

Analog Simulation Tools Aid Digital-Control-Circuit Designers

Gain a greater intuitive understanding of system operation by transforming your design into its most fundamental elements for simulation.

Feature Article Published in Electronic Design Magazine December 4, 2003.



FEATURE ARTICLE



**by David Caldwell
Flextek Electronics**

Popular analog tools can be used to analyze common digital controllers to leverage decades of proven knowledge.

Ten benefits of analog simulation for digital circuits:

- 1) Leverage decades of proven analog knowledge
- 2) Enhance intuitive understanding of system dynamics
- 3) Systematic circuit optimization
- 4) Time savings over trial and error iterations
- 5) Design validation through multiple domain operation
- 6) Partition analog and digital circuitry
- 7) Substitute low-cost high-speed op-amps
- 8) Develop valuable education tools
- 9) Share results over multiple platforms
- 10) Free evaluation software adequate for most circuits

Digital vs. Analog Controllers

Digital controllers have many benefits over their analog counterparts, including reduced parts count, greater flexibility, and ease of modification. However, something valuable has been cast aside in the transition from analog to digital, decades of useful knowledge. Countless articles have been written about analyzing analog circuits to optimize bandwidth and stability margin, while an equal number of articles have been written about tuning digital controllers through trial and error approaches that often result in less than optimal performance.

Although analog and digital controllers may appear vastly different, their principles of operation are usually quite similar. Therefore, popular analog tools like SPICE can still be used to benefit common digital PI (Proportional-Integral) controllers through analysis, without spending hours on complex math or a fortune on specialized software. Merging proven analog and digital technologies may achieve the best of both worlds.

Digital Controller Example

The goal of a closed-loop controller is to maintain a desired response despite system changes and disturbances. Typical examples include heater temperature, motor speed, light intensity, and fluid flow. Software driven microcontrollers with mixed-signal peripherals often accomplish these tasks. The PI software algorithm is popular for closed-loop control since it is a direct software adaptation of traditional op-amp circuits.

Flextek Electronics developed the digital controller pictured in Figure 1 for educational purposes. The PIC Microcontroller drives current through an infrared emitter with its Pulse Width Modulator and measures the sensor voltage with its Analog to Digital Converter, as shown in Figure 2.

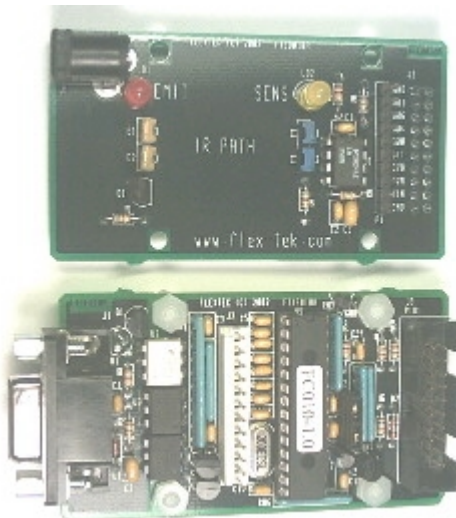


Figure 1. Digital Controller with Infrared and PIC Microcontroller Boards

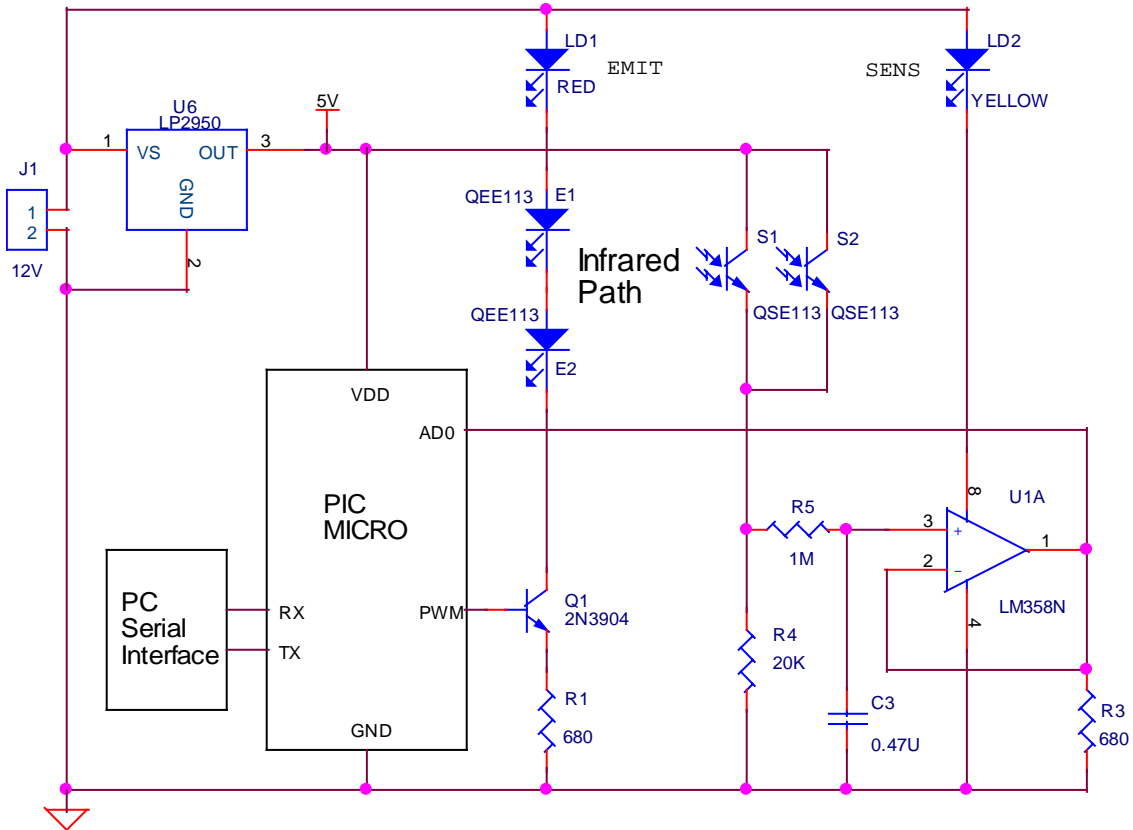


Figure 2. Digital Infrared Controller Simplified Schematic

The gain and time constant of the infrared circuit are set by resistors R4 and R5, which are selected for full ADC swing with timing in human detection range. The PWM drive current illuminates the red LED (EMIT), while the supply current of the op-amp response with burden resistor illuminates the yellow LED (SENS). This circuit reacts to external light or shadowing and has visual feedback to provide an intuitive touch and feel for control experiments.

Figure 3 is the Visual Basic interface to the digital infrared controller that allows different PI gains to be evaluated for a fast stable response with minimal overshoot. This PC screen shot illustrates how easily the system may be adjusted for various applications, which is one of the primary advantages of a digital controller.

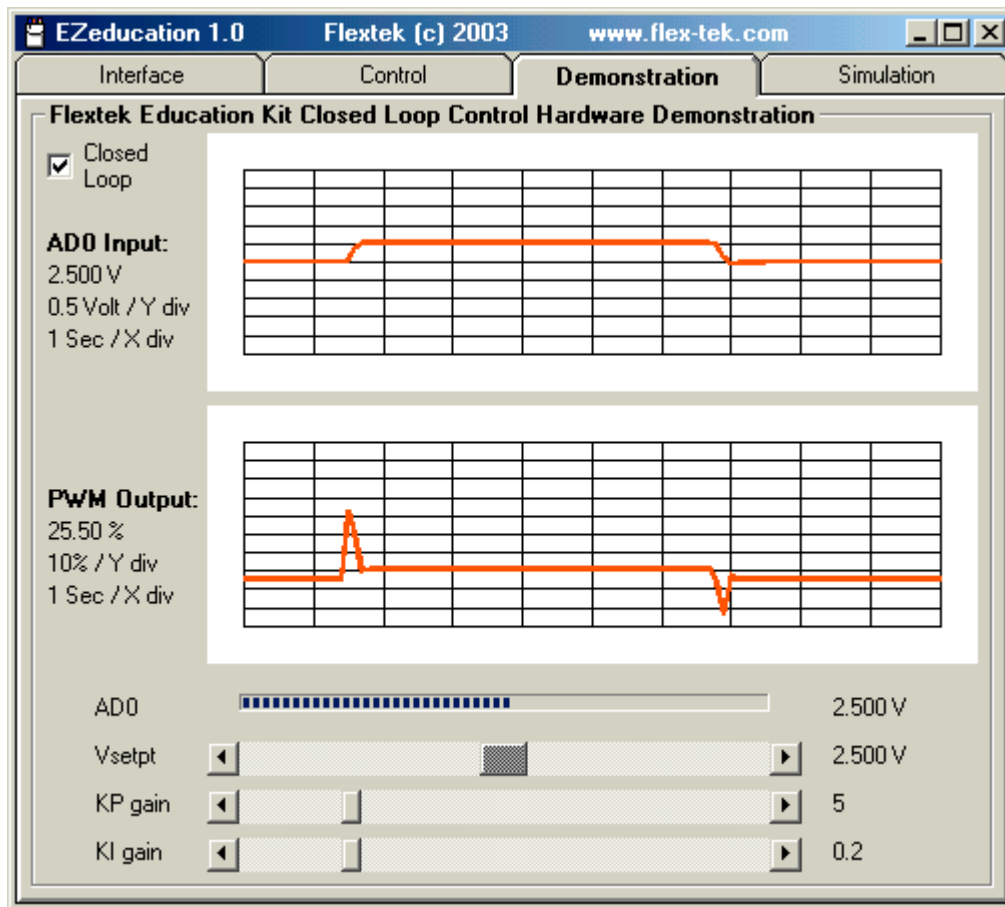


Figure 3. Visual Basic Interface Program to Digital Infrared Controller

The benefit of a properly tuned control loop can be observed by noting the tall spike on the PWM drive signal with absence of spiking on the ADC response. The infrared emitter is initially driven hard to quicken the sensor response, but softens to steady state before the sensor can overshoot its desired value. Too much gain will cause the system to oscillate while too little gain results in a sluggish response. Two independent gains complicate coincident optimization and the third gain in PID make it even more challenging to properly tune. The Differential term is not required for most dominant-pole systems so it is omitted in this demonstration.

Gains are typically adjusted empirically rather than analytically because math in the S and Z domains tends to be cumbersome and mixed-signal control software is often expensive. A few simple approximations, however, enable this digital controller to be simulated with common analog software for systematic optimization of PI gains.

Analog Simulation of Digital Controller

One of the most popular programs for simulating analog circuits is SPICE. Advantages of this program include decades of validated operation with countless application notes and ease of access including free student versions.

Early SPICE versions possessed limitations, including lack of digital simulation, so third-party vendors introduced upgraded versions with increased capabilities. Despite mixed-signal upgrades, there are still advantages to defining systems in terms of simple analog components, including portability between software packages and the intuitive understanding associated with universally defined parts like op-amps and transistors.

Figure 4 shows the analog equivalent circuit of the digital infrared controller executing the PI algorithm. The upper portion of the schematic is the infrared circuit from the PWM to ADC. Voltage gain is dependent source EIR is derived empirically by inspection of Figure 3 data. A 25% PWM duty cycle drive yields a 2.5V ADC response that is 50% of its 5V range, for an effective gain of two. The buffered RC filter dominates the response time of the infrared circuit.

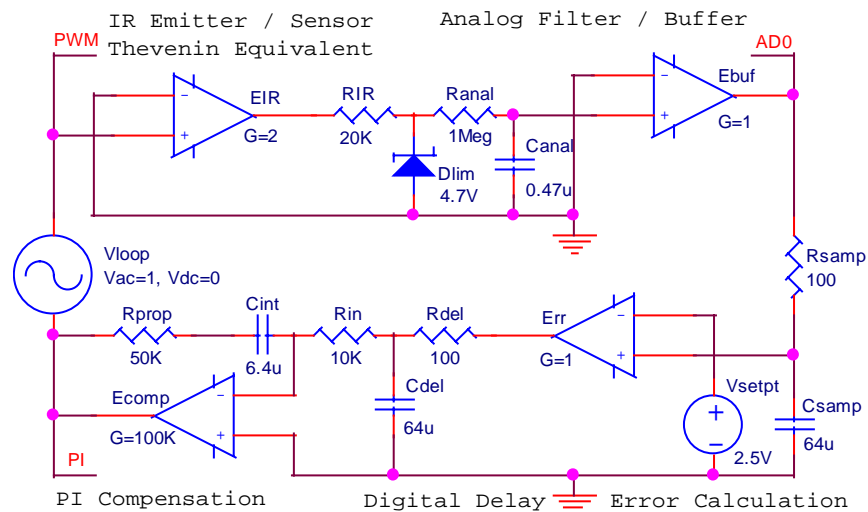


Figure 4. SPICE Analog Equivalent of Digital Infrared Controller

The unique aspect of this conversion effort is to replicate operation of the PI software algorithm, which is not that difficult since PI is a direct adaptation of an op-amp circuit. However, one challenging aspect of this effort is to simulate sampled discrete operation of a digital system with continuous analog circuits, so reasonable approximations are employed for simplicity.

The PI algorithm is executed through the PC in Visual Basic for development ease, which results in an average sample interval of 12.8mS for serial data transfer and response to interrupts. This sample time is simulated with a 6.4mS RC network that is a reasonable approximation based on Nyquist criteria. This delay network is repeated since the microcontroller contributes an additional sample cycle delay due to its ADC over-sampling and digital filtering for enhanced accuracy and resolution.

The remaining circuitry is the PI closed-loop controller. The error-amp takes the difference between the desired set point and measured response to be compensated for desired dynamics by an op-amp with proportional and integral gains. These amplifiers are configured such that a single inversion occurs throughout the control loop for negative feedback. Control textbooks usually use separate op-amps for each PI gain term, but practical circuits combine them for reduced parts count.

PI code used in infrared digital controller:

$V_{setpt} = 2.5$; $K_P=5$; $K_I = 0.2$	‘ Initialization
$V_{err} = V_{setpt} - AD0$	‘ Measured error
$IntSum = K_I * V_{err} + IntSum$	‘ Integrator running sum
$PI = K_P * V_{err} + IntSum$	‘ PI calculation
$PWM = PI * 100/5$	‘ Convert PWM (100%) from ADC (5V)

Loop every 12.8mS data sample update.

Analog conversion of PI code:

$R_{prop} = K_P * R_{in} = 5 * 10K = 50K\Omega$
 $C_{int} = T_{samp} / (K_I * R_{in}) = 12.8mS / (0.2 * 10K) = 6.4\mu F$
 $C_{samp} = C_{del} = T_{samp} / (2 * R_{samp}) = 12.8mS / (2 * 100) = 64\mu F$

SPICE circuit component calculated values.

The SPICE circuit also contains two independent voltage sources. V_{setpt} is varied in the time domain for transient response and V_{loop} is used to evaluate stability in the frequency domain. These sources serve as tools to quantify behavior of the circuit.

Loop gain is the ratio of AC response (PI in Figure 4) to AC stimulus (PWM in Figure 4) through the feedback loop. A system is stable with less than 180° lag at unity gain or 0dB. Phase margin indicates the additional lag before instability is reached. The loop starts at 180° in a negative feedback system so 180° lag actually occurs at 360° or 0°.

Figure 5 is the loop gain of the simulated infrared controller, which shows 3.25Hz bandwidth and 72.3° phase margin. These values were calculated automatically by the post-processor in the evaluation version of Intusoft SPICE.

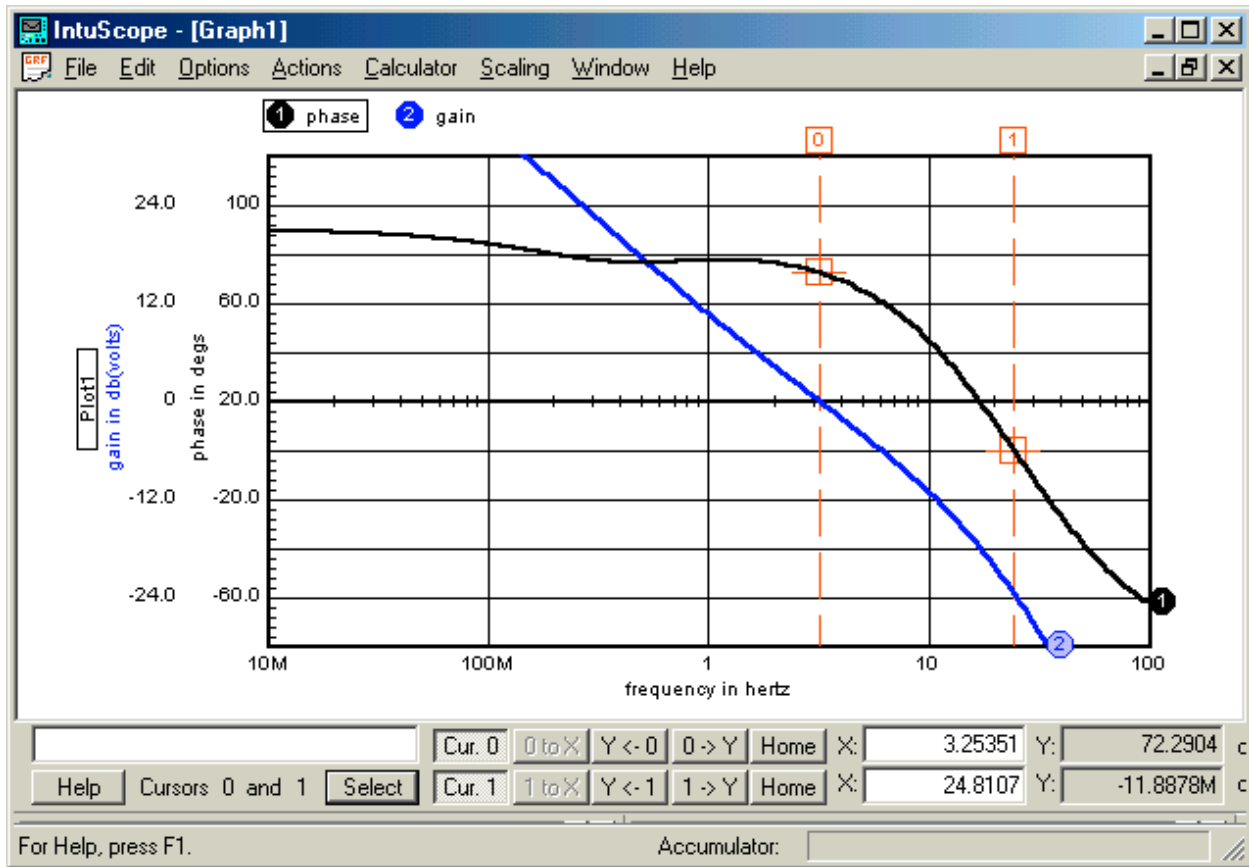


Figure 5. SPICE Plot of Loop Gain indicates 3.25Hz Bandwidth and 72.3° Phase Margin.

While the hardware step response in Figure 3 gives a rough indication of loop bandwidth and phase margin, it does not pave the way to optimization as well as the loop gain plot in Figure 5. This graphic demonstrates that bandwidth is ultimately limited by sample frequency, and that proportional correction must exceed integral correction at the unity gain frequency. This is because the infrared circuit introduces 90° lag and the integrator adds another 90° lag, so the pole of the integrator must be compensated for reasonable phase margin. These types of valuable observations are easier to reach after working with analytical tools than trial and error methods.

Figure 6 is the transient step response of the simulated infrared controller, which matches the hardware measurement in Figure 3 very well. Gains and time constants of the infrared hardware and software compensation were varied, and multiple simulations were successfully compared to hardware results to validate the proposed techniques.

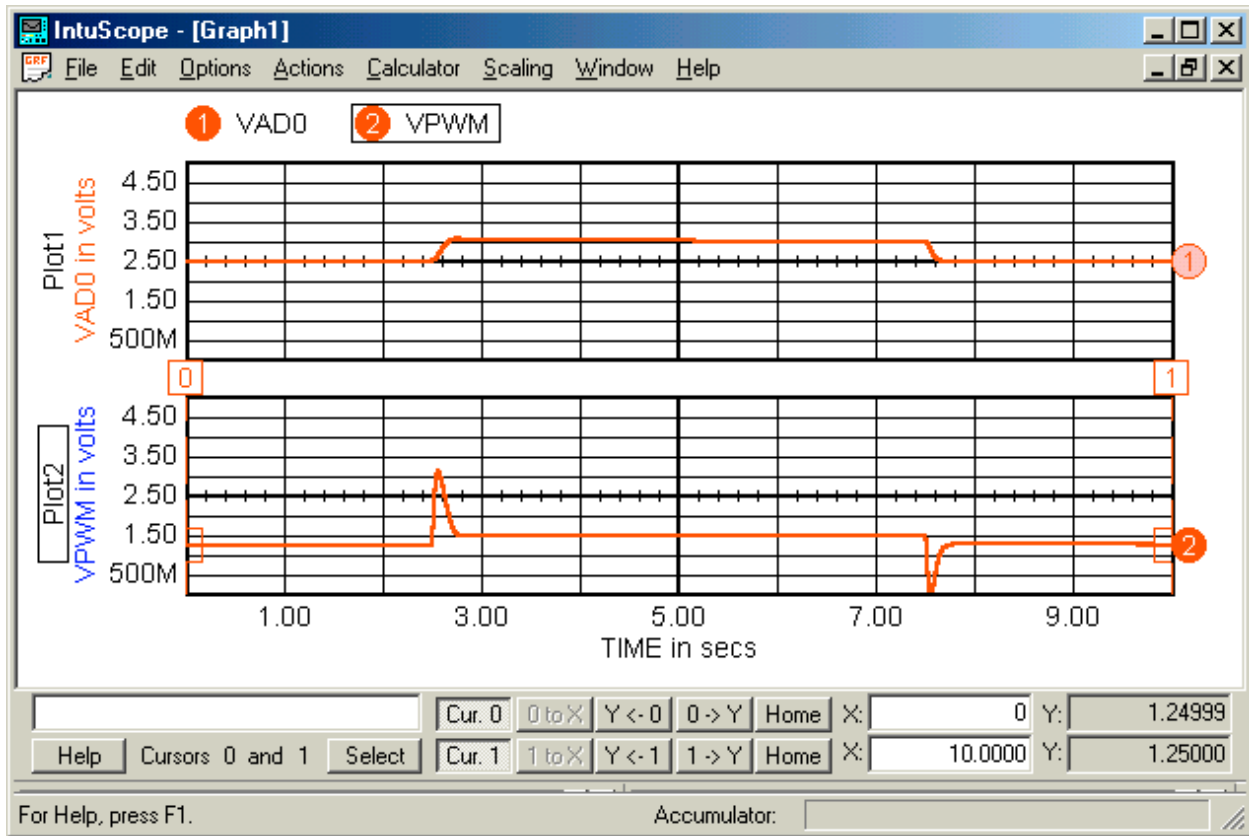


Figure 6. SPICE Plot of Transient Step Response matches Hardware Measurement.

The advantages of transforming a system into its most fundamental elements for simulation are many. Greater intuitive understanding is achieved through both the transformation and simulation processes. Many people using varied software can share results, making more tools and application assistance available. Errors are less likely to go undetected when calculations are performed by multiple methods and compared. Better optimization with respect to performance, robustness, and parts selection is enabled with this enhanced understanding and availability of tools. Most importantly, decades of useful knowledge are put to work to benefit new and emerging technologies, which is essential for continued growth.