_1 \qquad [24.2]$$

$$Y = w_0 + w_1 \qquad [24.3]$$

$$w_1 = z^{-1}w_0 \qquad [24.4]$$

$$w_0 = zw_1 \qquad [24.5]$$

We substitute equation 24.5 into equation 24.2:

$$zw_1 = \left(\frac{T_S}{2}\right)X + w_1 \qquad [24.6]$$

$$[z - 1]w_1 = \left(\frac{T_S}{2}\right)X \qquad [24.7]$$

$$\frac{w_1}{X} = \left(\frac{T_S}{2}\right)\frac{1}{[z-1]} \qquad [24.8]$$

$$\frac{w_0}{X} = \left(\frac{T_S}{2}\right)\frac{z}{[z-1]} \qquad [24.9]$$

$$Y = w_0 + w_1 = \left(\frac{T_S}{2}\right)\frac{z}{[z-1]}X + \left(\frac{T_S}{2}\right)\frac{1}{[z-1]}X \qquad [24.10]$$

$$Y = \left(\frac{T_S}{2}\right)\frac{[z+1]}{[z-1]}X = \left(\frac{T_S}{2}\right)\frac{\left[1+z^{-1}\right]}{\left[1-z^{-1}\right]} \qquad [24.11]$$

We have constructed a model for the discrete-time integrator and will next substitute it for the continuous-time integrator.

25.0 PZ Compensated Digital System Diagram –

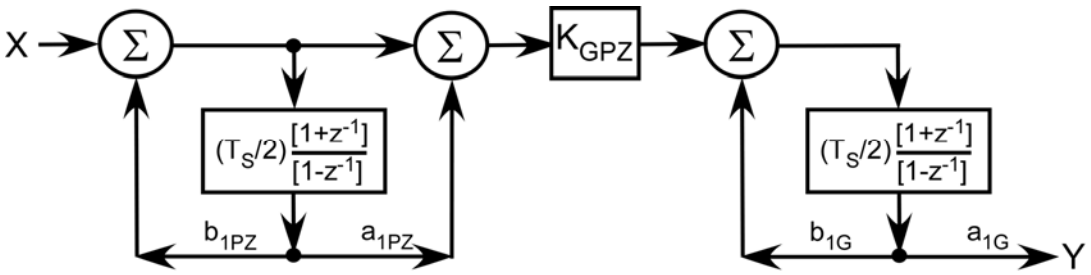

Figure 25.0 – *$G_{PZ}(z)$* First-Order Single-Pole Digital System Cascade

Figure 25.0 above shows the System Block Diagram for the discrete-time version of the Analog Control Blocks. Each coefficient has already been developed, except the $\boldsymbol{K_{GPZ} = K_{PZ} * K_G}$ to avoid duplication.

26.0 PZ Compensated Digital System Implementation –

The Digital System described above can be implemented using several techniques including a custom Application Specific Integrated Circuit (ASIC), a Field Programmable Gate Array

(FPGA), the new Programmable System on a Chip (PSoC), or as a program in a MicroController Unit (MCU) such as a TI OMAP family processor or a MicroChip PIC family unit.

For an ASIC or FPGA design, the Block Diagram could be expressed as an electrical schematic or as a textual description in a Digital Synthesis language such as VHDL or Verilog. An ASIC design requires a longer development time, including mask specification and fabrication by an IC foundry. The FPGA allows a similar design process except the synthesis results in a connection map for logic on the FPGA device selected. In both cases, the result involves reduction of the design to a logic gate level from the higher level of abstraction that the System Bloch Diagram represents. ASIC designs are favored in highest volume, long-running cases because the unit cost is lowest although the Non-Recurring Engineering (NRE) and tooling costs are high. The FPGA designs are favored in lower volume and market study cases because the higher unit cost is offset by lower tooling costs and considerably less time and cost in retooling for design changes.

For a MCU implementation, the A2D and D2A Blocks are resources that are already included as options in the MCU selection process. The Block Diagram is translated to a program for the MCU unit selected. The MCU is considerably more complex than an FPGA, may include the conversion resources and has higher unit costs than either an ASIC or FPGA implementation but offers the most implementation flexibility.

The PSoC device is somewhat of a hybrid, and allows continuous magnitude samples to be processed without conversion to discrete magnitudes. In that regard, it is not a truly Digital implementation, but offers some advantages for the skilled designer.

26.0 Bilinear Transform Integrator Logic Schematic Example –

A logic-level implementation is a relatively simple translation of the Block Diagram. The Bilinear Transform Integrator is used as an example as follows:

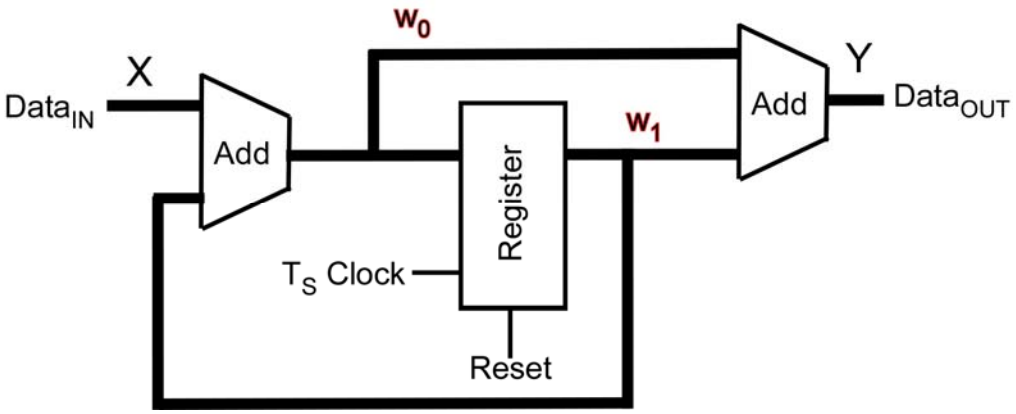

Figure 26.0 – Register-Level Bilinear Transform Logic Implementation

The signal paths are multi-bit digital quantities that are implementation specific. A discussion of the requirements for selection of the number of bits resolution is deferred until later, but note that the sum of two digital words generates an extra bit of width in the "carry" signal. The system implementation must be designed for the worst-case dynamic range to prevent overflow truncation of the digital signals.

The T_S Clock input synchronizes the block with the rest of the blocks and the ***Reset*** input initializes the Register on start-up. Generally, other blocks are needed to deal with the power-on/off sequencing, power supply checking, Enable inputs and other "Supervisory Functions" that are application specific.

Scaling-factor and gain blocks are implemented as combinational logic and can be judiciously inserted into the signal paths above to implement each other block of the digital controller.

A Register-Level ASIC or FPGA solution would be implemented for economic reasons or in applications needing very high T_S Clock rates.

27.0 Bilinear Transform Integrator Verilog RTL Example –

Modern Digital Logic Synthesis practice permits the use of a high-level Hardware Design Language (HDL), including VHDL and Verilog to describe an implementation without drawing a schematic diagram. The code is compiled to produce a "Netlist" with objects mapped from a library of logic gates or FPGA programmable block descriptions.

The following is a Register Transfer Level (RTL) description of the Bilinear Transform Integrator Block written in Verilog:

```
/*************************************************************/
// MODULE:          Bilinear Transform Integrator
//
// FILE NAME:       bti_rtl.v
// VERSION:         1.0
// DATE:            February 31, 2020
// AUTHOR:          Raymond L. Barrett, Jr.
//
// CODE TYPE:       Register Transfer Level
//
// DESCRIPTION:     This Module Implements an Unscaled Bilinear
// Transform Integrator Core
//
/*************************************************************/

// DEFINES
`define DEL    1              // Clock-to-output delay. Zero
                              // time delays can be confusing
                              // and sometimes cause problems.
`define OP_BITS 4     // Number of bits in each operand

// TOP MODULE
module BLT(
                clk,
                reset,
                x,
                y);

// INPUTS
input                           clk;            // Clock
input                           reset;          // Reset input
input [`OP_BITS-1:0] x;                         // Signal Input

// OUTPUTS
```



```
output [2*`OP_BITS-1:0]y;          // Signal Output

// INOUTS

// SIGNAL DECLARATIONS
wire                               clk;
wire                               reset;
wire                               y;
wire [`OP_BITS-1:0]          w0;
reg  [2*`OP_BITS-1:0]        w1;

// PARAMETERS

// ASSIGN STATEMENTS
assign #`DEL w0 = x + w1;
assign #`DEL y = w0 + w1;

// MAIN CODE

// Look at the rising edge of the clock
// and the rising edge of reset
always @(posedge reset or posedge clk) begin
        if (reset)
                w1 <= 0;
        else
                w1 <= w0;
end
endmodule // BLT
```

The code above is for illustration only and is not guaranteed to produce a useful device. It is defined with a “DEFINE” block so that bit width and salient time delays can be changed in one location and used throughout the code. The “module” declaration assigns a name to this block and lists the signal names connected as inputs and outputs. The “INPUTS” section defines which signals are inputs and the number of bits width for multi-bit signals. The “OUTPUTS” section defines which signals are outputs and the number of bits width for multi-bit signals. No “INOUT” or bi-directional signal are used or listed.

The "SIGNAL DECLARATIONS" section lists the signal properties of each signal name with the "wire" declaration implying no memory associated with a signal name and the "reg" declaration implying that a signal is held remembered between defining events to change that signal.

No "PARAMETERS" are assigned or used. The list of "ASSIGN STATEMENTS" define the algebraic relationship between signals that are derived from other signals and allows the attachment of a delay to represent the propagation delay of the associated computation.

The "MAIN CODE" section defines the operational activities of the module and particularly the assignment of values to the memory associated with the register as the variable "w_1" represents.

Each section follows the syntax of the Verilog language definition using the ";" character to denote the end of a line and the "end" statement to delineate the termination of a concurrent block of code. The "module" and "endmodule" keywords are used to inform the compiler as to the location of the beginning and end of the definition of this functional block.

Just as with the Verilog example, VHDL has its own syntactical requirements to produce a correct result. Both are supported by design entry tools that perform a check of the syntax with some indication of where errors may be found in the code. Simulators are available that allow checking of the code for correct function and possible errors that can only be detected by operating the system.

28.0 Bilinear Transform Integrator MCU Programs –

Producing a program for MCU implementation is much like the Verilog or VHDL code. The high-level programming language "C" appears much like Verilog and is compiled in a similar syntax checking and simulation environment to support program development.

For those applications that require higher speeds, it may be required to write programs directly in the "assembly language" of the MCU chosen. The tools that support "C" language development produce assembly language as an intermediate level, but it is not often optimized to produce the highest performance. In some cases, particular modules alone may be written in assembly language and invoked from the "C" program to get good speed and good programmer productivity too.

One advantage of writing programs in a high-level language like "C" is that they can often be used on differing MCU units from the same manufacturer, or from different MCU manufacturers

and on different MCU architectures with little or no changes. Assembly language programming is far less "portable" between MCU devices.

29.0 Digital Word Bit Width –

Digital quantities are represented by some number of bits, each representing a twice the resolution in relation to an adjacent bit in the digital word. Initially, we are faced with deciding the level of resolution required or possibly whether the bit width offered by a particular implementation is sufficient. MCU units are available in 8-bit, 16-bit, and even 32-bit architectures. However, even an 8-bit MCU can be programmed to represent 32-bit quantities at the cost of slower speed of computation.

A2D and D2A converters are available in a wider range of bit-widths and speeds and present more of an issue for choosing signal representation.

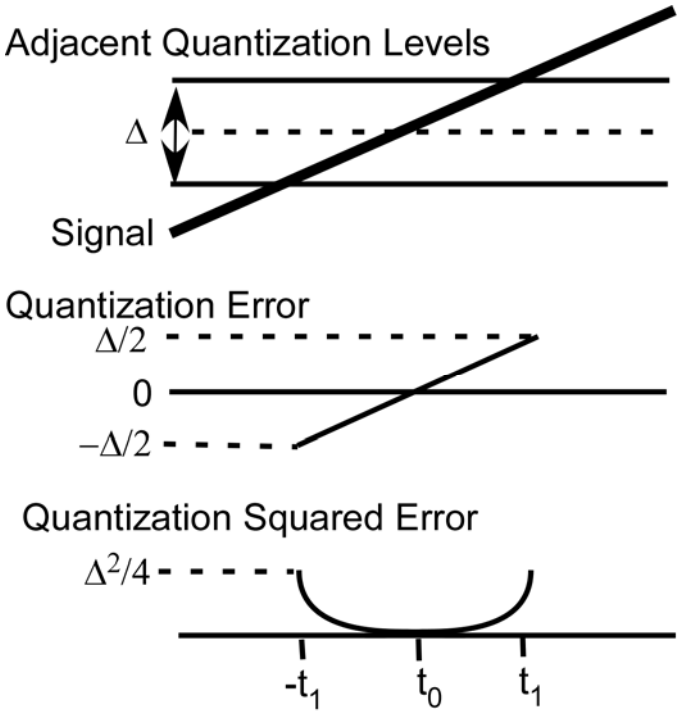

Figure 29.0 – Quantizing Error Quantities

We will continue with the assumption that we are dealing with Sinusoidal signals and define the signal above as a segment from that "Sinusoid."

$$V_{Sine} = V_p \sin(2\pi F t) \qquad [29.0]$$

The power of the waveform is:

$$P(V_{Sine}) = \frac{V_p^2}{2} \qquad [29.1]$$

We can write the error ε as:

$$\varepsilon(t) = \frac{\Delta}{2t_1} t \qquad [29.2]$$

Using a "dummy variable" λ, we can integrate the square error over the time interval to find the mean-square error, or noise power associated with the error as:

$$P(\varepsilon) = \frac{1}{t_1} \int_0^{t_1} \left(\frac{\Delta}{2t_1} \right)^2 \lambda^2 d\lambda = \frac{\Delta^2}{12} \qquad [29.3]$$

The "Sinusoid" has a signal range of $2V_p$ and for an n-bit word, we know that there are 2^n levels. It follows, then that:

$$\Delta = \frac{2V_p}{2^n} = 2^{(1-n)} V_p \qquad [29.4]$$

Each of the 2^n levels contributes equal average mean-square power, despite each being of different duration, so the average powers is:

$$P(\varepsilon) = \frac{\Delta^2}{12} = 2^{2(1-n)} \frac{V_p^2}{12} \qquad [29.5]$$

We now can find the "Signal to Noise Ratio" (SNR) as:

$$SNR = \frac{P(V_{Sine})}{P(\varepsilon)} = \frac{\frac{V_p^2}{2}}{2^{2(1-n)} \frac{V_p^2}{12}} = \frac{\frac{1}{2}}{2^{2(1-n)} \frac{1}{12}} = \frac{6}{4 \bullet 2^{-2n}} = \frac{3}{2} \bullet 2^{2n} \qquad [29.6]$$

It is common to express the power ratio in decibel (dB) units as follows:

$$10\log_{10} SNR = 10\log_{10}\left(\frac{3}{2}\bullet 2^{2n}\right) = 10\log_{10}\left(\frac{3}{2}\right) + 20n\log_{10}(2) \qquad [29.5]$$

$$10\log_{10} SNR = 1.76 + 6.02n \qquad [29.6]$$

This important result shows a 6dB improvement for each added bit of width. For an 8-bit word width, the SNR is ~50dB, or less than 0.5% but only at the maximum signal. Likewise, a for a 16-bit word width, the SNR is ~98dB, or about 0.001% but only at the maximum signal.

The noise generated by the Quantization of a continuous magnitude signal into discrete levels produces noise in the system that is different from any other noise sources in any remaining analog system components. Both the A2D and D2A produce these noise effects and should be analyzed for independent effects.

30.0 Summary and Conclusions–

30.1 - We have characterized an Analog Feedback Control System comprised of an original single (or dominant) pole model ***H(f)***, and added a representative Operational Amplifier Feedback Control Block with Pole-Zero (PZ) $\boldsymbol{G_{PZ}(f)}$ compensation, showing the need from "Open Loop" stability arguments. We showed the "Closed Loop" response of the system in the frequency domain.

30.2 - We showed the importance of setting a sampling rate at a high enough frequency to determine delay effects from the Zero-Order Hold (ZOH) nature of the sampling on the loop stability and also high enough to avoid Aliasing effects.

30.3 – We introduced a generalized Block Diagram methodology for translating the Analog System behavior into a structured System Diagram with continuous-time integrators. We developed a discrete-time model of the Bilinear Transform integrator and substituted it for the continuous-time integrator, preserving the system stability of the Analog System.

30.4 – We discussed issues for ASIC, FPGA, and MCU implementation alternatives, showing a high-level logic circuit schematic and a Verilog code example fro implementation of the Bilinear Transform integrator Block. We discussed the similarity of the Verilog model to MCU code written using the "C" language and the possibilities for assembly language programming of MCU units.

30.5 – We discussed the relationship between digital word bit-width and the noise associated with Quantization error of the discrete magnitude levels.

In conclusion, we have covered the continuous-time to discrete-time issues, the continuous-magnitude to discrete-magnitude issues, showed a methodology for conversion, and implementation of "Converting Feedback Systems from Analog to Digital Control."

