PW2 Practical Considerations for Digital Control for Power Converters

Kip Haggerty, Ph.D., P.E.

H&A Systems Engineering

Controls • Communications • Sensors

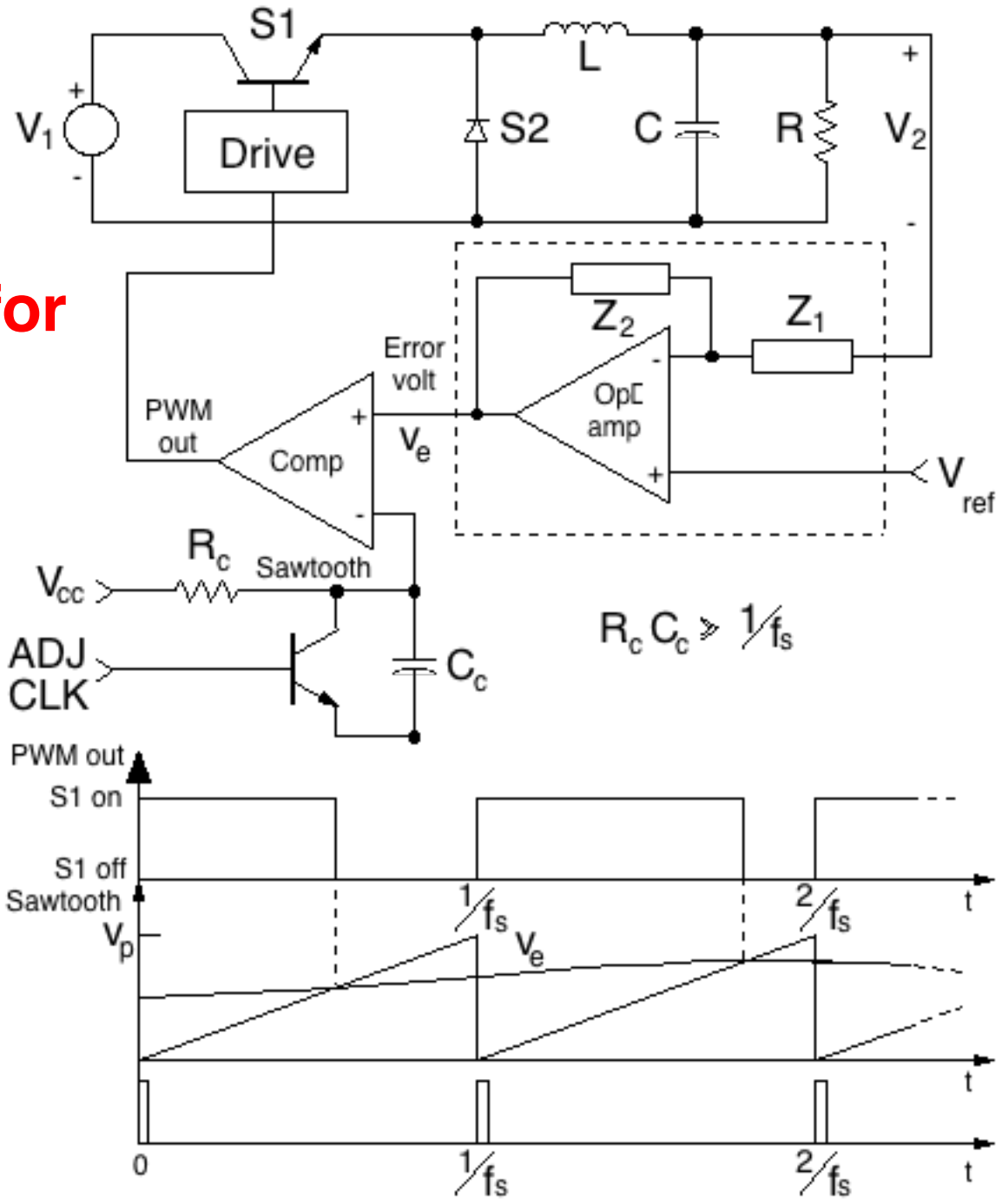
P.O. Box 2875 • El Segundo, CA 90245
(310) 607-9609 • http://www.hasys.com/

Download at: http://www.hasys.com/tutorials.html

Outline

- Quick Comparison Analog vs. Digital Control
- Design Approach
- Difference Equations
  - Derivation
  - Clamping and Preset
  - Cascading
- Timeline and Sampling Issues
- Reference to Plant Modeling

Analog Control for Buck Converter is Relatively Simple



Analog Controls Are Often a Good Choice

Advantages

- Wider Bandwidth than Digital
- Higher Resolution
- Easier Design

Disadvantages

- Degradation Due to Component Aging
- Component Temperature Drift
- More Parts (Lower Reliability)
- Limited Upgrades
- Difficulty of Integrator Clamping and Preset
- No State Clamping and Preset

Digital Controls Provide Unique Capabilities at a Cost

Advantages

- No Drift in Digital Part
- Precise Well-defined Behavior
- Software Upgrade Path
- Less Parts (Higher Reliability)
- Easy State Clamp and Preset
- Advanced Control Algorithms
  - Adaptive Control
  - Observers
  - Kalman Filtering

Disadvantages

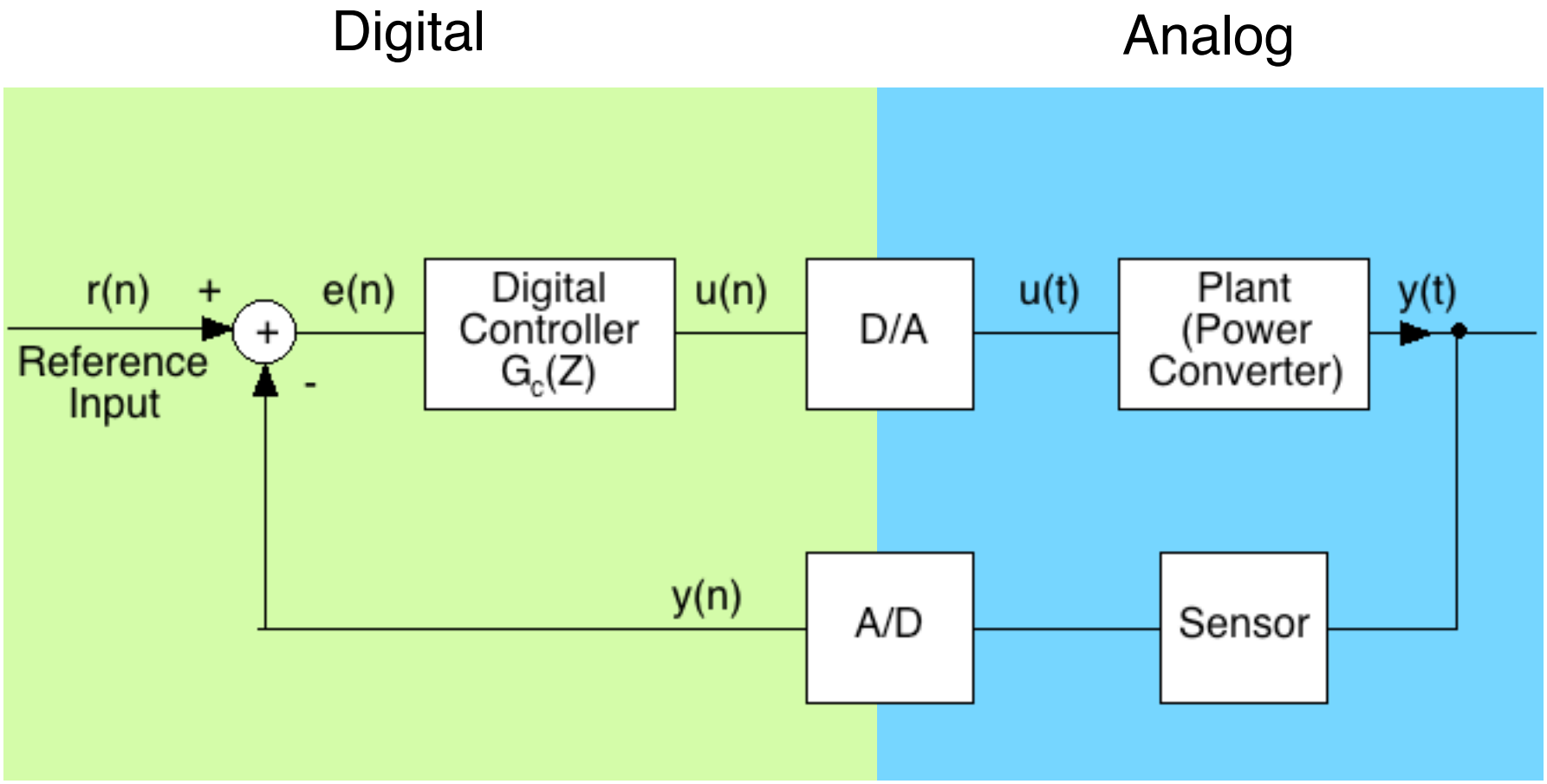
- Latency
- Limited Bandwidth

$$f_{BW} \leq \frac{f_S}{10} \text{ to } \frac{f_S}{6}$$

- Harder Design

Digital Control Block Diagram

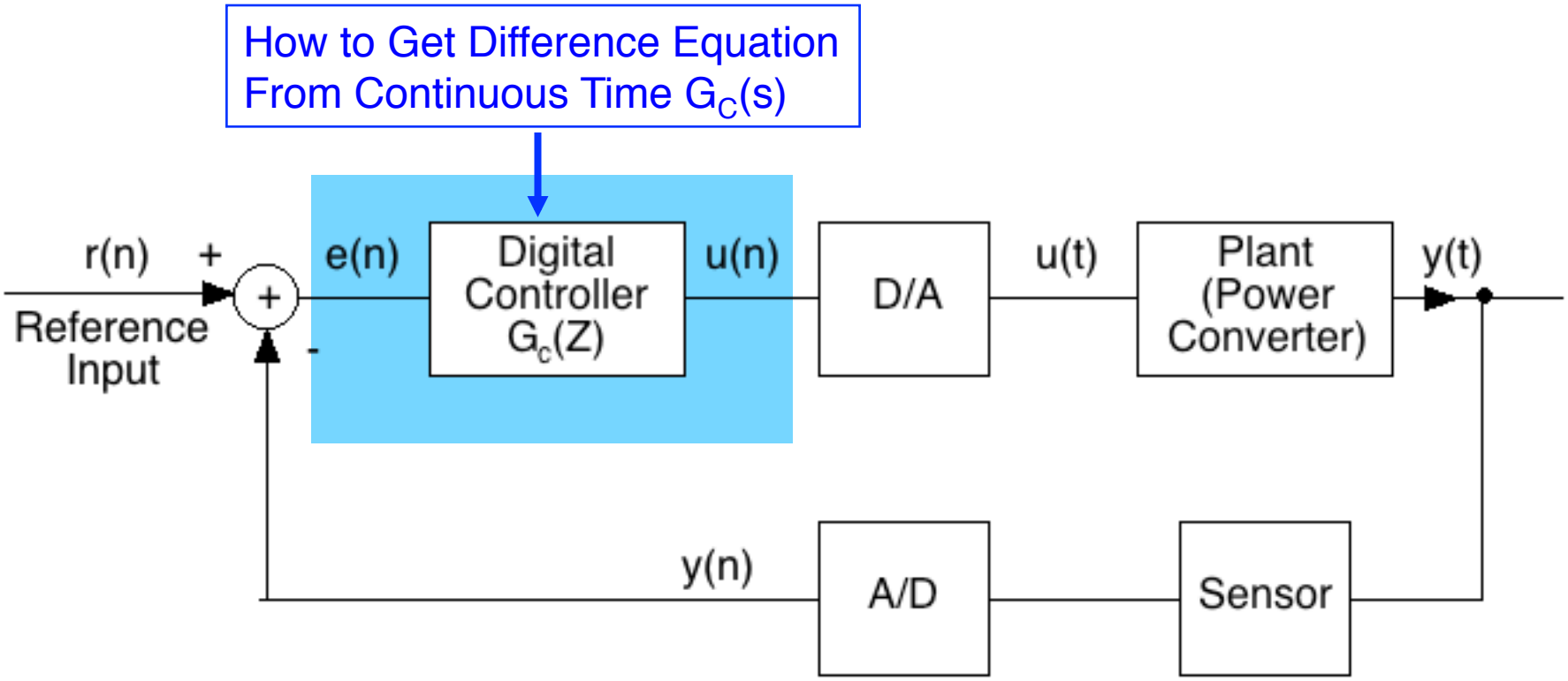
NOTE: Power Converters Are Analog



One Design Approach

- Analyze "Plant" (i.e. the Power Converter) in Analog (Continuous Time) Domain
  - Look for Type of Controller Needed
- Select Feedback Controller (PID, PI, Integrator, LPF, etc.)
- Analyze Continuous Time Closed Loop System with Pure Time Delay
  - Is Processing Interval Short Enough to Provide Desired Bandwidth?
- Convert Feedback Controller to Discrete Time
  - Check Digital Resolution
  - Check for Numerical Problems in Computations
  - Check if Resolution Expansion by Averaging is Required

Deriving Digital Control Difference Equations from Analog Control Blocks



There are Multiple Methods of Digitization, Including:

- Backward difference
  - Approximates derivative
- Forward Difference
  - Approximates integral
- Combined Forward/Backward Difference
  - Better derivative estimate
- Bilinear transform
  - End point trapezoidal integration
- Impulse invariance
  - Sampled version of impulse response
- Transition Matrix Method
  - Solve first order matrix differential equation
- Bootstrap Method (overkill)
  a. Use x_k and x_{k-1} to compute x'_k
  b. Use forward difference to compute x_{k+1}
  c. Use backward difference to compute x'_{k+1}
  d. Use bilinear transform to recompute x_{k+1}

Multiple Methods of Digitization Provide Slightly Different Difference Equations

- **Backward difference** $\dot{y}_k = \frac{y_k - y_{k-1}}{T} \quad s = \frac{1 - z^{-1}}{T} \quad z = \frac{1}{1 - s \cdot T}$
- **Forward Difference** $\dot{y}_k = \frac{y_{k+1} - y_k}{T} \quad s = \frac{z - 1}{T} \quad z = 1 + s \cdot T$
- **Combined Forward/Backward** $\dot{y}_k = \frac{y_{k+1} - y_{k-1}}{2 \cdot T} \quad s = \frac{z - z^{-1}}{2 \cdot T}$
- **Bilinear transform** $\frac{\dot{y}_{k+1} + \dot{y}_k}{2} = \frac{y_{k+1} - y_k}{T} \quad s = \frac{2}{T} \cdot \frac{1 - z^{-1}}{1 + z^{-1}} \quad z = \frac{1 + \frac{T}{2} s}{1 - \frac{T}{2} s}$
- **Impulse invariance** $z = e^{s \cdot T}$
- **Transition Matrix Method** $\dot{y} = A \cdot y(t) + B \cdot x(t)$

$$y_{k+1} = e^{A \cdot T} \cdot y_k + \int_0^T e^{A \cdot (T - t)} \cdot B \cdot x(k \cdot T + t) \cdot dt = \Phi \cdot y_k + \Gamma_k$$

Example: A simple Integrator Illustrates a Potential Implementation Problem

- **Backward difference** $y_k = y_{k-1} + T \cdot x_k$
- **Forward Difference** $y_k = y_{k-1} + T \cdot x_{k-1}$
- **Bilinear transform** $y_k = y_{k-1} + \frac{T}{2} \cdot \left(x_k + x_{k-1} \right)$
- **This can lead to:**
  - **Time wasting arguments**
  - **Confusing requirements**
  - **Questions about which transformation to use for each compensator**
  - **Incorporation of transformation into delivered code**
  - **Software design and coding errors**
  - **Wasted time in integration**

Bilinear Transform Produces All First Order Compensators and Filters from the Same General Model

Continuous Time Transfer Function

$$H(s) = k \cdot \frac{\tau_z \cdot s + u_z}{\tau_p \cdot s + u_p} \quad \text{where}: \begin{array}{l} \tau_z, \tau_p \geq 0 \\ u_z, u_p = [0 \text{ or } 1] \\ k \neq 0 \end{array}$$

Bilinear Transformation

$$s = \frac{2}{T} \cdot \frac{1 - z^{-1}}{1 + z^{-1}} \quad \text{where}: T = \text{update time}$$

Discrete Time Transfer Function

$$H(z) = \frac{b_0 + b_1 \cdot z^{-1}}{1 - a_1 \cdot z^{-1}}$$

$$\text{where}: a_1 = \frac{2 \cdot \tau_p - u_p \cdot T_s}{2 \cdot \tau_p + u_p \cdot T_s}$$

$$b_0 = k \cdot \frac{2 \cdot \tau_z + u_z \cdot T_s}{2 \cdot \tau_p + u_p \cdot T_s}$$

$$b_1 = -k \cdot \frac{2 \cdot \tau_z - u_z \cdot T_s}{2 \cdot \tau_p + u_p \cdot T_s}$$

Difference Equation

$$y_n = a_1 \cdot y_{n-1} + b_0 \cdot x_n + b_1 \cdot x_{n-1} \quad or$$

$$u_n = a_1 \cdot u_{n-1} + b_0 \cdot e_n + b_1 \cdot e_{n-1}$$

Same Difference Equation Provides 6 Different First Order Compensators Depending on Model Parameters

Type:	k	τ_z	u_z	τ_p	u_p
Lead Compensation	$\neq 0$	$> \tau_p$	1	> 0	1
Lag Compensation	$\neq 0$	> 0	1	$> \tau_z$	1
Low Pass Filter (LPF)	$\neq 0$	$= 0$	1	> 0	1
Differential LPF	$\neq 0$	$= 1$	0	> 0	1
Integrator	$\neq 0$	$= 0$	1	1	0
Proportional plus Integral (PI)	$= k_I \neq 0$	$k_P/k_I \neq 0$	1	1	0

Clamping and Preset is Accomplished by Modifying the Stored State Regardless of Type

- **Accomplished by Clamping the Output and Saving Clamped Value as the Stored Previous State for the Next Iteration**

$$v_n = a_1 \cdot u_{n-1} + b_0 \cdot e_n + b_1 \cdot e_{n-1} \quad \Rightarrow \quad u_n = Max[u_{\min}, Min[u_{\max}, v_n]]$$

- **Clamping:**
  - **Clamp v_n before output and use previous clamped output as previous state**
  - **Output u_n is clamped**
  - **On next iteration, previous u_{n-1} state is clamped (no build-up)**
  - **Similar to analog integrator wind-up correction**
    - » **Same for pure integrator**
    - » **For PI, can be better than imitating analog devices**
- **Preset:**
  - **Compute initialization value**
  - **Store as previous state before first or next execution**

Let's Compare to an Alternative Implementation

- **Gleaned from TI Application Report SPRA589A on <http://www.ti.com/>**
- **It Implements Clamping for a PI as Follows:**

$$v_n = k_P \cdot e_n + I_{n-1}$$

Proportional Term

$$u_n = Max\{u_{\min}, Min[u_{\max}, v_n]\}$$

Clamped Control Signal

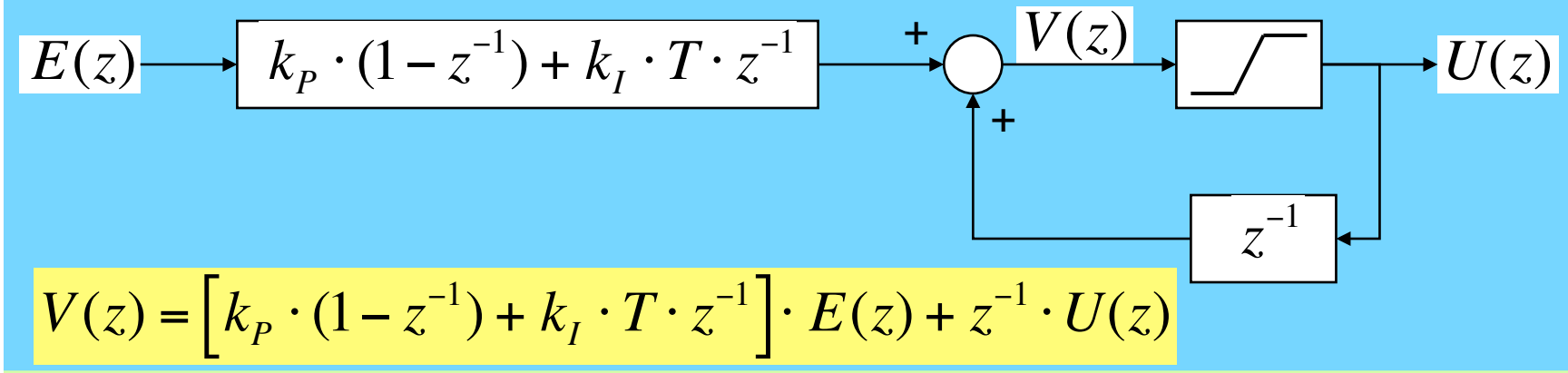
$$I_n = I_{n-1} + k_I \cdot T \cdot e_n + \frac{k_I \cdot T}{k_P}\left(u_n - v_n\right)$$

Integral Term with Wind-up Correction

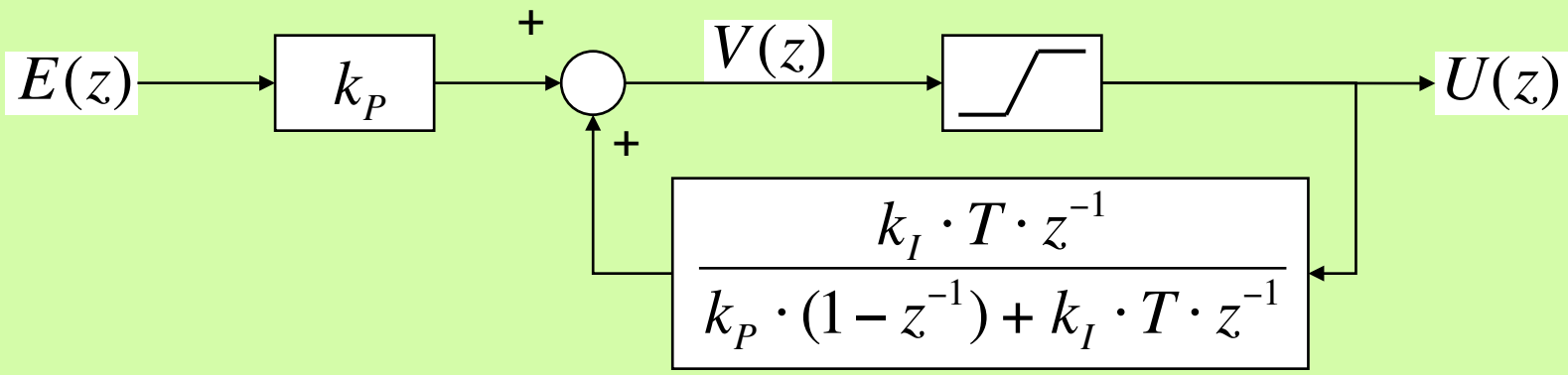
- **This is an Imitation of an Analog PI with Wind-up Correction**
- **Realizable Reference (Can make from R, L, C, and Ideal Transformers)**
- **NOTE: We Can Do Things in Digital that We Cannot in Analog**

PI State Clamping Differs from PI with Analog Integrator Wind-up Correction

Inputs to Nonlinear Clamp Block are Not the Same

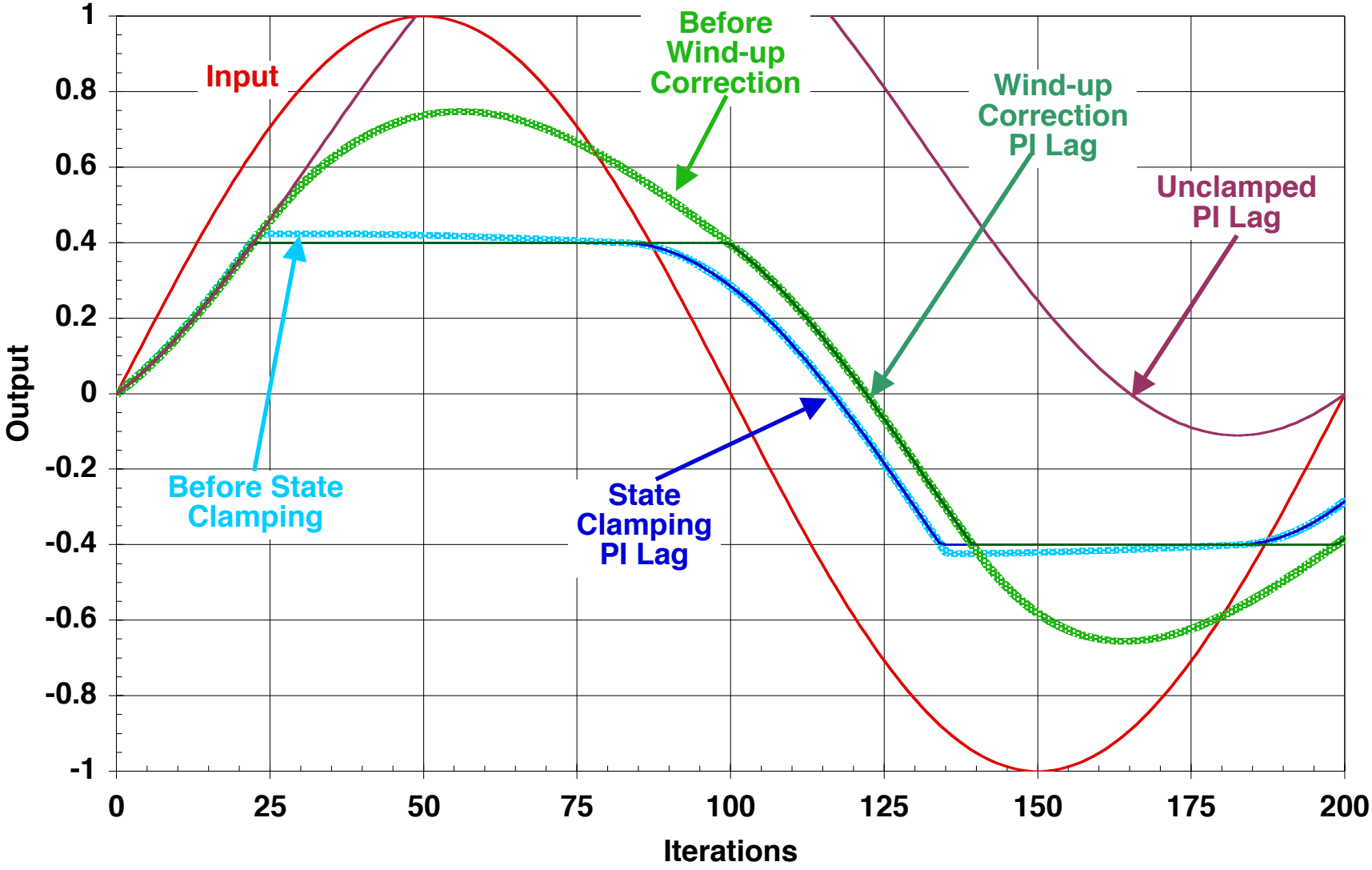


$$V(z) = \left[k_P \cdot (1 - z^{-1}) + k_I \cdot T \cdot z^{-1}\right] \cdot E(z) + z^{-1} \cdot U(z)$$



$$V(z) = k_P \cdot E(z) + \frac{k_I \cdot T \cdot z^{-1}}{k_P \cdot (1 - z^{-1}) + k_I \cdot T \cdot z^{-1}} U(z)$$

Comparison Shows That State Clamping Has Less Lag



Cascading Digital Control Blocks Is Easy

- **Control Block 1 Difference Equation**

$$\psi_n = \alpha_1 \cdot \psi_{n-1} + \beta_0 \cdot \chi_n + \beta_1 \cdot \chi_{n-1}$$

- **Control Block 2 Difference Equation**

$$y_n = a_1 \cdot y_{n-1} + b_0 \cdot x_n + b_1 \cdot x_{n-1}$$

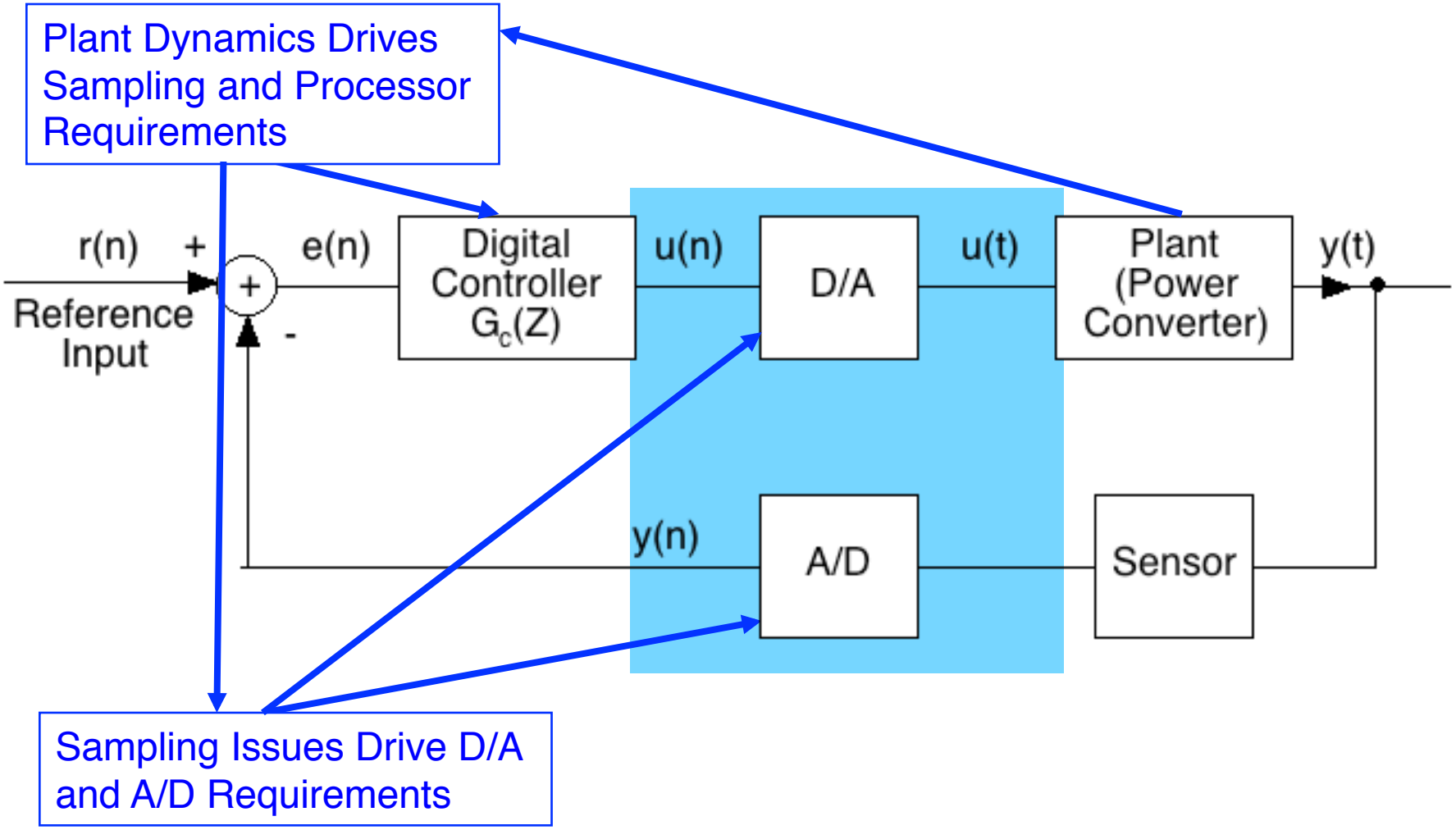
- **Connect Output of 1 to Input of 2**

$$x_n = \psi_n$$

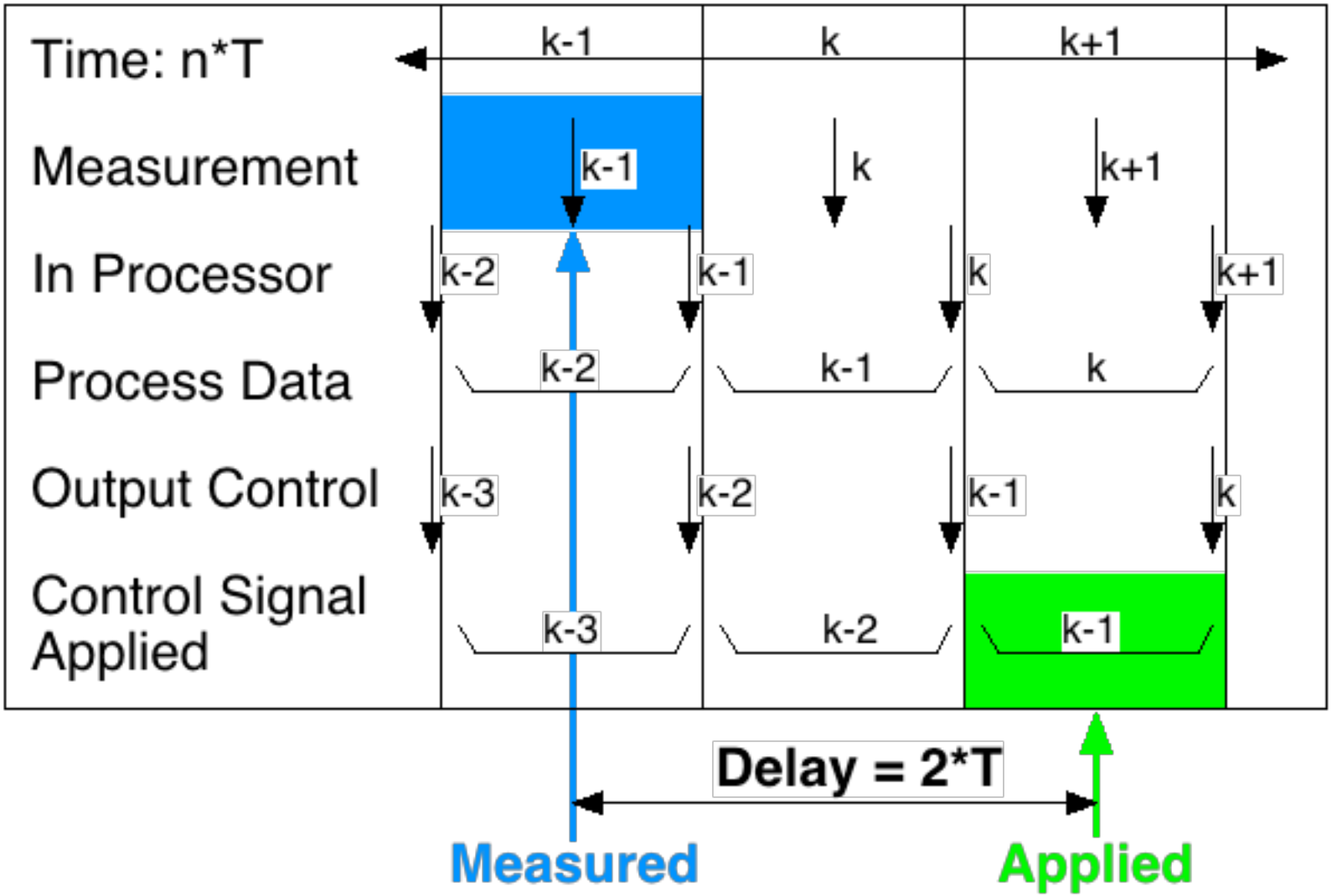
- **Resulting in:**

$$y_n = a_1 \cdot y_{n-1} + b_0 \cdot \psi_n + b_1 \cdot \psi_{n-1}$$

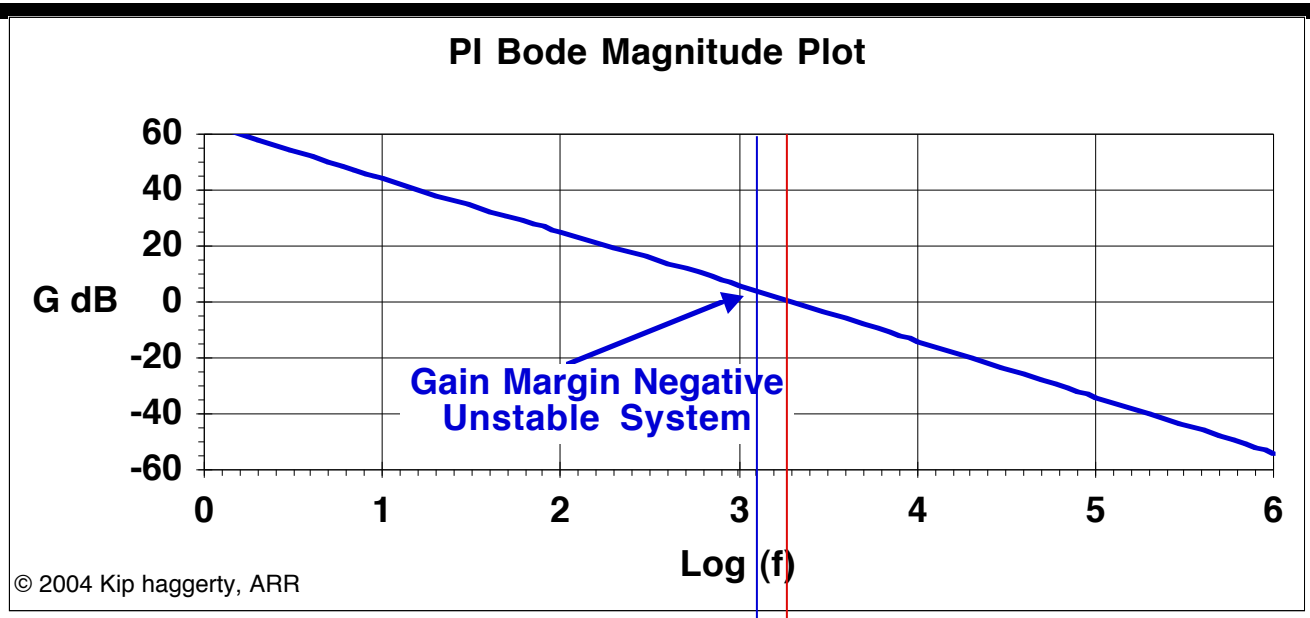
Digital Control Requirements Set Processor and Sampling Requirements

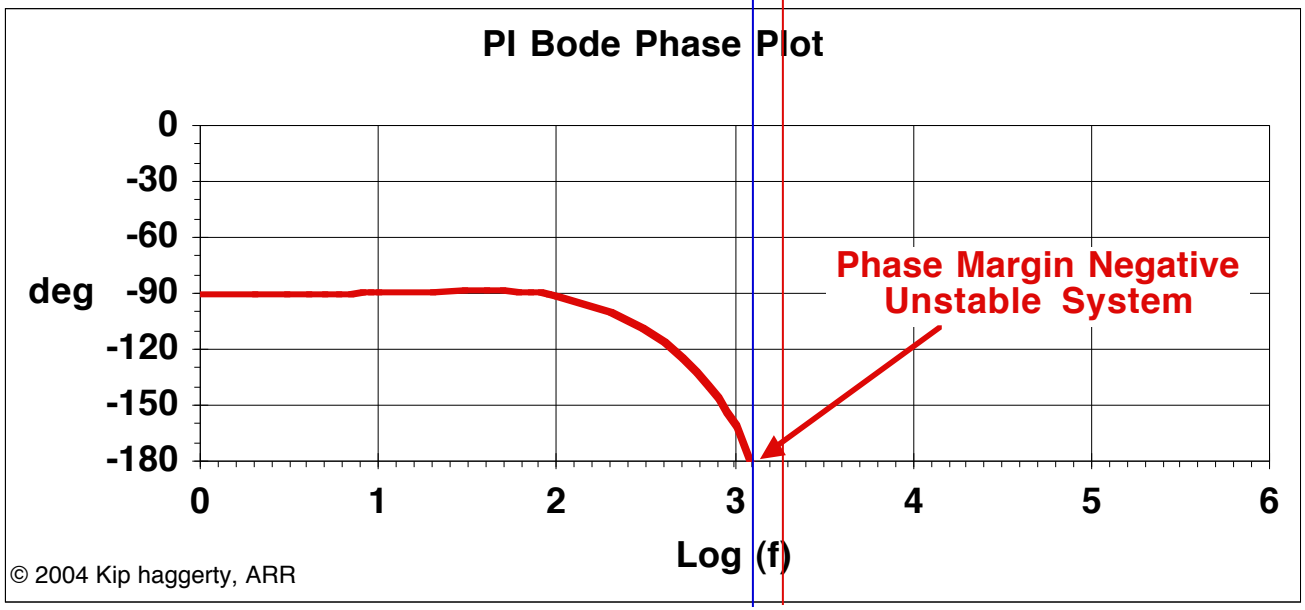


Typical Digital Control Time Line Introduces Pure Time Delay of 2 Processing Intervals

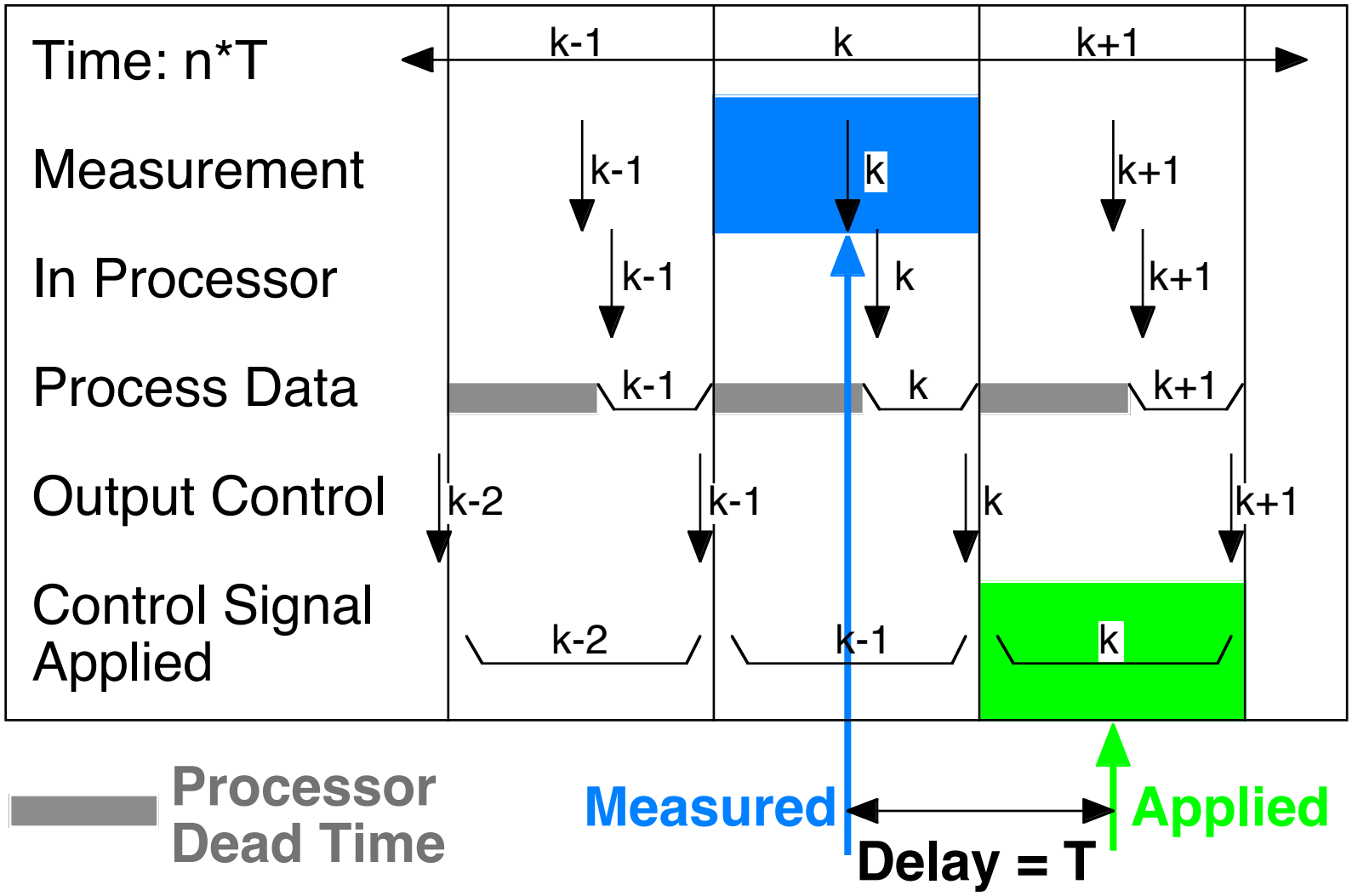


Delay of 2 Processing Intervals Makes Example System Unstable

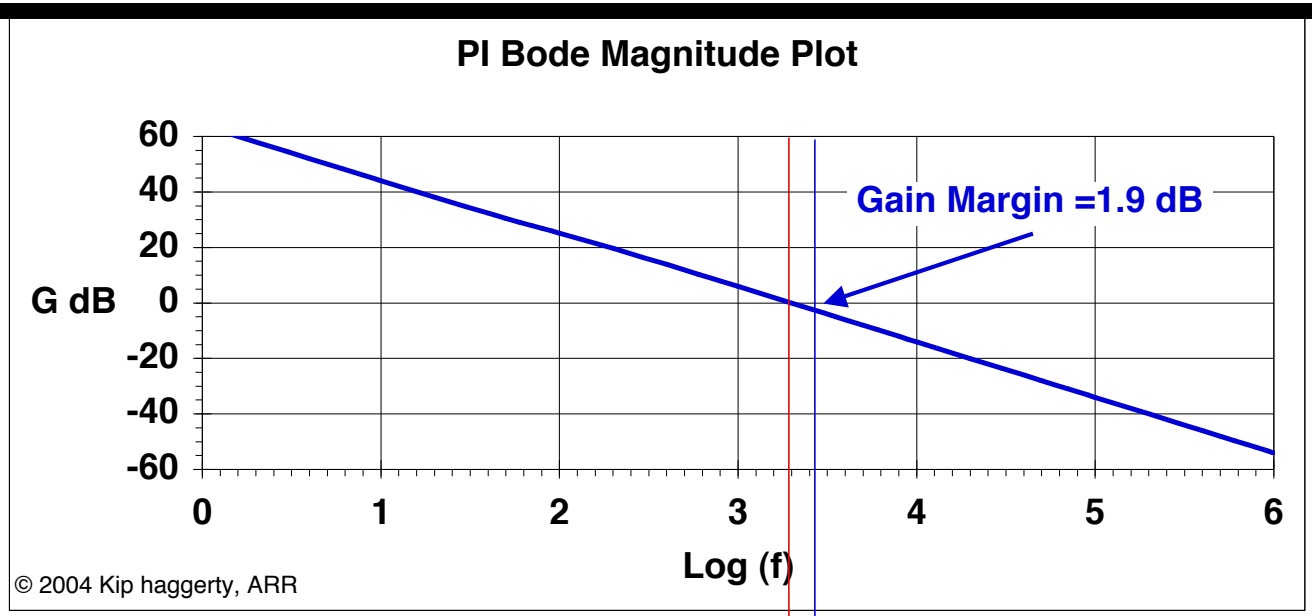


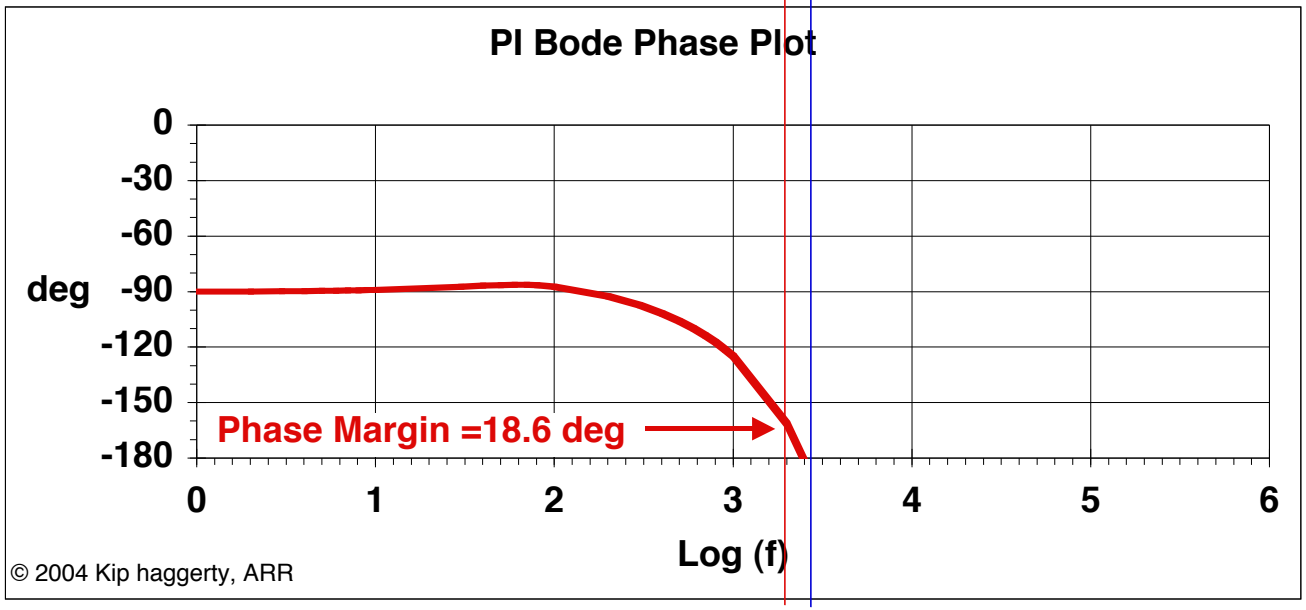


Alternative Digital Control Time Line Introduces Pure Time Delay of 1 Processing Intervals



Reducing the Delay Time to 1 Processing Interval Makes Example Control System Stable





Sampling Issues

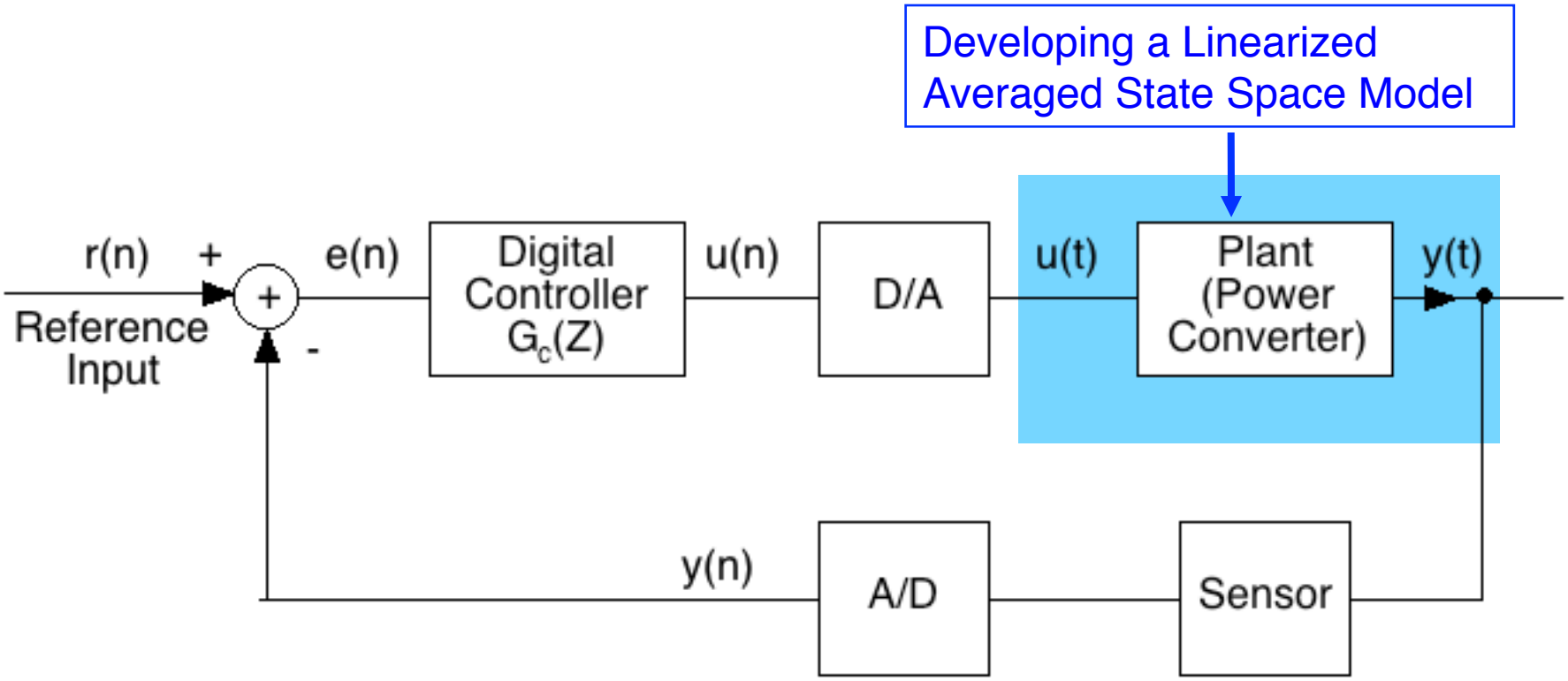
- **Sampling Principle: Sample More than Twice the Highest Frequency of Interest**
  - **Usually 3 - 10 Times Faster**
- **A/D Quantization Error**

$$\sigma^2 = \frac{[LSB]^2}{12} \quad or \quad \sigma = \frac{LSB}{\sqrt{12}}$$

- **D/A Resolution Expansion**
  - **Keep Track of Error Between Desired Value and Output Quantized Value**
  - **Use in Calculating New Desired Value**
  - **Provides Average Resolution Approximating Processor Resolution**

Modeling Plant Required to Predict Performance

NOTE: We Will Not Address This Today



- **References:**
  - **Daniel M. Mitchell, DC-DC Switching Regulator Analysis, NY, McGraw-Hill, 1988. See Chapter 4.**
  - **[SB] Rudolf P. Severns and Gordon (Ed) Bloom, Modern DC-to-DC Switchmode Power Converter Circuits, NY, Van Nostrand, 1985. See Chapters 2 and 3.**