Applying Digital Technology to PWM Control-Loop Designs

Mark Hagen and Vahid Yousefzadeh

Abstract

This topic discusses the application of digital-control to DC/DC-switching converters and how to model the digitally controlled system. The main blocks that appear in almost every digital controller—the error ADC, the compensator, and the digital PWM engine—are discussed and used to model small-signal characteristics such as frequency response, stability criteria, the effects of quantization, as well as the impact of sampling rate and delay introduced by the digital controller to the system. This model is extended to include nonlinear gain and its benefits. Finally, a graphical user interface is introduced and demonstrated for use with the design of a two-phase synchronous-buck converter.

I. Introduction

Switch-mode power-supply (SMPS) converters find use in a wide variety of applications, ranging from a fraction of a milliwatt in on-chip power management to hundreds of megawatts in power systems. All of these applications require efficient and cost-effective static and dynamic power regulation over a wide range of operating conditions. An analog or digital controller closes the feedback loop around the switching converter and actively controls the on/off states of the power-semiconductor devices to achieve input or output regulation. Fig. 1 shows a typical analog controller that uses analog feedback to provide output voltage regulation.

Over the past few decades, digital controllers in the form of digital-signal processors (DSPs), microcontrollers, and field-programmable gate arrays (FPGAs) have seen extensive application in motor-drive controllers and high-voltage and high-current power electronics. In these applications the control algorithms are generally sophisticated, while the semiconductor devices operate at relatively low switching frequencies, e.g. at tens of kilohertz.

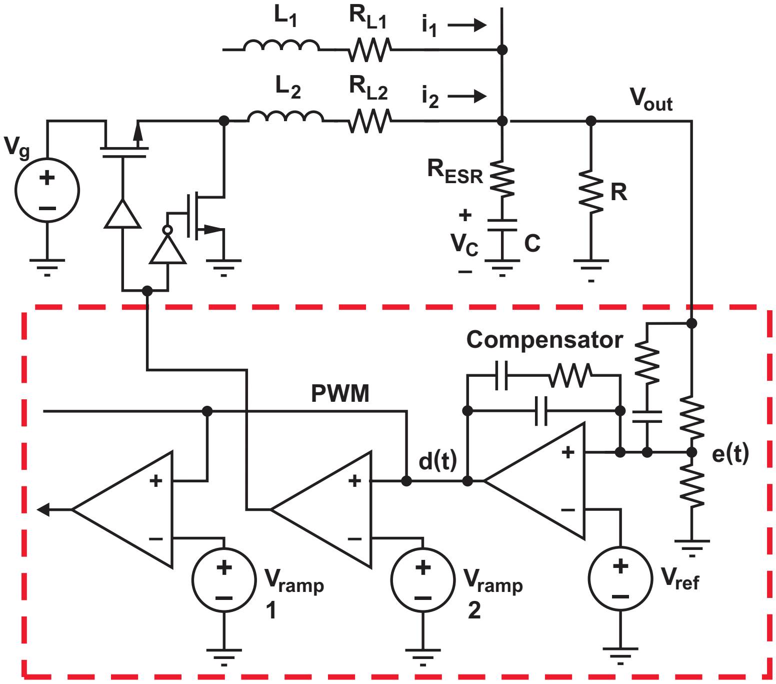


Fig. 1. Analog-controlled SMPS.

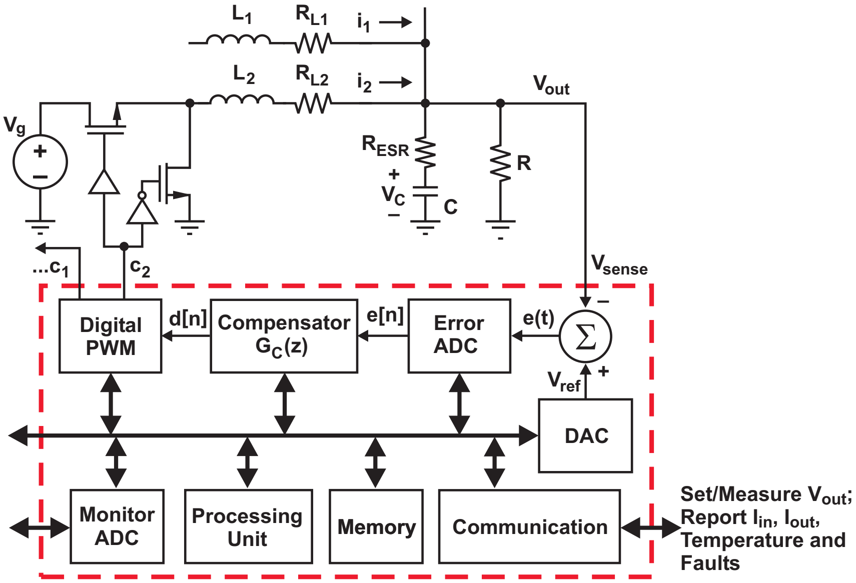


Fig. 2. Digital controller in an SMPS.

Continued rapid advances in CMOS and VLSI technology have enabled the development of a high-performance, practical, cost-effective, and low-power digital SMPS controller. Fig. 2 shows a block diagram of an advanced digital controller that closes the feedback around a SMPS. Such a controller, because it is implemented in a digital silicon technology, usually includes a standard communication block; general-purpose ADCs (ADCs); digital I/Os; memory; and a processing unit (microcontroller) that handles programming, communication, diagnostics, power management, etc. The result is that a digital controller not only regulates the output voltage, but also can perform complex sequencing and can monitor key parameters like average current and power for the host system.

This topic focuses on the use of digital technology to implement a SMPS controller. Specific examples are for a non-isolated point-of-load (POL) application. We first review the techniques necessary to model the discrete time controller. Then the new features and functions that digital control enables are discussed. There are three specific blocks that enable the digital controller to achieve the high-performance regulation requirements of an SMPS: the ADC used to sample the error voltage (and an associated setpoint reference DAC), the digital filter that compensates the error signal, and the digital pulse-width modulator (DPWM) that converts the sampled, compensated error signal into the gate-drive signals.

Because most digital controllers contain a serial interface, they can be easily configured from design software. This allows the design software to do the "heavy lifting" in terms of modeling the system and calculating appropriate compensation for the SMPS. In this topic we discuss the what goes on "under the covers" of the design software.

II. Modeling a Digital Controller

Switch-mode power supplies have always had a digital component; they have a control effort with a discrete update interval. That interval is the switching period. The net result is that there can be a latency in the response to disturbance in the control effort. When we analyze a SMPS system, this latency shows up as a rotation in the phase of the open-loop system. When we introduce digital components into the system, there are additional phenomena that must be taken into account. These things are:

1. Feedback quantization
2. Control effort quantization
3. Delay needed to sample the feedback and calculate the control effort

The key to implementing a digitally-controlled power supply is understanding these effects.

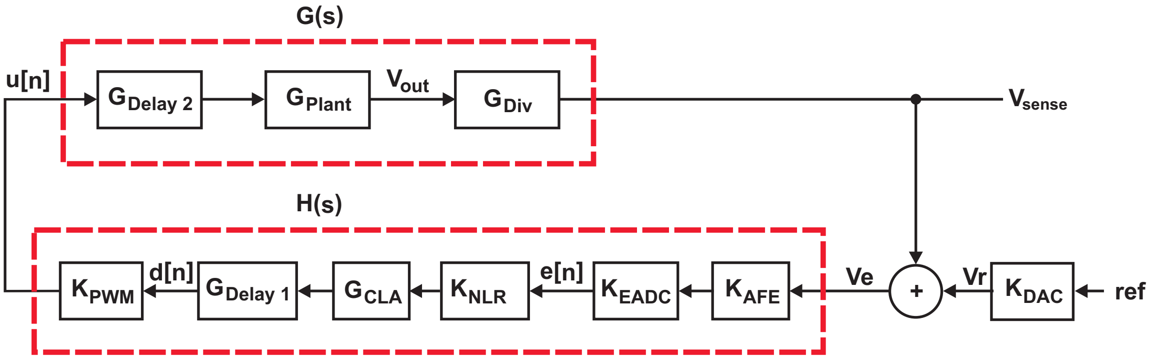


Fig. 3. Closed-loop block diagram.

Fig. 3 shows the closed-loop block diagram for a digitally controlled SMPS that first generates an analog error voltage and then digitizes that voltage to calculate the PWM control effort. For this system the total open-loop gain is

$$T(s) = G(s) \times H(s) \qquad (1)$$

Then the closed-loop gain, from the PWM control effort, u, to the sensed output voltage, v_{sense}, is

$$\frac{v_{sense}}{u} = \frac{G(s)}{1 + G(s) \times H(s)} \qquad (2)$$

The contributors to the closed-loop system are itemized in Table 1. To determine the frequency response of the power supply, and from that, determine the stability margin of the system, we need to define the dynamic gain for each block. Once we have the transfer function for each block, the standard measure of stability can be applied:

- **Gain Margin**—The inverse of the magnitude of the open-loop gain, expressed in dB, at the frequency where the phase of the open-loop gain is 180 degrees.
- **Phase Margin**—The phase of the open-loop gain, expressed in degrees, where the magnitude of the open-loop gain is 1.0 (0 dB).

In addition to developing a frequency-domain model of the system, it is important to develop a time-domain model of the digitally-controlled power supply so that the effects of quantization can be observed. In the following sections we will work our way around the feedback loop and develop the necessary description of each functional block so that both a frequency-domain model and a time-domain model can be created.

Table 1. Closed-Loop System Contributors

K_{AFE}	Analog front-end gain in V/V
K_{EADC}	Error ADC gain in LSB/volt
K_{NLR}	Nonlinear boost gain
G_{CLA}	Control-law accelerator (digital compensator) gain
G_{Delay1}	Total sampling and CLA computational delay
K_{PWM}	PWM gain in duty/LSB
G_{Delay2}	On-time and any delay to multiple power stages driving V_{out}
G_{Plant*}	Transfer function from the time location of the falling edge of the PWM signal to V_{out} of the power stage
G_{Div}	Divider network transfer function in V/V

*The frequency response of the plant is derived from the average model of the power stage(s).

III. Power-Stage Modeling

The development of the frequency response of the plant is identical for analog or digitally-controlled power supplies. It is derived for the average model of the power stage(s). For a buck regulator, such as used in the POL power supply, the continuous-mode, small-signal-transfer function is simply

$$G_{plant}(s) = \frac{v_{out}}{duty} = V_{in} \times G_{LC}(s), \qquad (3)$$

where $G_{LC}(s)$ is the transfer function of the LC low-pass filter and load resistance of the power stage.

There are several reasons that the derived frequency response of the average model may be insufficient when designing a digitally-controlled power supply. Digital control introduces quantization of the error voltage and quantization of the output-PWM control effort. There also are additional delays in a digital system such as the time it takes to convert the error voltage to a numerical value, and the time it takes the digital filter to calculate the control effort. Finally, processing the error signal digitally enables non-linear gains to be applied to the signal. All of these effects are best observed in a time model of the system.

A time-domain model of the system should describe how variations in input voltage, load current, and duty cycle affect the output voltage. Since the power stage of a switching converter forms a nonlinear system, designing a linear controller usually involves linearizing this power stage. The traditional approach uses average modeling to provide a dynamic model for PWM-operated DC/DC converters [1]. The frequency range of interest is much smaller than the switching frequency, so the model ignores the switching frequency and its harmonics. The small-signal linear model of the power stage is easily derived from the average model.

Although a linearized model based on switch averaging provides accurate results at low frequencies, it starts to break down beyond about 1/5 of the switching frequency, f_{sw}. The phase response exhibits large discrepancies between the model and the measured behavior. This modeling error becomes particularly important in the design of high-bandwidth switching converters.

Reference [2] discusses discrete-time modeling of DC/DC switching converters that take into account the sampling effect of the PWM for analog controllers. Recently, the growing focus upon digital control for high-frequency DC/DC converters has revived interest in discrete-time analysis and modeling [3]. Discrete-time modeling easily incorporates the propagation delay of discrete components, computational delay, and DPWM delay into a dynamic model [4, 5]. The following sections discuss the techniques for switch averaging and discrete-time modeling in further detail.

A. Review of the Average Model

Switch averaging removes the switching ripples in the inductor-current and capacitor-voltage waveforms over the switching period. The following equations define the low-frequency components of the inductor and capacitor waveforms:

$$L\frac{d\langle i_L(t)\rangle_{T_s}}{dt} = \langle v_L(t)\rangle_{T_s}, \text{ and}$$
$$C\frac{d\langle v_C(t)\rangle_{T_s}}{dt} = \langle i_C(t)\rangle_{T_s}, \tag{4}$$

where

$$\langle x(t)\rangle_{T_s}$$

denotes the average of x(t) over an interval of one switching period, T_S:

$$\langle x(t)\rangle_{T_s} = \int_t^{t+T_s} x(\tau)d\tau. \tag{5}$$

Although averaging removes the high-frequency switching ripple, the average value still varies from one switching period to the next so as to model low-frequency variations. The differential equations for capacitor voltages and inductor currents are derived for each switching interval (for example, the on time and the off time of the switch in a simple buck converter). Averaging the differential equations for each switching interval provides the desired average equations.

The next step in obtaining the small-signal model of the power stage involves linearizing the average equations. The capacitor voltage and inductor currents are generally nonlinear functions of the signals in the converter (input and output voltages, duty cycle, etc.). To obtain the small-signal behavior of the power stage, perturb the signals and the average values of capacitor voltages and inductor currents around their steady-state operating point:

$$u(t) = U + \hat{u}, \tag{6}$$

where $\hat{u} << U$. Next, replace the signals, the capacitor voltages, and the inductor currents with their perturbed values from Equation (6). Eliminating the DC components, U, and the high-frequency components, $\hat{u}_1\hat{u}_2$, yields the differential

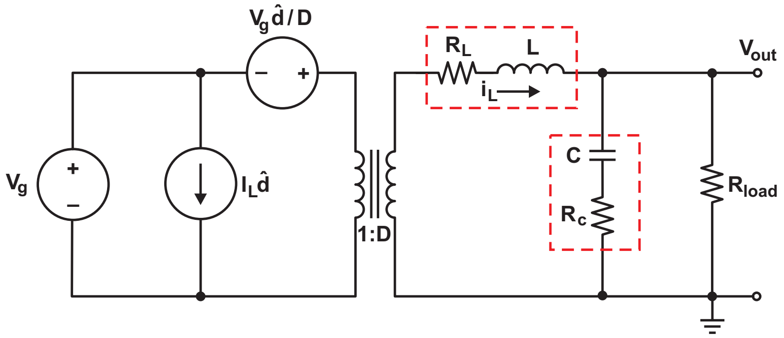


Fig. 4. Small-signal model of a buck converter with losses.

equations for the small-signal values of û. Fig. 4 shows the small-signal model of a buck converter with inductor parasitic resistance, R_L, and output-capacitor equivalent-series resistance, R_{esr}. D indicates the steady-state duty cycle, and $D' = 1 – D$.

The model of Fig. 4 provides the basis for computing control-to-output-voltage (d-to-v_{out}) and input-voltage-to-output-voltage (v_g-to-v_{out}) transfer functions.

B. Average Model with Delay

A digital controller with an eADC sampling once per switching period, generates a delay associated with the sampling point within each switching interval. The average model presented in the previous section does not include this delay. Therefore, the phase response differs significantly from the actual small-signal transfer function. The average model can be augmented with the known delay, t_d, to produce a more accurate model:

$$G_{vd-del}(s) = \frac{\hat{v}_{out}}{\hat{d}} = e^{t_d s} G_{vd}(s), \tag{7}$$

where $G_{vd}(s)$ is the control-to-output-voltage transfer function found with average modeling and t_d is the delay. From the point of view of the power stage, the total delay is defined as the time from when the output voltage is sampled to the falling edge of the controlled MOSFET switching node (for a trailing-edge modulation system).

$$t_d = t_{falling\,edge} - t_{sample} \tag{8}$$

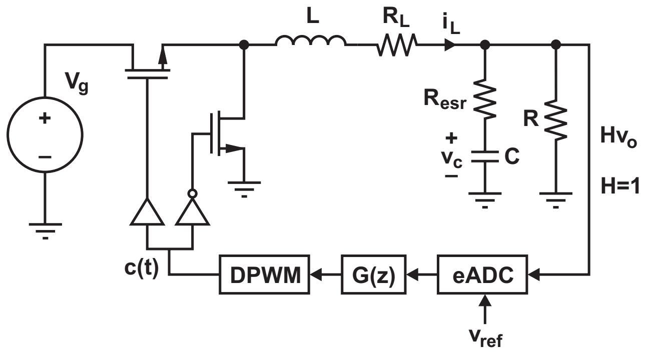


Fig. 5. Voltage-mode, closed-loop buck converter.

C. Discrete-Time Model

Because averaging eliminates the high-frequency components of the frequency response, it cannot predict sampling effects like period doubling that can occur in peak-current control mode. Discrete-time modeling [4, 5] takes into account PWM sampling. Also, an average model dues not easily display the effects of quantization and non-linear gain.

Fig. 5 shows a switching buck converter operating in closed-loop, digital-voltage-mode control. Fig. 6 shows waveforms illustrating the derivation of a discrete-time model that approximates the perturbation of the control-signal duty cycle, d, as a train of impulses. The perturbation of the duty cycle occurs at the modulating edge of the PWM signal. The duty-cycle perturbation changes the state variables of the switching converter (capacitor voltages and inductor currents) at the modulating point and propagates through the switching period. Discrete-time modeling obtains the capacitor-voltage and inductor-current values at the eADC sample points as a function of previous capacitor and inductor samples; the control signal, d; and the input

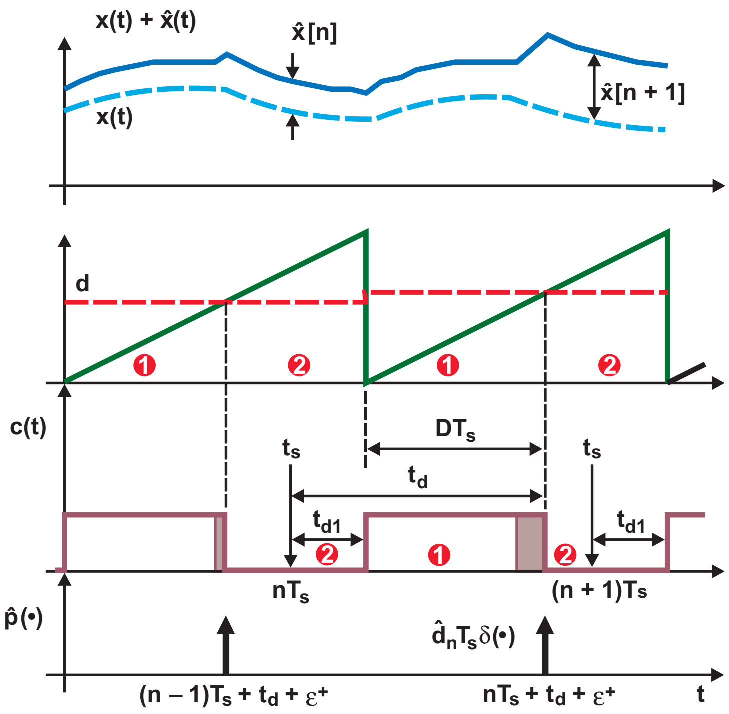


Fig. 6. Waveforms illustrating derivation of a discrete-time model for a digitally-controlled SMPS.

voltage, V_g. The total delay, t_d, shown in Fig. 6 equals the time between eADC sampling and the modulation edge of the PWM signal.

To derive the discrete-time differential equations for the capacitor voltage and inductor current, the state matrices for the power stage at each switching interval must first be obtained:

$$\begin{aligned} &\dot{x} = A_q x + B_q V_g \text{ and} \\ &v_{out} = C_q x + D_q V_g, \end{aligned} \qquad (9)$$

where q = 1 denotes on time, and q = 2 denotes off time.

For the switching buck converter with parasitic parameters shown in Fig. 5, the state matrices and variables become:

$$A = A_1 = A_2 = \begin{bmatrix} \dfrac{-1}{(R+R_{esr})C} & \dfrac{R}{(R+R_{esr})C} \\ \dfrac{-R}{(R+R_{esr})L} & \dfrac{-(R_L + R \| R_{esr})}{L} \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 0 \\ \dfrac{1}{L} \end{bmatrix}, B_2 = 0,\ D_1 = D_2 = 0, \text{ and} \qquad (10)$$

$$C = C_1 = C_2 = \begin{bmatrix} \dfrac{R}{R+R_{esr}} & R \| R_{esr} \end{bmatrix}, x = \begin{bmatrix} v_C \\ i_L \end{bmatrix}.$$

The z-domain control-to-output-voltage transfer function can be determined from the small-signal discrete-time equations

$$\begin{aligned} &\hat{x}[n] = \Phi \hat{x}[n-1] + \Gamma \hat{d}[n-1] \text{ and} \\ &\hat{v}_{out}[n] = C \hat{x}[n]. \end{aligned} \qquad (11)$$

TABLE 2. THE DISCRETE-TIME-STATE MATRICES

	Φ	Γ
Trailing-Edge DPWM		
$0 \le t < DT_s$, on-time sampling	$e^{A_1(DT_s - t)} e^{A_2 D'T_s} e^{A_1 t}$	$e^{A_1(DT_s - t)} e^{A_2 D'T_s} \alpha T_s$
$DT_s \le t < T_s$, off-time sampling	$e^{A_2(T_s - t)} e^{A_1 DT_s} e^{A_2(t - DT_s)}$	$e^{A_2(T_s - t)} \alpha T_s$
Leading-Edge DPWM		
$0 \le t < D'T_s$, off-time sampling	$e^{A_2(D'T_s - t)} e^{A_1 DT_s} e^{A_2 t}$	$e^{A_2(D'T_s - t)} e^{A_1 DT_s} \alpha T_s$
$D'T_s \le t < T_s$, on-time sampling	$e^{A_1(T_s - t)} e^{A_2 D'T_s} e^{A_1(t - D'T_s)}$	$e^{A_1(T_s - t)} \alpha T_s$

Table 2 shows the resulting discrete-time-model matrices Φ and Γ for the trailing- and leading-edge DPWM and the eADC sampling instants during switch on and off time.

In Table 2, $\alpha = (A_1 - A_2)X_p + (B_1 - B_2)V_g$, where $X_p = [V_C \; I_L]^T$ represents the steady-state capacitor voltage and inductor current. The small-signal z-domain control-to-output-voltage transfer function of the power stage is

$$G_{vd}(z) = \frac{\hat{v}_{out}}{\hat{d}} = \frac{C(I - z^{-1}\Phi)^{-1}\Gamma}{z}. \qquad (12)$$

A software program such as MATLAB® or Mathcad® can easily solve this equation. Alternatively, one can further simplify the equation by reducing the exponential matrices using the approximation $e^{AT_s} \approx I + AT_s$. The simplified results for the buck switching converter are shown in Reference [4].

Fig. 7 compares the control-to-output-voltage magnitude and phase response of a buck converter modeled with the approaches described earlier in this section. The parameters for this converter are: L = 1 µH; C = 4 x 200 µF; R_{esr} = 2 mΩ; R_L = 8.6 mΩ; V_g = 10 V; V_{out} = 1 V; R = 10 Ω; f_{sw} = 500 kHz; and $t_d = 0.25T_s + DT_s$, where D ≈ 0.1 and T_s = 2 µs. This shows that the average model including the total delay, t_d, closely approximates the accurate frequency response of the discrete-time model.

The Texas Instruments *Fusion Digital Power Designer* PC program provides both an average model with delay and a discrete-time model of the target system to assist in configuring the design of digital-power applications.

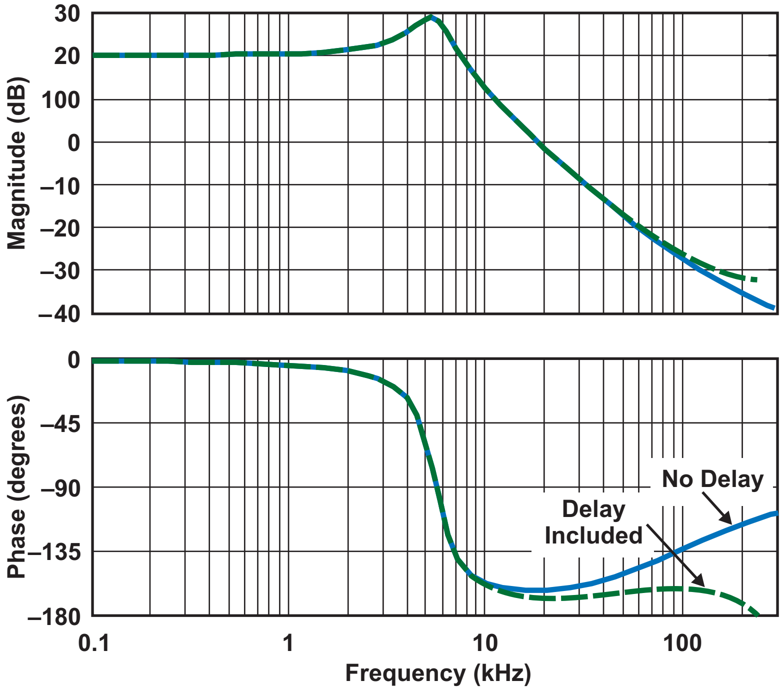


Fig. 7. The control-to-output-voltage magnitude and phase response of a buck converter.

MATLAB is a registered trademark of The MathWorks, Inc. Mathcad is a registered trademark of Parametric Technology Corporation.

IV. Divider Network Model

To estimate the stability margin from the average model or examine the cycle-by-cycle behavior with the discrete-time model, we need to describe each circuit around the feedback loop. The next circuit after the power stage is the voltage-divider network. This circuit produces a scaled sense voltage for the controller. It also can provide a compensation zero and a pole that acts as an anti-alias filter for the digital-controller ADC.

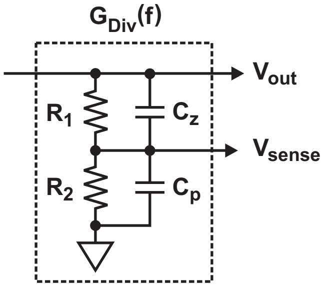


Fig. 8. Divider network.

To model the divider we first write out the transfer function for the circuit in s. Here K_{div} is the DC divider ratio $R_2/(R_1+R_2)$.

$$G_{div}(s) = K_{div}\frac{R_1C_zs+1}{K_{div}R_1\left(C_z+C_p\right)s+1} \tag{13}$$

To calculate the system open-loop gain for a Bode plot, this transfer function is evaluated at s = j2πf and the resulting vector of values multiplied by the complex vector of gain values for the power stage. A compensation zero can be created by including a capacitor in the C_z location. However, this zero will not be programmable like the compensation zeros that are created by the digital filter in the controller, so C_z is typically not populated.

Adding capacitor C_p adds a pole to the overall open-loop gain and can serve the useful purpose of attenuating frequency content in the sensed voltage that has a frequency greater than 1/2 the sample rate of the controller ADC and compensator. Sampling theory states that harmonic frequencies around integer multiples of the sample rate (typically the same as the switching frequency, f_{sw}) will alias to near DC and can cause an error in the regulated voltage. By rolling off the sense voltage for frequencies above the ADC sample rate, we attenuate these errors.

$$C_p = \frac{1}{R_1\|R_2\times 2\pi f}\ \text{(for } C_2 = 0\text{)} \tag{14}$$

To model the divider network in the discrete-time domain, we translate the transfer function from the s domain to the z domain. In this case the bilinear transformation is chosen. The bilinear, or "Tustin" transformation estimates the expression $z = e^{sT}$ as

$$z = e^{sT} \approx \frac{1+sT/2}{1-sT/2} \tag{15}$$

Then we can express s as

$$s = 2f_{sw}\left(\frac{z-1}{z+1}\right) \tag{16}$$

Substituting Equation (16) into (13) yields a transfer function in *z*. From the z-domain transfer function, we can write the discrete difference equations shown in Equation (17) below.

This is the technique used in the TI *Fusion Digital Power Designer* program.

$$v_{sense}[n] = -\frac{1-\tau_{pole}}{1+\tau_{pole}}v_{sense}[n-1] + K_{Div}\frac{1+\tau_{zero}}{1+\tau_{pole}}v_{sense}[n] + K_{Div}\frac{1-\tau_{zero}}{1+\tau_{pole}}v_{sense}[n-1], \tag{17}$$

where

$$\begin{aligned} \tau_{zero} &= R_1C_z\frac{2}{T_{sim}},\ \text{and} \\ \tau_{pole} &= K_{Div}R_1\left(C_z+C_p\right)\frac{2}{T_{sim}}. \end{aligned} \tag{18}$$

V. Compensator Model

Digital controllers compensate the error voltage using digital-filter techniques. This enables the compensator to be programmable. It also allows the manufacturer to incorporate a nonlinear response to the error. The required number of poles and zeros in the digital filter depends on the application. For a voltage-mode buck regulator, two zeros are needed to compensate for the second-order plant (power stage) and a pole at the origin is needed to minimize steady-state error. Control engineers will recognize this as a proportional, integral, derivative (PID) compensator. Equation (19) is this two-zero, one-pole filter expressed in the discrete-time, or *z* domain. Here z^{-1} represents a unit-sample delay.

$$G_c(z) = \frac{d(z)}{e(z)} = \frac{K_0 + K_1 z^{-1} + K_2 z^{-2}}{1 - z^{-1}} \qquad (19)$$

One approach to constructing a digital filter to be used as the compensator is shown in Fig. 9. Here the multiplication of the error by the numerator-filter coefficients K_0, K_1 and K_2 is done using a table look-up technique. This is the method used by the Texas Instruments UCD9112 digital controller. By using a table for each product in the numerator of the compensator-transfer function, non-linear gains can be built into the table. The down side to this technique is that the tables grow geometrically with increasing dynamic range on the error signal. In the UCD9112 the sampled-error signal has a 4-bit range so each table is 16 elements long.

Additional poles can be incorporated into the compensator to shape noise in the compensated error. The addition of a second pole has the effect of smoothing the quantization error in the compensator output. In general, a two-zero/two-pole digital filter has the following form:

$$G_c(z) = \frac{d(z)}{e(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 - a_1 z^{-1} - a_2 z^{-2}} \qquad (20)$$

The discrete-time-domain difference equation resulting from Equation (20) is:

$$\begin{aligned} d[n] = {} & b_0 \times e[n] + b_1 \times e[n-1] + b_2 \times e[n-2] \\ & - a_1 \times d[n-1] - a_2 \times d[n-2] \end{aligned} \qquad (21)$$

Expressing the difference equation in this form is called the *direct* form. A direct-form digital filter has a topology that follows this equation as shown in Fig. 10. The Texas Instruments UCD92xx family of digital-POL controllers uses this filter arrangement for the compensator.

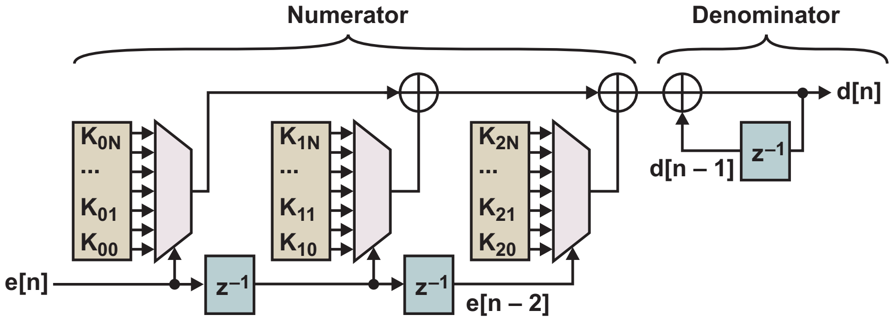


Fig. 9. Table look-up implementation of a digital filter.

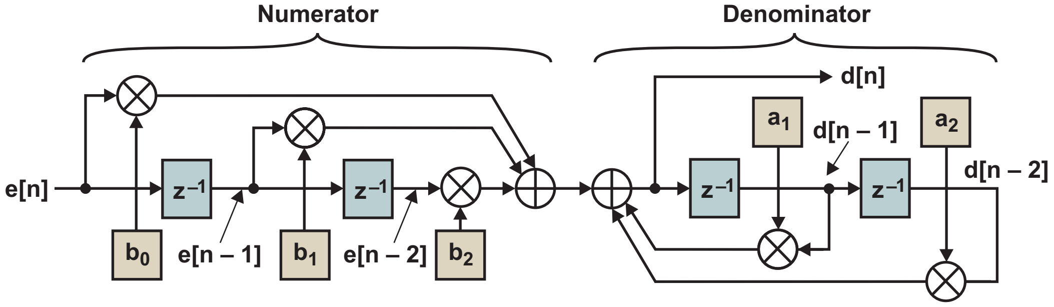


Fig. 10. Direct-form digital-filter implementation.

Finally, the two-pole, two-zero digital filter can be realized as a parallel-filter structure. Some digital controllers use this implementation for the compensator. The classic proportional, integral, derivative (PID) controller is a parallel filter structure. Even if the hardware realization of the filter is not implemented as a PID controller, it is useful to express the filter in this form mathematically.

$$G_c(z) = \frac{d(z)}{e(z)} = K_P + K_I \frac{z}{z-1} + K_D \frac{z-1}{z-\alpha} \quad (22)$$

In Equation (22), we see that a PID compensator is the sum of the voltage error multiplied by a proportional gain, K_P, the voltage error multiplied by a integral gain, $K_{I,}$ and accumulated, and the voltage error subtracted from the previous voltage error and multiplied by K_D. We can multiply this expression out so that it is expressed as a ratio of polynomials with a common denominator as in Equation (23) below to show that the PID controller is an equivalent representation of the direct-form filter.

In Fig. 11, $d_I[n]$ is the integrator state and represents the average duty cycle for the controlled loop. $d_D[n]$ is the derivative state and is zero at steady state. Of the three K gains, K_D is the largest and is a function of the location of the zeros in the compensator-transfer function. We will come back to this interpretation of the compensator when we discuss quantization.

A. Choosing the Digital-Compensator Coefficients

One of the most straightforward ways to determine what compensation to apply to a switch-mode power supply is to express the total open-loop gain as described in Equation (1) and plot the magnitude and phase of the loop gain as a Bode plot. From the Bode plot, the stability metrics of phase margin and gain margin can be determined and adjustments to the compensation made until the desired metrics are obtained. Since most power engineers are familiar with the behavior of the total loop gain as the gain and compensating zeros are changed, it is advantageous to first define the compensation as a continuous-time transfer function with specified gain, zeros, and poles. Then this continuous-time transfer function is

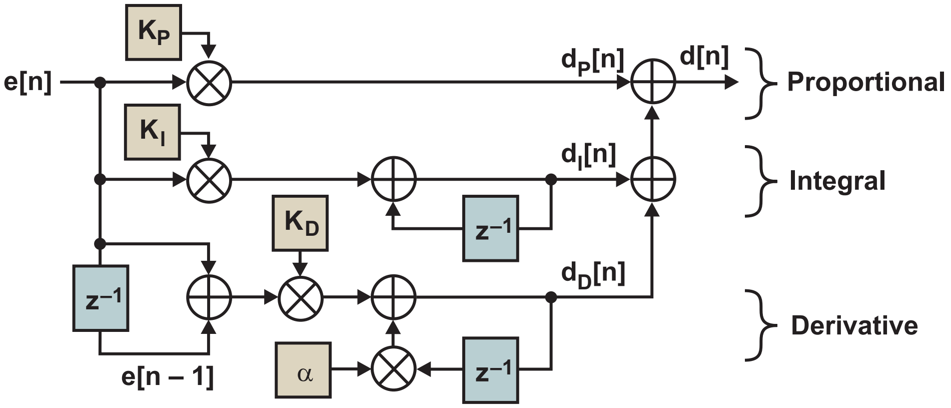


Fig. 11. PID digital-filter implementation.

$$G_c(z) = \frac{\left(K_P + K_I + K_D\right)z^2 - \left(K_P\left(1+\alpha\right) + K_I\alpha + 2K_D\right)z + \left(K_P\alpha + K_D\right)}{z^2 - \left(1+\alpha\right)z + \alpha} \quad (23)$$

transformed to the discrete-time domain to determine the discrete filter coefficients.

$$G_c(s) = K_{DC} \frac{\left(\frac{s}{\omega_{z1}}+1\right)\left(\frac{s}{\omega_{z2}}+1\right)}{s\left(\frac{s}{\omega_{p2}}+1\right)} \qquad (24)$$

Either Equation (24) or (25) can be used to describe the continuous-time prototype controller. If the compensating zeros are complex, Equation (25) is a more convenient form.

$$G_c(s) = K_{DC} \frac{\frac{s^2}{\omega_r^2}+\frac{s}{\omega_r Q}+1}{s\left(\frac{s}{\omega_{p2}}+1\right)} \qquad (25)$$

Once the design parameters of ω_r, Q, ω_{p2} and K_{DC} are determined, the continuous-time transfer function is mapped to the z-domain using the bilinear transformation.

$$s = 2f_{sw} \frac{z-1}{z+1} \qquad (26)$$

Here f_{sw} is the sampling frequency used by the compensating digital filter. This is typically the switching frequency. This mapping results in the following relationship between the continuous-time design parameters and the discrete-filter coefficients:

$$\begin{aligned} b_0 &= K_{DC}\left(\frac{\omega_p}{2f_{sw}+\omega_p}\right)\left(\frac{2f_{sw}}{\omega_c^2}+\frac{1}{\omega_c Q}+\frac{1}{2f_{sw}}\right) \\ b_1 &= K_{DC}\left(\frac{\omega_p}{2f_{sw}+\omega_p}\right)\left(\frac{-4f_{sw}}{\omega_c^2}+\frac{1}{f_{sw}}\right) \qquad (27) \\ b_2 &= K_{DC}\left(\frac{\omega_p}{2f_{sw}+\omega_p}\right)\left(\frac{2f_{sw}}{\omega_c^2}-\frac{1}{\omega_c Q}+\frac{1}{2f_{sw}}\right) \end{aligned}$$

$$\begin{aligned} a_1 &= \frac{-4f_{sw}}{2f_{sw}+\omega_p} \\ a_2 &= -\omega_p\left(\frac{2f_{sw}-\omega_p}{2f_{sw}+\omega_p}\right) \end{aligned} \qquad (28)$$

To evaluate the frequency response of the compensator as part of the total open loop gain, the discrete-time transfer function in z is evaluated by substituting

$$z = e^{j\omega T}, \qquad (29)$$

where T is the sample period for the compensating digital filter. This is preferable to evaluating the continuous-time prototype compensation-transfer function because there are distortions introduced when doing the bilinear mapping from continuous time to discrete time. This is because the bilinear transformation maps the entire left-half s-plane into the unit circle of the z-domain. As a result, the specified continuous-time transfer function will differ slightly from the discrete-time transfer function at frequencies near the Nyquist frequency. This distortion does not cause a problem as long as the Bode plot, and the stability metrics derived from it, are constructed using the discrete-time

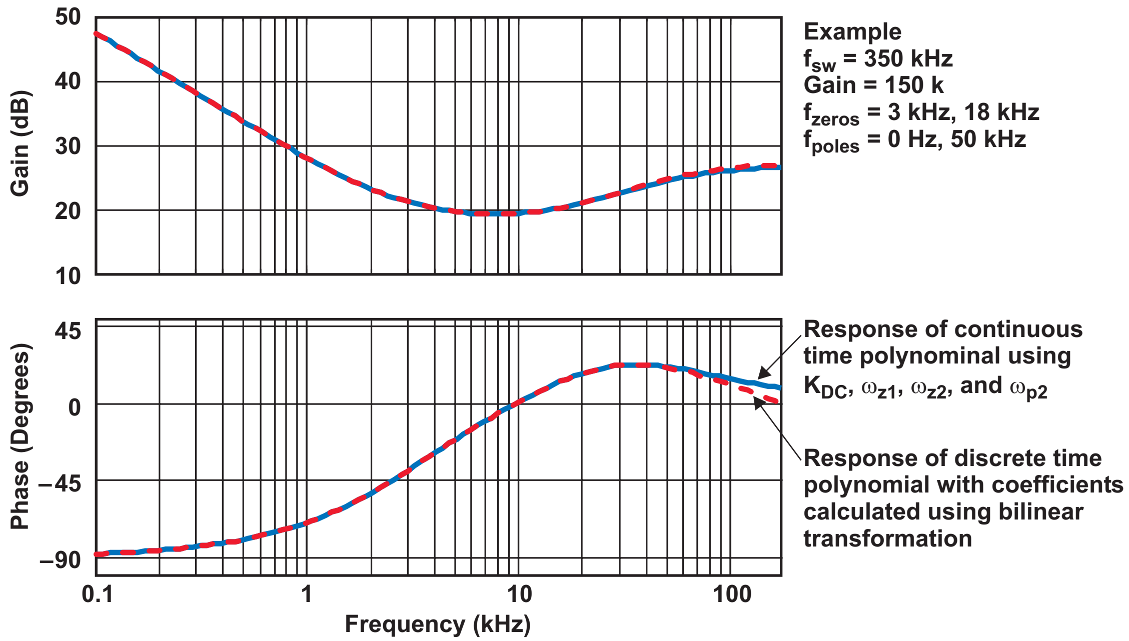


Fig. 12. Frequency response of continuous-time and discrete-time compensator.

transfer-function polynomial and the substitution of Equation (29). Equation (29) assumes there is an exact mapping between the continuous and discrete-time domains (see Fig. 12).

The most basic method of designing the compensator tunes the zeros and poles to obtain the desired bandwidth while maintaining reasonable phase and gain margins (>45º and >10 dB, respectively). However, these criteria may not be sufficient to provide optimal performance. Fig. 13 shows the loop gain of the switching converter introduced later in this paper for two different compensators. Both were designed in the s-domain and mapped to the z-domain with a bilinear transformation. Furthermore, both compensators contain a high-frequency pole at $\omega_p = 2\pi \times 50$ kHz and a DC gain of $K_{dc} \approx 21150$. The first compensator includes one real zero at the corner frequency, $\omega_{0,}$ of the power stage's control-to-output-voltage transfer function, $\omega_{z1} = \omega_0 = 2\pi \times 5.4$ kHz, and a second zero at $\omega_{z2} = 2\pi \times 6$ kHz. This compensator, $G_{c1}(z)$, achieves a phase margin of $\phi_m = 50°$, a bandwidth of $f_c = 30$ kHz, and a gain margin of $G_m = 18$ dB.

The second compensator, G_{c2}, provides the same bandwidth, f_c, while maintaining a higher phase margin. This compensator contains two complex zeros at the corner frequency of the power stage's control-to-output-voltage transfer function, ω_0. The Q_c value of the compensator in Equation (25) is close to the Q value of the power stage's control-to-output-voltage transfer function.

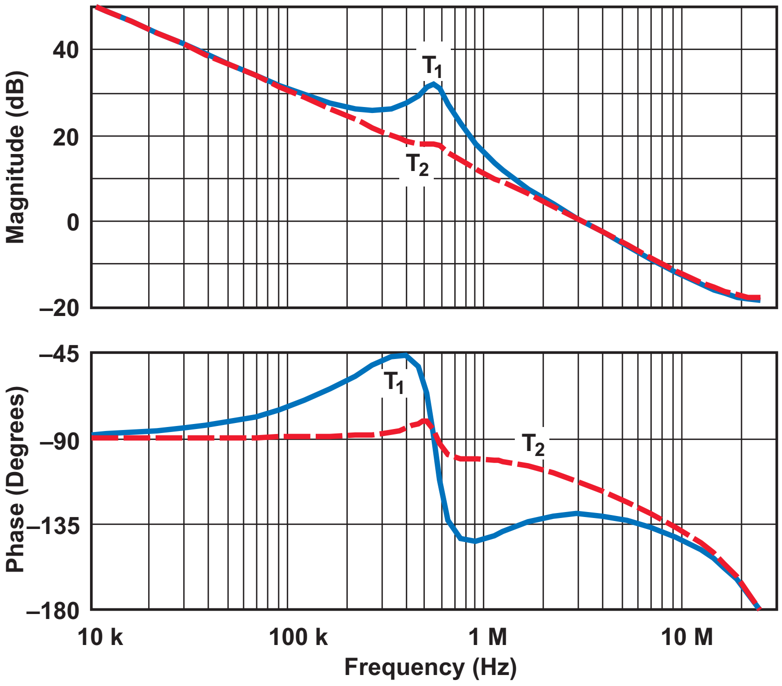


Fig. 13. The closed-loop gain of the buck converter with (1) real zeros and (2) complex zeros.

This results in approximate cancellation of the double pole of the power-stage output filter. Fig. 13 shows that $G_{c2}(z)$ flattens the loop gain and increases the phase margin to $\phi_m = 65°$. Intuitively, the second compensator would be expected to perform better than the first. However, the closed-loop converter with $G_{c2}(z)$ responds poorly to load transients. Fig. 14 shows why. The closed-loop output impedance, Z_{out_cl}, of the converter with the Q of the zeros set to match the Q of the power stage, has a higher peak output impedance and therefore has a poorer response to load transients. For this reason, it is advantageous for the design software to present the expected closed-loop output impedance. The closed-loop output impedance is

$$Z_{out\_cl} = \frac{Z_{out\_op}}{1+T}. \qquad (30)$$

Fig. 15 shows the results of simulating the closed-loop buck converter in MATLAB Simulink® with the two compensators, $G_{c1}(z)$ and $G_{c2}(z)$, for a load transient of 1 to 10 A. The closed-loop feedback using $G_{c2}(z)$ introduces significant ringing at ω_0 due to the larger magnitude of Z_{out_cl2} at this frequency.

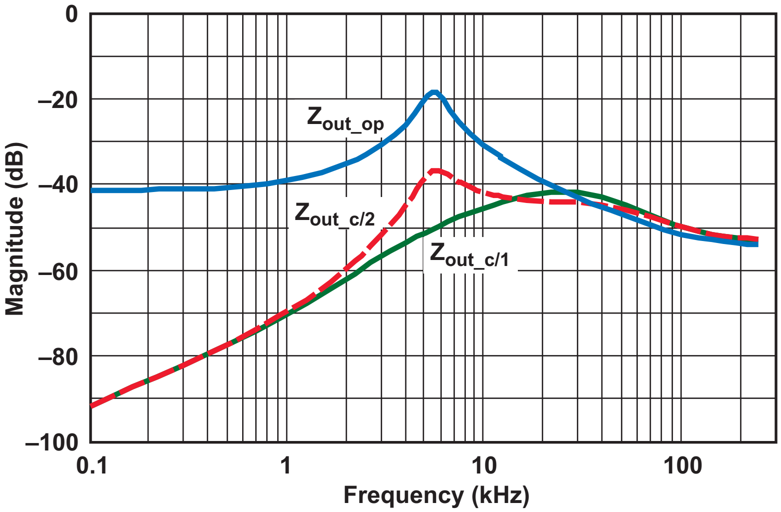


Fig. 14. The magnitude response of the open-loop and closed-loop output impedance.

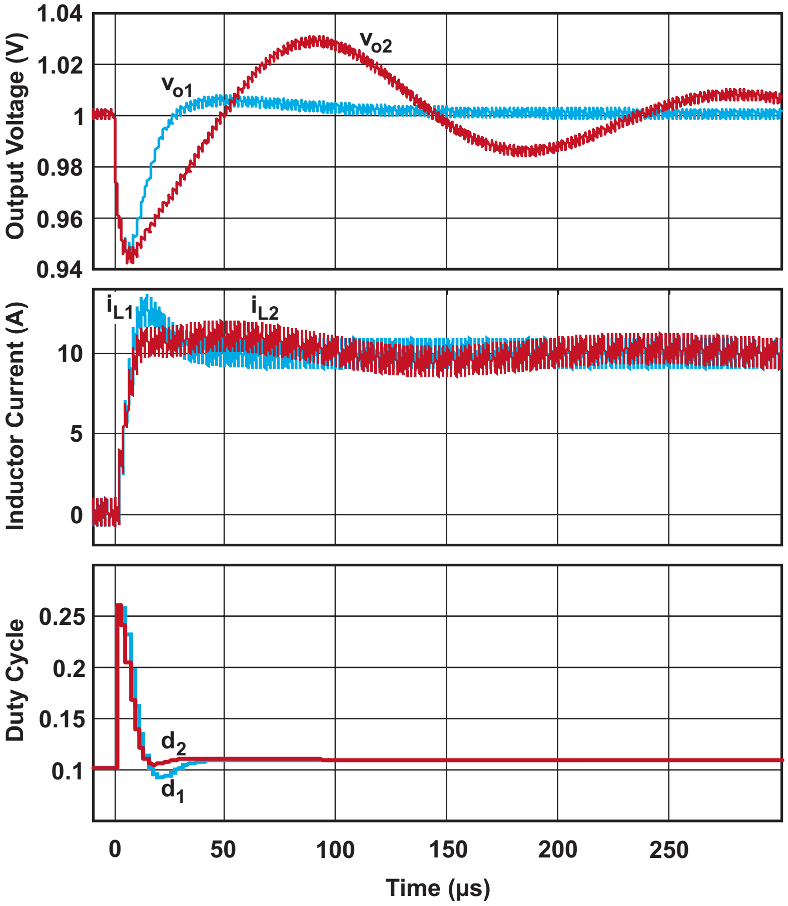


Fig. 15. Simulation results for a load transient of 1 to 10 A.

Simulink is a registered trademark of The MathWorks, Inc.

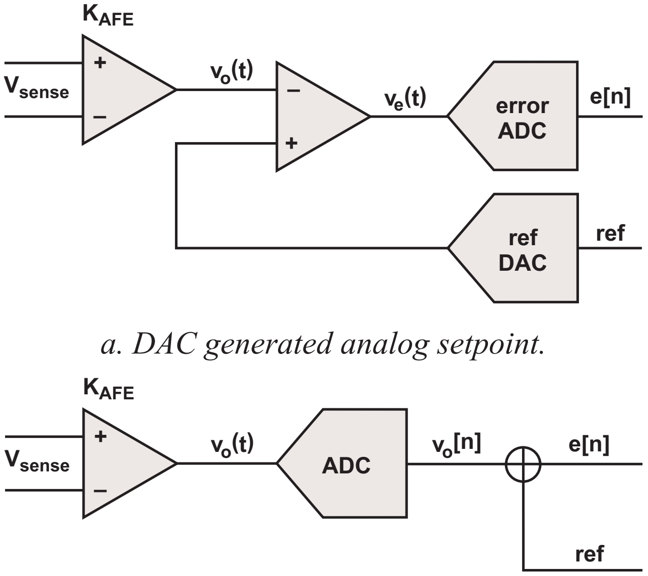
VI. Modeling the Voltage-Sense ADC and Digital-PWM Engine

The three blocks that close the feedback loop and regulate the output voltage in a switching converter are the input ADC, the compensating digital filter, and the digital-PWM engine. The input ADC and the output digital PWM introduce quantization error into the system. Also, these functional blocks are the key elements in determining the precision at which the output voltage can be regulated.

Regulating the sensed feedback signal to the desired level requires a precise reference voltage, V_{ref}. When this reference is programmable, it can be used to precisely control soft start/stop ramps under closed-loop control, to perform margining, and to provide a means of automated calibration. Factors that define the resolution requirements for the setpoint-reference DAC are: 1) the desired setpoint resolution, and 2) the allowable error in the output voltage during the soft start/stop ramp due to the discrete steps in the DAC output compared to a linear ramp. Resolution of a few millivolts is usually adequate.

There are two ways to subtract the sensed output voltage from the setpoint reference. It can be done in a summing amplifier ahead of the ADC or it can be done digitally (see Fig. 16), after the sensed output voltage has been converted to numerical values. In the first case, in order to provide programmability to the voltage setpoint, a setpoint reference DAC is provided and the error voltage is converted by the ADC. Since the resulting error voltage has a much smaller dynamic range than the sensed output voltage, a fast, narrow-range ADC can be implemented in the controller. In this case the setpoint resolution, that is, the precision to which the output voltage can be programmed, is defined by the setpoint reference DAC. The resolution of the error ADC is important to the dynamic control of the output voltage, but does not influence the average DC-output voltage. An extreme example of this is the thermostat in your house. It contains a precision setpoint reference and a one-bit ADC with infinitely large resolution.

The other way a digital controller can subtract the sensed output voltage from the setpoint reference is to first digitally sample the output voltage with an ADC with a wide dynamic range and then subtract this numerical value from a digital setpoint reference. In this case, the resolution of the ADC does in fact define the resolution at which the system can set the output voltage.



a. DAC generated analog setpoint.

a. Digital setpoint.

Fig. 16. Comparison of analog and digital setpoint reference.

Examples of digital controllers that use the narrow-range ADC and separate setpoint-DAC are the UCD9112 and UCD9240. On the other hand, TI DSP controllers such as the TMS320C2801 use a full-range ADC.

A. Consequences of Sense-Voltage Quantization

With any ADC, whether an absolute type or window type, there is a trade-off between dynamic range and resolution. By including an analog programmable gain block in the error-voltage signal path, the dynamic range and resolution can be tailored for a given application. The system in Fig. 17 has programmable gains of 1, 2, 4, and 8. For an analog gain of $G_{AFE} = 1$, this error ADC provides a resolution of 8 mV/LSB and a dynamic range of ±256 mV, while for $G_{AFE} = 8$, it has a resolution of 1 mV/LSB and a dynamic range of ±32 mV. Voltage-mode digitally-controlled DC/DC converters usually require an eADC resolution of better than 0.2%. For example, non-isolated DC/DC applications usually need error-ADC

Topic 7

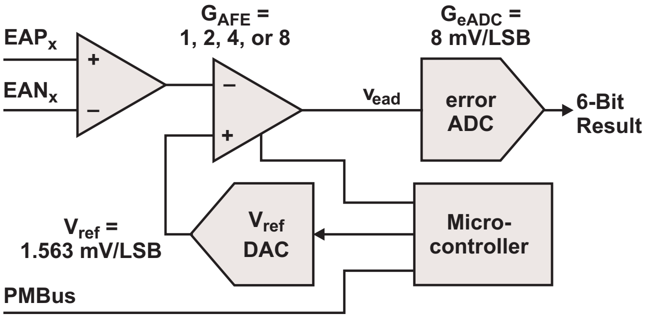


Fig. 17. Block diagram of an eADC together with a programmable DAC.

resolutions of 1 to 5 mV. The required range of the eADC depends on its application and should cover the required operational range for the load.

When we calculated the coefficients for the digital-compensation filter, the dynamic range of the digitized output-sense voltage was not taken into account. Most of the response to fast disturbances to the control effort (such as a load step) is provided by the effective differential gain of the compensator. The digital filter generates this control effort by subtracting the most recent error-voltage sample from the previous error sample and multiplying by a gain. In fact the differential gain is typically the largest gain in the compensator. If the error voltage saturates because of the dynamic range of the ADC, the differential contribution to the control effort will drop to zero and the transient response will be substantially degraded.

As an example of the effect of error-ADC saturation, consider a system based on a continuous-time compensator with a DC gain of 14500, and zeros at 1.9 kHz and 16 kHz. Expressing the digital compensator as a parallel-form PID filter, the gain coefficients are:

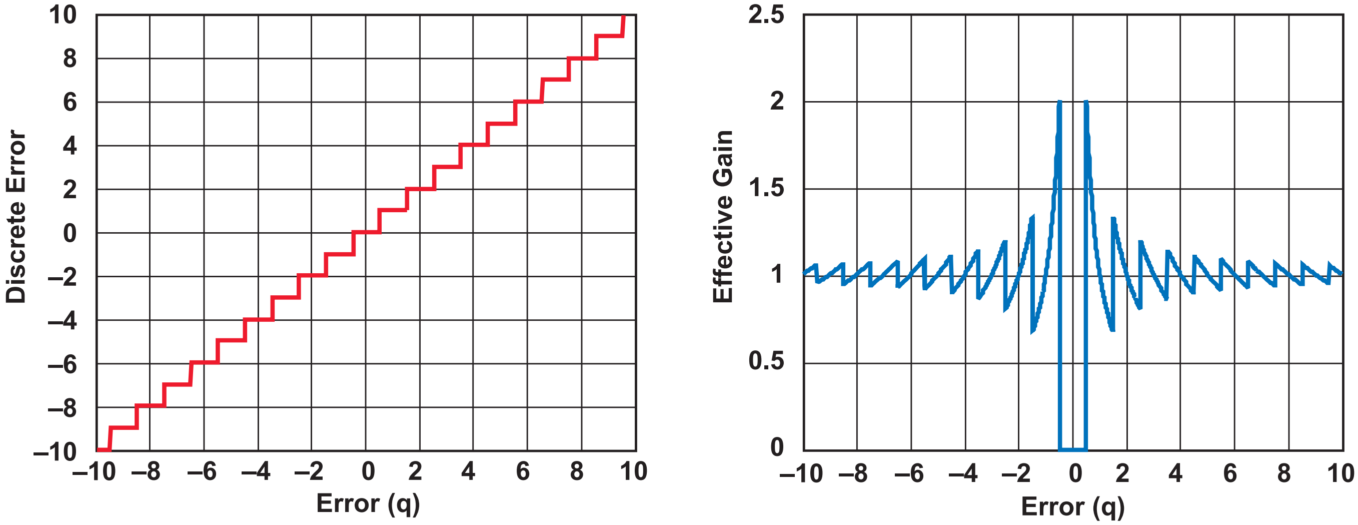
$$K_P = 0.86;\ K_I = 0.03;\ K_D = 2.31$$

In this case, the differential gain is 77 times larger than the integral gain, so if the error ADC saturates, a substantial portion of the control effort will be lost.

If the dynamic range of the error voltage is one side of the coin, quantization error introduced by the error ADC is the other side of the coin. The error ADC effectively rounds the error voltage to discrete integer values. Therefore, the output of the ADC, e[n], changes at voltages defined as

$$e[n] = q\left(n + \frac{1}{2}\right), \tag{31}$$

where q is the quantization interval, or resolution, of the ADC. The ADC output gain, relative to the continuous-time input-error voltage is then shown in Fig. 18b. At steady state, the error voltage will be near zero and a discrete sequence of error values {0, 0, 0, ...} will be applied to the compensator. When the system is perturbed by a small amount, the sequence will include isolated +1 or –1 values. Here it is instructive to use a direct-form compensation filter to observe what happens. For the example where the continuous-



a. ADC in/out transfer function with quantization.

b. ADC gain due to quantization.

Fig. 18. Error-voltage quantization.

time prototype compensator has a DC gain of 14500, and zeros at 1.9 kHz and 16 kHz, the direct-form numerator coefficients are

$$b_0 = 0.539; b_1 = –0.942; b_2 = 0.406.$$

This means that an isolated +1q error value will generate a control-effort output sequence of

$$\{d, d, d + 0.539\%, d – 0.942\%, d + 0.406\%,$$
$$d + 0.0034, d + 0.0034, d + 0.0034,...\},$$

where d is the nominal duty cycle and % means percent of the switching period. So, in this example, if the switching frequency is 333 kHz, the pulse width will swing from increase by 16.2 ns, then decrease by 28.3 ns, then increase by 12.2 ns, then return to the nominal pulse width plus 100 ps.

B. Application of Nonlinear Gain

To comply with today's tight requirements for the transient performance of switching converters, the digital controller can apply nonlinear signal processing to the compensation. This application of nonlinear gain can in some cases make up for the loss of phase margin due to the computational delay inherent in a digital controller. Digital nonlinear control techniques remain the subject of much research. The main difficulty with these techniques is to define stability criteria in the presence of all transients and operating conditions. This section introduces one of the digital nonlinear-control techniques implemented in TI devices.

Fig. 19 shows the diagram of a digital nonlinear gain block. This block is situated between the error ADC and the digital-compensation filter.

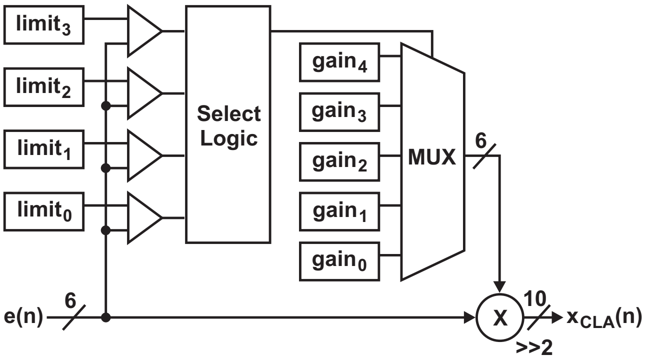
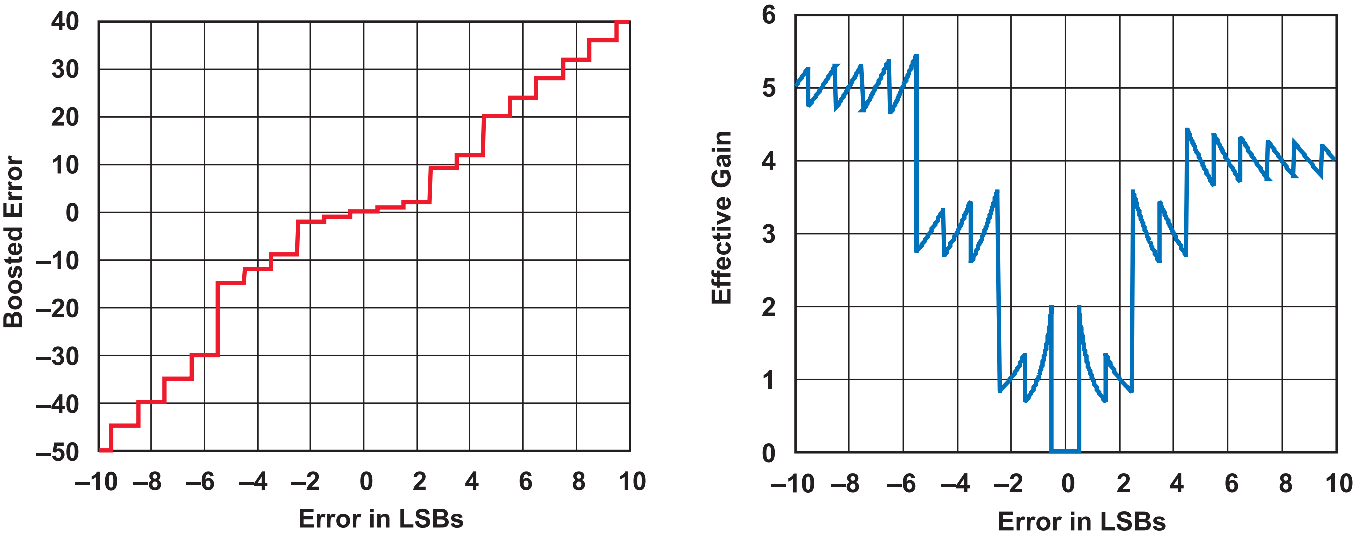


Fig. 19. Digital controller with nonlinear quantization.

The error signal is compared to a set of programmable limits which partition the dynamic range of the error signal into segments. This allows the gain to be individually programmed for each error-signal segment. As a result, the loop gain can be reduced for quiescent operating conditions where the error signal is near zero and the loop gain can be boosted when a transient forces the error away from zero. Fig. 20 shows an example where the comparator limits are set to –6, –3, +3, and +5 error-signal LSBs and the gains are set to 5.0, 3.0, 1.0, 3.0, and 4.0.

Loop stability should be determined using the largest gain applied by the nonlinear-gain block. This is particularly true if the gain is larger for error-signal values away from zero. It is these gain values that will be seen by the system during a transient and will determine the amount of ringing in the transient response.



a. Transfer function after nonlinear gain.

b. Gain with nonlinear gain applied.

Fig. 20. Application of nonlinear gain.

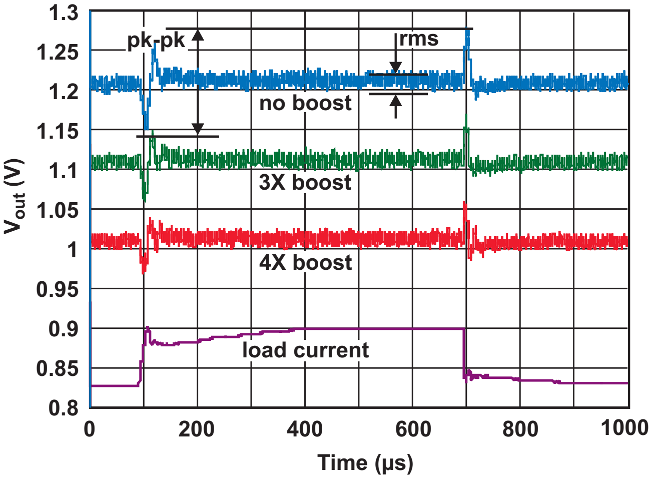
Fig. 21 shows captured scope traces of the output voltage with varying amounts of nonlinear-gain boost. The top trace has a uniform gain through the nonlinear-gain block, the second trace is the transient response for a case where a 3X gain is applied when LSBs. The third trace shows the response when the gain is 4X when LSBs.

VII. Digital PWM Engine

The DPWM forms one of the most important elements of a digital controller for a switching converter. As shown in Fig. 2, the DPWM obtains the duty-cycle information, d, at the output of the compensator and generates switching signals, c_1 to c_n, for the switching converter. High-resolution, high-frequency DPWMs are required to achieve high-bandwidth, precise-voltage regulation in digitally-controlled switching converters. Although dithering or sigma-delta approaches can be applied to improve effective DPWM resolution to some extent, it is very desirable to achieve high-hardware resolution using relatively minimal-hardware resources. A direct implementation of an analog PWM in the digital domain produces a counter-based DPWM [6]. An n-bit counter increments at each input clock period, t_{clk}. When the counter reaches its maximum value, it resets and starts counting from zero. The counter-based DPWM's input-clock frequency, f_{clk}, directly depends on the number of bits in the counter, n, and the desired switching frequency, f_{sw}: $f_{sw} = 1/T_S$, and $f_{clk} = 2^n f_{sw}$. A counter-based DPWM has the advantages of simplicity and linearity, however, this scheme requires many bits to achieve high resolution. Therefore, the required clock frequency, f_{clk}, can become unreasonably large. For example, a 10-bit DPWM at f_{sw} = 1 MHz requires a clock frequency of $f_{clk} \approx$ 1 GHz. This makes the implementation of a high-resolution, high-frequency counter-based DPWM very costly.

Hybrid DPWMs [7, 8, 9–11] can provide high resolution and high frequency without the very high input clock frequencies required by counter-based DPWMs, or the very large areas required by pure delay-line-based DPWMs [12]. Fig. 22 shows an example of a hybrid DPWM with an open-loop delay line [13]. The counter provides the most significant portion of the duty-cycle command, and the delay line provides the least significant portion.

The counter increments at each input clock, clk. The output, c_1, is set high at the zero value of



Parameter	Peak-to-Peak Output Excursion	RMS Error During Quiescent Operation
Uniform gain of 1X	130.5 mV	5.2 mV
Gain boosted 3X for $\lvert v_{err}\rvert > 5$	108.1 mV	5.0 mV
Gain boosted 4X for $\lvert v_{err}\rvert > 5$	88.4 mV	5.0 mV

Fig. 21. V_{out} with and without nonlinear-gain boost.

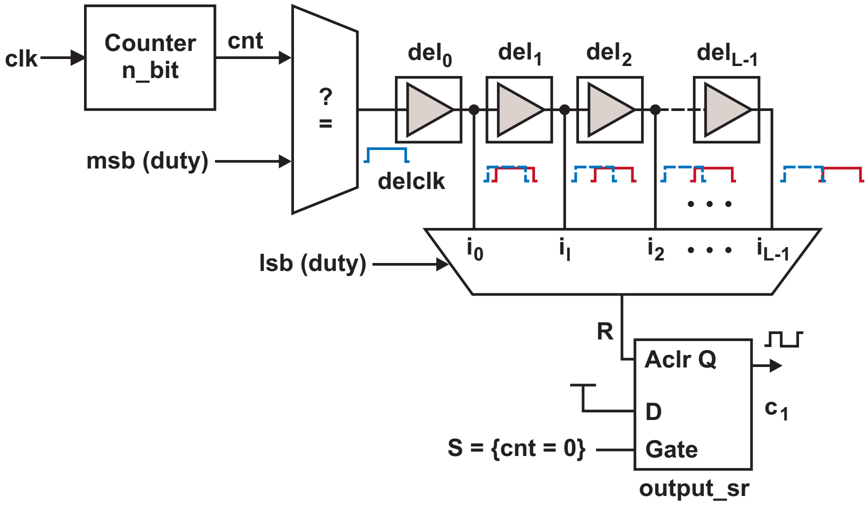


Fig. 22. Hybrid DPWM with external clock.

the counter. Comparing the output of the counter with the most significant bits of the input duty-cycle command, msb(duty), produces the signal delclk. The width of delclk equals one clock period of the input clk signal. The signal delclk propagates through the delay line. The output of each delay cell taps out and connects to an L:1 multiplexer. The multiplexer output is selected by observing the least significant portion of the input duty-cycle command, lsb(duty). The appropriate input of the multiplexer is then connected to the output, R, which resets c_1. It is important to note that the frequency of signal c_1 is still determined according to the relationship $f_{clk} = 2^N f_{sw}$, where N represents the number of bits in the counter. The hybrid DPWM uses a smaller counter and therefore requires a lower input-clock frequency than a counter-based DPWM with the same resolution.

To guarantee monotonicity and nearly optimal linearity, the total delay of the delay line should equal one clock period of the input clock. However, even in a very careful design, the cell delay varies with process and temperature. Therefore, an adjustable delay cell is required to achieve the desirable delay through the delay line. Furthermore, an active-control scheme must control the delay through each individual delay cell. In References [12, 14, 15], analog or digital phase-locked loops or delay-locked loops adjust the delay of the delay line in a DPWM in order to synchronize the operation to an external clock.

Fig. 23 shows another implementation of a hybrid DPWM using a combination of a ring oscillator and a counter to provide high resolution [7, 9]. This structure propagates a pulse around a ring containing L-delay cells. Unlike the structure shown in Fig. 22, this topology does not require an external clock. Instead, the ring oscillates and so provides an internal clock. The output of one of the delay cells connects to the input clock of the n-bit counter that determines the switching frequency of the DPWM. The output, c_1, is set when the counter value, cnt, equals zero. Like the hybrid delay line of Fig. 22, both the appropriate delay-cell output and the output of the n-bit counter determine the reset point of the DPWM. Reference [10] provides a complete list of different hybrid DPWM architectures.

We can determine the effect of the finite pulse-width resolution out of a digital PWM by noting that the output voltage of a buck regulator is D × V_{in}. Then, expressing D in terms of the effective clock rate of the PWM, the switching period and the number of clock cycles determined by the digital-compensator calculated control-effort, n, we can quantify the change in V_{out} for each discrete change in the PWM output. This is shown in Equations (32) and (33).

$$V_{out} = DV_{in} = \frac{NT_{clk}}{T_{sw}} V_{in} \tag{32}$$

$$V_{out}\ \text{Resolution} = T_{clk} f_{sw} V_{in} \tag{33}$$

For example, if V_{in} is 12 V, T_{clk} = 250 ps, and f_{sw} = 500 kHz, the V_{out} Resolution is 1.5 mV.

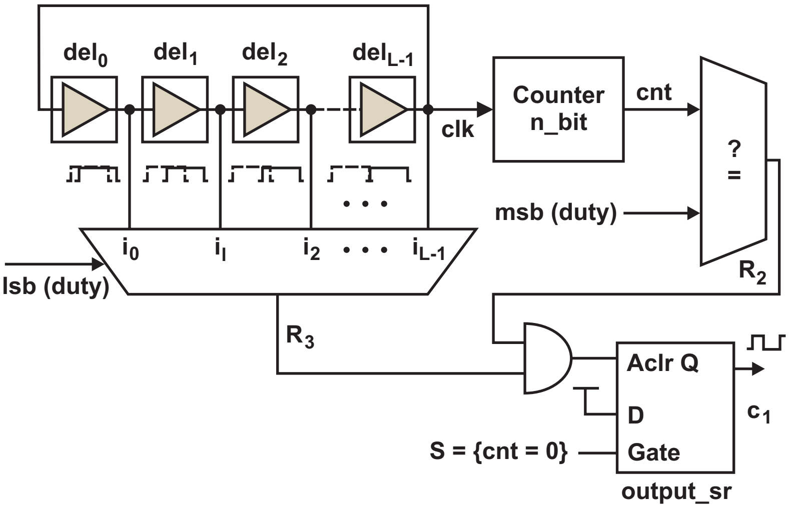


Fig. 23. Hybrid DPWM with ring oscillator.

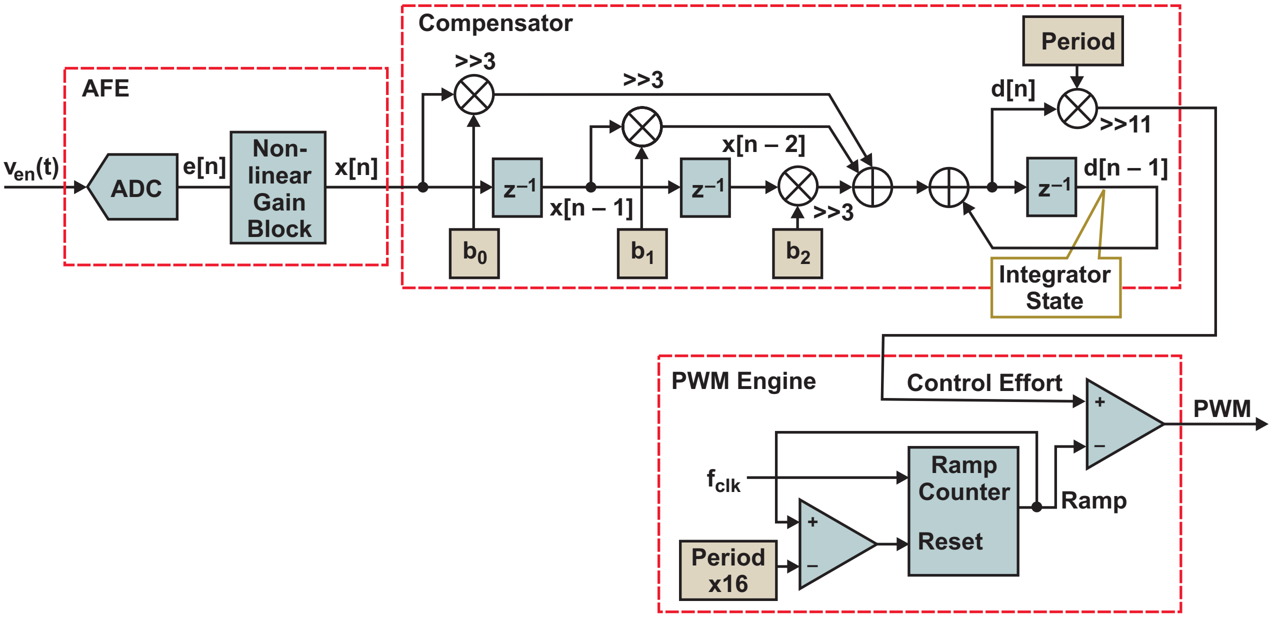


Fig. 24. Location of the integrator state in the controller system.

A. Limit Cycling

Digital-PWM resolution and error-ADC resolution together determine output-voltage regulation. If the change in the output voltage resulting from a one-step change in the digital PWM is much greater than the input resolution of the error-voltage ADC, limit cycling can occur [16, 17]. Limit cycling is the condition where the output voltage oscillates around the nominal output voltage. It can be predicted by examining the quantization in each of the three major blocks in the controller. Limit cycling occurs when the voltage resolution coming out of the integrator state is larger than the voltage resolution coming into the integrator state. When this occurs, the register or variable holding the integrator state acts as a hysteretic memory element, where we have to dump "charge" into or out of the state before a change in the output can be observed.

To determine if limit cycling will occur, the quantization (expressed in volts) must be calculated for the signal path into the integrator state and for the signal path out of the integrator state. In Fig. 24, we can work backwards from the PWM signal to the integrator state. From Equation (33), we know the change in output voltage for each LSB change in the PWM counter threshold. The counter threshold is the result of the compensator output times the switching period. In the UCD9240, this is a fixed-point multiply where d[n] is a 16-bit signed value and Period is a 14-bit unsigned value. The result is a 30-bit value, of which we keep 18 bits by rounding and then dropping the bottom 11 bits and the now redundant sign bit. So for this device, when the switching frequency is 500 kHz, the value in the Period resistor is 500. This means that the integrator state must change by 4 LSBs for the counter threshold to change by one LSB and for the output voltage to change by 1.5 mV. Further simplifying, for this device, each one LSB change in the integrator state causes a $V_{in}/2^{15}$, or 0.366 mV, change in the output voltage with a 12-V input.

$$\begin{aligned}\frac{V_{out}}{d[n-1]} &= \frac{f_{sw}}{16\times f_{clk}}\times V_{in}\times\frac{\text{Period}}{2^{11}}\\ &= \frac{f_{sw}}{16\times f_{clk}}\times\frac{V_{in}}{2^{11}}\times\frac{f_{clk}}{f_{sw}}=\frac{V_{in}}{2^{4+11}}\end{aligned}\qquad(34)$$

Working from the input, the effect on the integrator state for a one-LSB error is the ADC gain times the nonlinear gain times the integrator gain of the compensator. The integrator gain (K_I) is the sum of the b_0, $b_{1,}$ and b_2 multiply operations. For an example power supply, we choose the ADC gain to be 1/resolution = 1/2 mV; the nonlinear gain to be 2.0; and the integrator gain to be 933 – 1532 + 611 = 12. Then the input voltage change that causes a one-LSB change in the

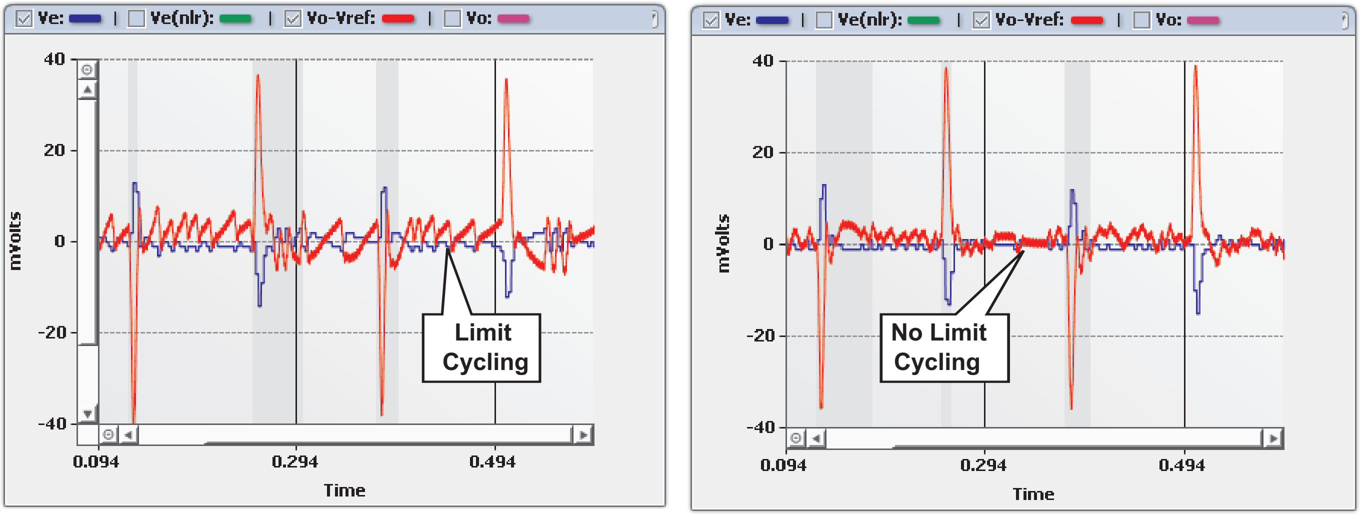


Fig. 25. Limit cycling with low nonlinear gain.

integrator state is 0.667 mV. So in this case, the criteria that the output resolution be finer than the input resolution is satisfied.

$$d[n-1] = v_{err}(t) \times K_{ADC} \times K_{NLR} \times (b_0 + b_1 + b_2) \times 2^{-3}$$

$$\frac{v_{err}(t)}{d[n-1]} = \frac{2^3}{K_{ADC} \times K_{NLR} \times (b_0 + b_1 + b_2)} \qquad (35)$$

From Equation (35), we can see that the compensator gain, including the ADC and nonlinear gains, determines if limit cycling will occur. Fig. 25 shows the output voltage for a system where the nonlinear gain has been reduced by a factor of 8, from 4.0 to 0.5, for the zero-error voltage gain. In this case the input resolution is no longer larger than the output resolution and limit cycling occurs.

VIII. System Identification and Autotuning

Most SMPS controllers are designed to obtain the highest achievable dynamic performance. Such designs require precise knowledge of the system parameters and operating conditions. Furthermore, for a power module that operates in parallel or interacts with other modules, the overall system model might not be predictable. Therefore, a controller optimized for one set of operating conditions might actually become unstable under other conditions. Real-time system identification and controller autotuning can fine-tune controller parameters, either during development or in actual operation. This section describes the system identification and autotuning techniques used in the TI UCD9240 digital controller.

A. System Identification

Power-distribution systems containing multiple-power sources may experience dynamic-performance degradation and even instabilities caused by uncertainties in the system parameters and interactions between different power modules. An intelligent system within each power module can perform system identification. A central or distributed controller can use these results to fine-tune controller parameters. The inherent flexibility and programmability of digital controllers make them attractive platforms for automatic system identification. There are several digital-system identification techniques [18–20]. In each case, the system is perturbed by an excitation signal and then the response to that excitation is measured. The excitation signal can be an impulse, a step, white noise, or a sine wave. Exciting the system by injecting a sine wave into the feedback loop is the technique used by power system/dynamic network analyzers because it produces the highest signal-to-noise measurement of the system-transfer function. The UCD9240 digital controller and it's supporting design software use this approach.

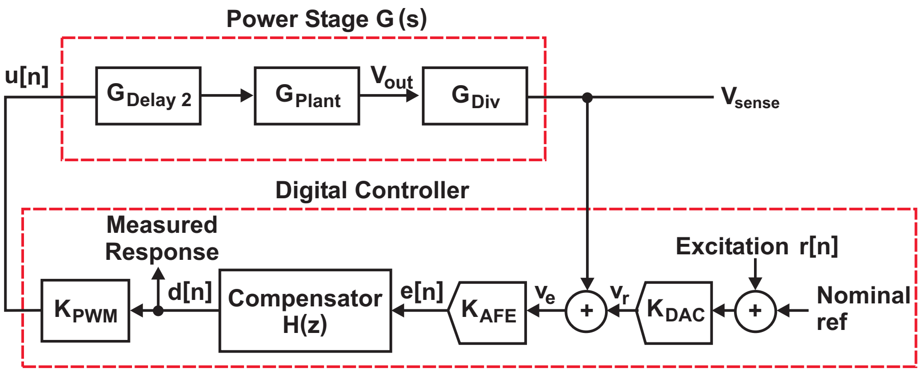


Fig. 26. The injection and measurement for system identification in the UCD9240.

The system identification is entirely performed in the UCD9240 and the results are reported back to the GUI. This technique synthesizes a digital-sinusoidal signal and injects it into the closed-loop system. Then the response to that excitation is measured at another point in the loop. From this measured closed-loop response the open-loop gain is calculated. Repeating this over a range of frequencies provides the data necessary to create the Bode plot for the system. Fig. 26 shows the location of the injected signal, r[n], and of the measurement points e[n] and d[n] in the UCD9240.

In the usual way, we can write the transfer function from the reference to the output voltage.

$$\frac{v_{out}}{ref} = \frac{K_{DAC}T(z)}{1+T(z)}, \qquad (36)$$

where

$$T(z) = K_{AFE}H(z)K_{PWM}G_{Delay}G_{Plant}G_{Div}.$$

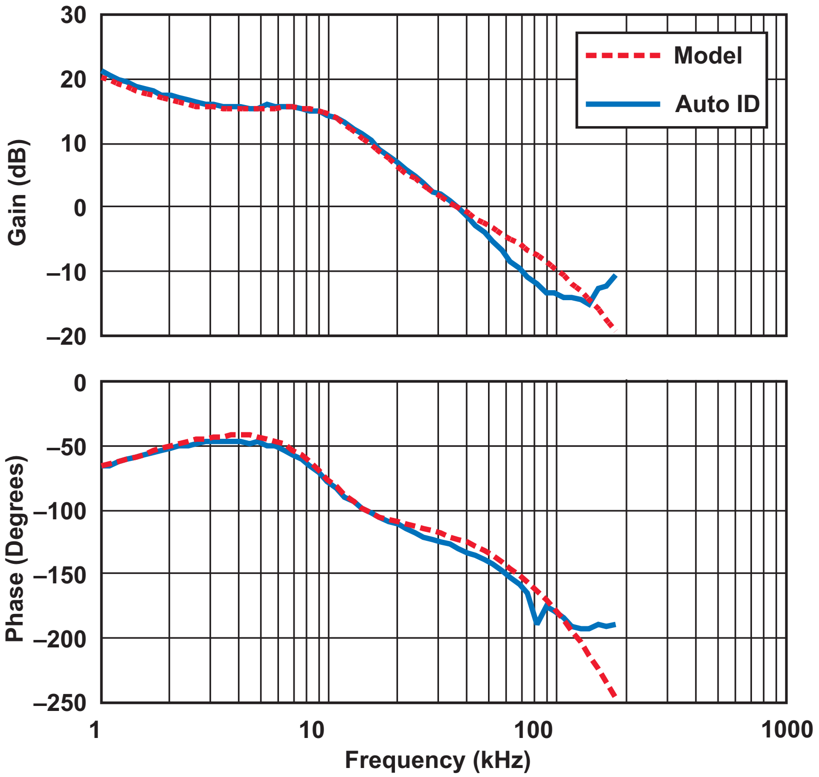


Fig. 27. Bode plot measured with Auto-ID versus model.

In the same way, we can write the transfer function from the excitation x[n] to d[n].

$$\frac{d(z)}{x(z)} = \frac{K_{DAC}K_{AFE}H(z)}{1+T(z)} \qquad (37)$$

Then we can solve Equation (37) for G_{Plant}.

$$G_{Plant} = \frac{1}{K_{PWM}G_{Delay}G_{Div}} \times \left(\frac{K_{DAC} \times x}{d} - \frac{1}{K_{AFE}H(z)} \right) \qquad (38)$$

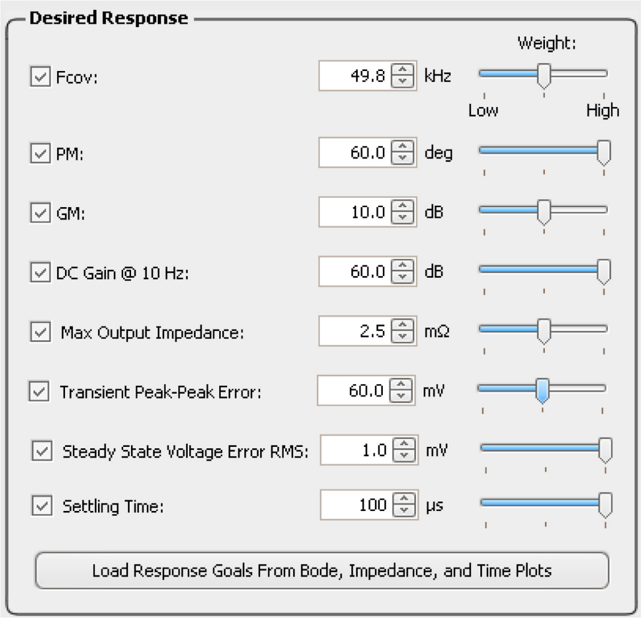
To generate the excitation signal, a table look-up technique is used. The table contains a sequence for one period of the sine wave and a pointer is stepped through the table at different rates to generate each excitation frequency. To measure the response, the same table is used to generate a cosine and a sine sequence. These two sequences are multiplied by the response vector, d[n], and summed to obtain a complex estimate of the response at the excitation frequency. This is repeated at each frequency for which a measurement is desired. From the complex estimate of the response, we can generate the Bode plot for the system (see Fig. 27).

Because most of the gains in the system are digital—the compensator, the digital PWM, the computational delay, etc.—this technique produces an accurate estimate of the transfer function of the power stage. Then, once we have that estimate, we can use it to make decisions about the optimal compensation of the loop.

B. Autotuning

Traditional controllers usually require redesign or retuning during development to account for wide variations in the power-supply parameters and operating conditions. Most of the well-known digital autotuning techniques [21, 22] try to shape the frequency response of the closed-loop gain. The controller coefficients are tuned to achieve the desired phase and gain margins and loop bandwidth. These techniques do not guarantee optimal time-domain-transient performance.

A user interface that can communicate with the digital controller provides an adequate tool for autotuning. The TI GUI [23] can perform autotuning based on both the frequency response of the system and the time-domain simulation results. Autotuning based on frequency response can use the simulation results or the system identification technique described in Section VI.A. Fig. 28 shows the Auto-Tune design screen of the TI GUI for the UCD9240. The autotuning process uses criteria such as crossover frequency, phase margin, gain margin, DC gain (the loop gain at 10 Hz), and the maximum closed-loop output impedance for frequency-response shaping. The user chooses the desired value and the weight applied to each criterion. The GUI also supports time-domain criteria including settling time, overshoot, and undershoot. The GUI iterates the compensator coefficients to achieve the desired frequency response and time-domain simulation results. After evaluating the results, the user can load the final compensator coefficients into the device register of the UCD9240.



Desired Response

☑ Fcov: 49.8 kHz

☑ PM: 60.0 deg

☑ GM: 10.0 dB

☑ DC Gain @ 10 Hz: 60.0 dB

☑ Max Output Impedance: 2.5 mΩ

☑ Transient Peak-Peak Error: 60.0 mV

☑ Steady State Voltage Error RMS: 1.0 mV

☑ Settling Time: 100 μs

Fig. 28. TI GUI Auto-Tune design screen for the UCD9240.

IX. Design Example

This section describes a design example based on the UCD9240 digital controller [7]. Fig. 29 shows the implementation of a two-phase buck converter using an integrated driver and power MOSFET (DrMOS) [24], and the UCD9240 digital-power POL-system controller [7]. The signals CS.1 and CS.2 are used for current sharing within the UCD9240.

The power-stage parameters are as follows: $L_1 = L_2 = 0.363\ \mu H$, inductor DC resistance DCR1 = DCR2 = 2.4 mΩ, total output capacitance $C = 3 \times 470 + 12 \times 47\ \mu F$, capacitor ESR = 1 mΩ, $V_{in} = 10$ V, $V_{out} = V_{ref} = 1$ V, $f_{sw} = 350$ kHz, and the voltage-sense gain = 0.8.

A. UCD9240 Digital Controller

Fig. 2 shows a simplified block diagram of the UCD9240 digital controller [7]. This controller incorporates a window-type error ADC with analog front-end gain, G_{AFE}, and programmable gains of 1, 2, 4, and 8. The output of the eADC has a 6-bit signed-integer format. The resolution of the error ADC equals 1 mV for $G_{AFE} = 8$. The UCD9240 also provides a programmable reference using a 10-bit DAC. The DAC has a dynamic range of 1.6 V, giving a resolution of 1.6 mV. Furthermore, the sampling frequency of the eADC depends on the digital PWM switching frequency, f_{sw}. The sampling location during one switching period is programmable.

The UCD9240 includes a high-resolution hybrid DPWM as shown in Fig. 23. The digital PWM implements a 4-bit ring oscillator. The clock frequency of the counter equals 250 MHz. However, because of the hybrid nature of the PWM engine, the resolution of the duty cycle is actually 250 ps. The PWM engine uses a programmable switching frequency between 15 kHz and 2 MHz with 4-ns resolution. It also provides several outputs for synchronous rectification and supports programmable dead times with 250-ps resolution. The dead-time programmability of the DPWM module helps optimize efficiency in a synchronously rectified DC/DC converter [25].

The input to the digital PWM consists of a signed 16-bit duty-cycle value from the compensator. The PWM-engine design ensures that for any programmed value of switching frequency, the maximum value of the duty cycle (7FFF hex) provides a 100% PWM output (always on).

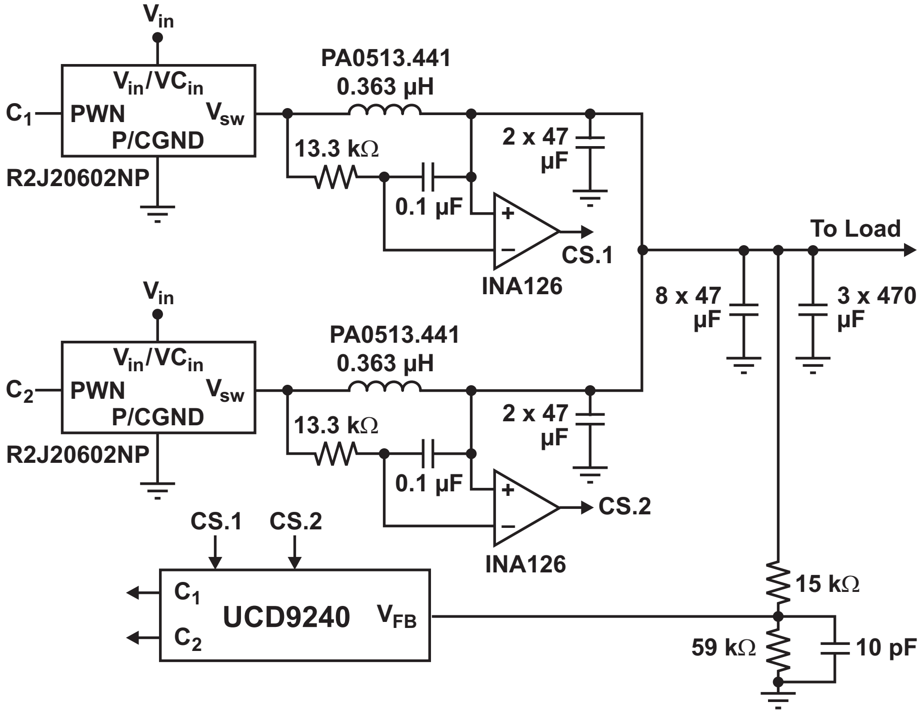


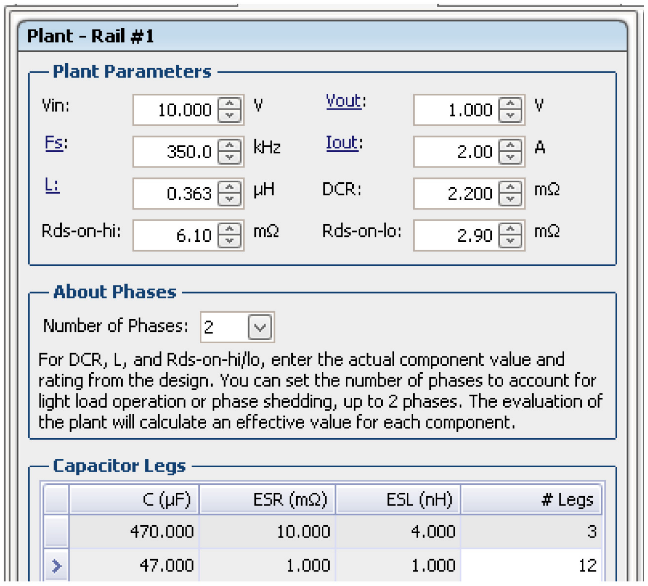
Fig. 29. Block diagram of a two-phase buck converter using the UCD9240 digital controller.

The UCD9240 also contains a direct-form, two-zero/two-pole compensator like the one shown in Fig. 10. It also includes a cascaded first-order filter that adds another programmable pole and zero to the compensator. The compensator coefficients occupy a programmable 12-bit format. The sampling rate of the compensator is synchronized with the sampling rate of the eADC. In addition, a nonlinear gain block, shown in Fig. 19 resides at the input of the compensator.

B. Controller Design

The first step in designing the controller is to model the power stage. A PC program [23] is available for the UCD9240 digital controller that allows the user to design the controller with minimal effort. The user interface provides the power stage, loop gain, and compensator frequency response. Nonlinear gain is applied to improve the time-domain-transient performance, and the controller coefficients are loaded into the UCD9240 device registers.

Fig. 30 shows the window that receives the parameters of the two-phase synchronous buck converter. To obtain the small-signal model of the power stage, the GUI models the two-phase synchronous buck converter as an equivalent one-phase synchronous converter with the inductor, $L = L_1 \times L_1/(L_{1+}L_2)$; inductor series resistance, $R_L = R_{L1}||R_{L2}$; and phase-capacitor equivalent-series resistance, $R_{esr} = R_{esr1}||R_{esr2}$. The GUI computes the control-to-output-voltage transfer function in a manner similar to that described. The GUI can configure the sampling point of the eADC as a function of the phase of the switching period. The GUI applies the total delay, from the eADC sample point to the falling edge of the DPWM, to the loop-gain response. The user can enter the s-domain compensator parameters by observing the frequency response of the plant (power stage).



Plant - Rail #1

Plant Parameters

Vin:	10.000	V	Vout:	1.000	V
Fs:	350.0	kHz	Iout:	2.00	A
L:	0.363	µH	DCR:	2.200	mΩ
Rds-on-hi:	6.10	mΩ	Rds-on-lo:	2.90	mΩ

About Phases

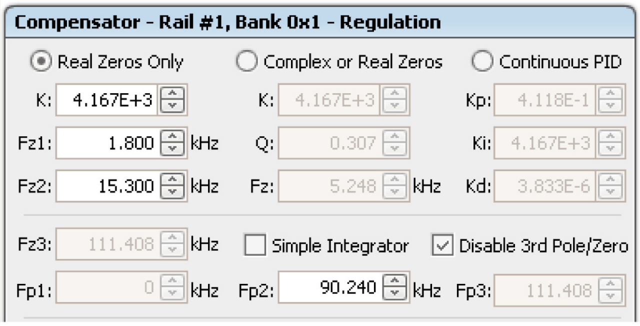
Number of Phases: 2

For DCR, L, and Rds-on-hi/lo, enter the actual component value and rating from the design. You can set the number of phases to account for light load operation or phase shedding, up to 2 phases. The evaluation of the plant will calculate an effective value for each component.

Capacitor Legs

	C (µF)	ESR (mΩ)	ESL (nH)	# Legs
	470.000	10.000	4.000	3
>	47.000	1.000	1.000	12

Fig. 30. Power-stage parameters window.



Compensator - Rail #1, Bank 0x1 - Regulation

● Real Zeros Only ○ Complex or Real Zeros ○ Continuous PID

K:	4.167E+3		K:	4.167E+3		Kp:	4.118E-1	
Fz1:	1.800	kHz	Q:	0.307		Ki:	4.167E+3	
Fz2:	15.300	kHz	Fz:	5.248	kHz	Kd:	3.833E-6	
Fz3:	111.408	kHz	□ Simple Integrator			☑ Disable 3rd Pole/Zero		
Fp1:	0	kHz	Fp2:	90.240	kHz	Fp3:	111.408	

Fig. 31. Compensator design window.

Fig. 31 shows the compensator design window. The GUI accepts the compensator coefficients in various forms, including real zeros, complex zeros, or PID coefficients. The user interface calculates the z-domain compensator-transfer function using a bilinear mapping technique. The resulting z-domain transfer function is then applied to the plant transfer function to provide the loop gain. The UCD9240 implements bit formatting and arithmetic in such a manner that the equivalent gain of the eADC and DPWM equals unity (neglecting the quantization effect on the gain). Therefore, the gain factors for the loop gain are the control-to-output-voltage transfer function of the power stage, G_{vd}; the output-voltage sense, H; and the compensator, G_c.

Fig. 32 shows the loop gain and the control-to-output-voltage frequency response. As shown in Fig. 31, the GUI also provides the frequency response of the open-loop output impedance, Z_{out_op}, and the closed-loop output impedance, Z_{out_cl}. Fig. 33 shows that the magnitude of Z_{out_cl} significantly decreases as compared to Z_{out_op} at the output-filter corner frequency. By observing the loop gain and Z_{out} frequency response, the user can fine-tune the compensator to provide optimal results. The user can also apply the autotuning technique described in Section VIII.B to obtain the desired frequency and time-domain response.

Fig. 34 shows the GUI window settings for the nonlinear-gain block. These nonlinear gains improve the transient performance. The threshold values are defined based on the range of error, e. These values have a a 6-bit signed-integer format and lie between –32 to 31. The gain blocks use a 4.2-bit format where a value of 1 indicates unity gain. For the power stage and controller mentioned earlier, the nonlinear gain and threshold values improve transient performance while maintaining stability.

To verify the performance of the designed controller, the user interface provides a time-domain load-transient simulation. In this part of the GUI, the user specifies the range and rate of the load step and the frequency of the load transient.

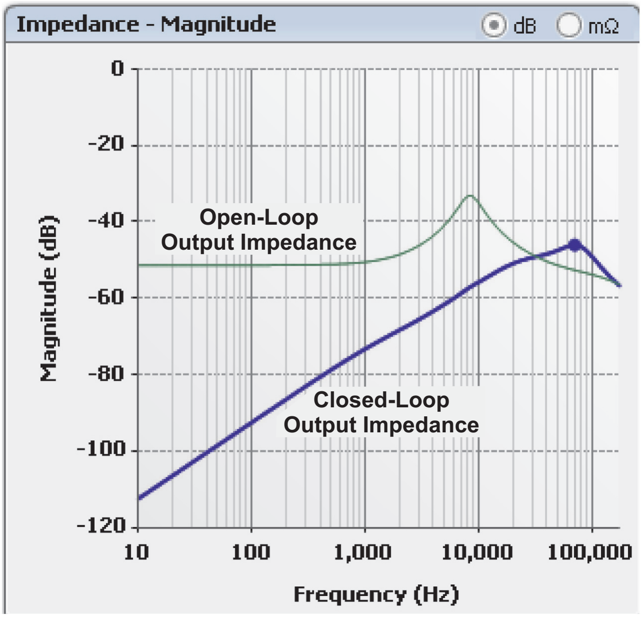


Fig. 33. Frequency response of open-loop and closed-loop output impedance.

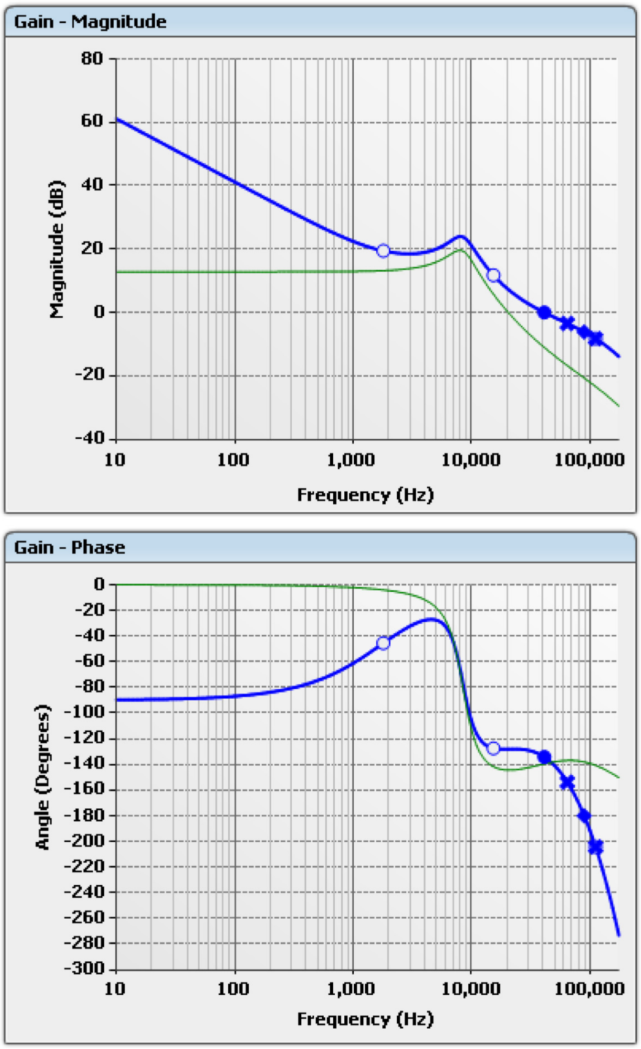
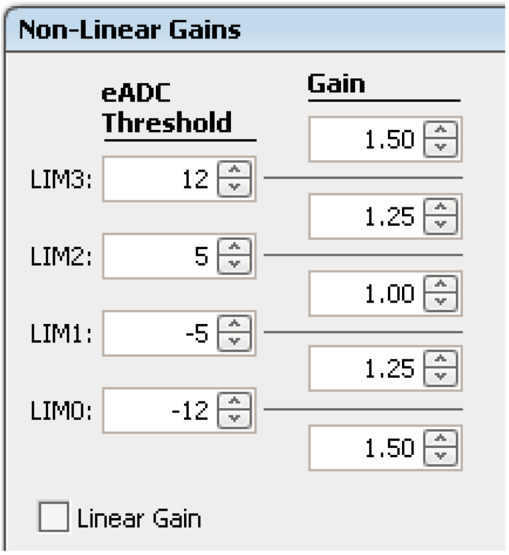


Fig. 32. Loop gain and control-to-output-voltage frequency response.



Non-Linear Gains

eADC Threshold

Gain

1.50

LIM3: 12

1.25

LIM2: 5

1.00

LIM1: -5

1.25

LIM0: -12

1.50

☐ Linear Gain

Fig. 34. Settings for the nonlinear-gain block.

The output voltage and inductor current are then computed based on the power-stage parameters, and the compensator coefficients are found with the discrete differential equations. Fig. 35 shows waveforms where a load transient of 50 A to 25 A with a 5-A/µs rate was applied to the output. The top waveform shows the value of $v_o - V_{ref}$. The lower waveforms show the inductor currents of the two phases.

C. Experimental Results

This section describes experimental results based upon a two-phase synchronous buck converter implemented with the UCD9240. The compensator and nonlinear gains were based on the values provided in the previous section. A load transient of 25 to 50 A with a rate of 5 A/µs was applied to the output. A LeCroy scope set at 2 mV/div and 100 MS/s recorded the effect of load transients on the output voltage. A low-noise LeCroy probe measured the voltage. The sampled data was captured through the USB port of the scope and plotted with MATLAB. Fig. 36 shows the response to the step-down load transient. The output-voltage overshoot closely follows the results derived from the simulation. Fig. 37 shows the experimental response to the 25- to 50-A step-up load transient. The step-up load-transient response exhibits a smaller-than-expected undershoot (around 25 mV). However, the output voltage suffers from a longer settling time in the step-up load transient. This longer settling time compared to the simulation results is the effect of the load transient on the supply voltage, V_{in}. Fig. 35 shows the effect of the load transient on the input voltage (V_g – 10 V). Increasing the low-esr bypass input capacitance, C_{in}, improves the results.

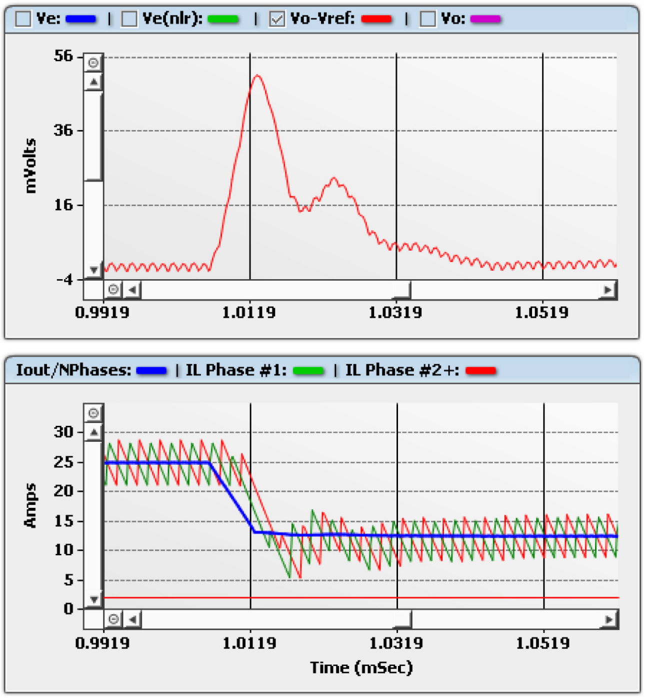


Fig. 35. Simulation results of a 50- to 25-A load transient for a synchronous two-phase buck converter.

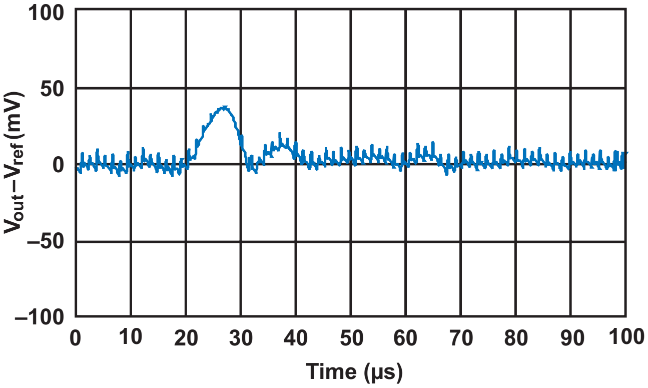


Fig. 36. The step-down load-transient response.

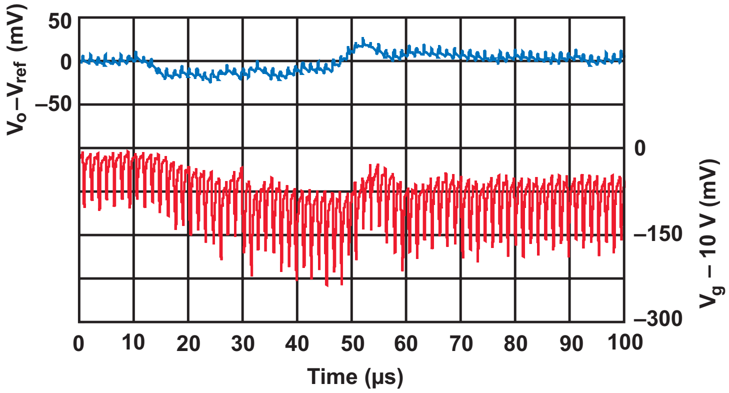


Fig. 37. The step-up load-transient response.

X. Conclusion

The development of a system model for a continuous-conduction, voltage-feedback buck regulator was reviewed. This model was augmented to include the effects of quantization error and computational delay, which are unique to digital control. This model was then used to show the onset of limit cycling due to error-voltage quantization and output-pulse-width quantization. The ability of a digitally controlled system to measure the transfer function of the closed-loop system without external test equipment was also presented. The system model was also used to identify (measure) the transfer function and how the programmability of the digital controller can be used to automatically tune the compensation for the feedback loop of a switch-mode power supply.

The above digital-control fuctions were implemented with the TI *Fusion Digital Power Designer* software. The software GUI simplifies the design steps required to develop a digitally controlled switching converter.

XI. Acknowledgements

The authors would like to thank our colleagues in TI's digital power group for their contributions to this article, particularly Michael Muegel for his work on the GUI. I would also like to thank Alan Hastings, whose advice greatly contributed to the quality of this seminar topic.

XII. References

[1] R. Erickson and D. Maksimovic, ["Chapter title," in] Fundamentals of Power Electronics, 2nd ed. Norwell, MA: Kluwer, 2000.

[2] A.R. Brown and R.D. Middlebrook, "Sampled-data modeling of switching regulators," Proc. IEEE Power Electron. Specialists Conf., pp. 349–369, 1981.

[3] L. Corradini and P. Matavelli, "Analysis of multiple sampling technique for digitally controlled DC-DC converters," Proc. IEEE Power Electron. Specialists Conf., pp. 1–6, June 2006.

[4] D. Maksimovic and R. Zane, "Small-signal discrete-time modeling of digitally controlled PWM converters," IEEE Trans. Power Electron., Vol. 22, No. 6, pp. 2552–2556, Nov. 2007.

[5] V. Yousefzadeh, M. Shirazi, and D. Maksimovic, "Minimum phase response in digitally controlled boost and flyback converters," Proc. IEEE Appl. Power Electron. Conf. and Exposition, pp. 865–870, 2007.

[6] W. Gu-Yeon and M. Horowitz, "A low power switching power supply for self-clocked systems," Int. Symp. Low Power Electron. and Design, 1996, pp. 313–317.

[7] "UCD9240 Digital Point of Load System Controller," UCD9240 Datasheet, TI Literature No. SLUS766

[8] "Digital Dual-Phase Synchronous Buck Controller," UCD9112 Datasheet, TI Literature No. SLVS711A

[9] B.J. Patella, A. Prodic, A. Zirger, and D. Maksimovic, "High frequency digital PWM controller IC for DC-DC converters," IEEE Trans. Power Electron., Vol. 18, No. 1, pp. 438–446, Jan. 2003.

[10] A. Syed, E. Ahmed, D. Maksimovic, and E. Alarcon, "Digital pulse width modulator architectures," Proc. IEEE Power Electron. Specialists Conf., Vol. 6, pp. 4689–4695, June 2004.

[11] A.P. Dancy, R. Amirtharajah, and A.P. Chandrakasan, "High efficiency multiple-output DC-DC conversion for low-voltage systems," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., Vol. 8, No. 3, pp. 252–263, June 2000.

[12] A.P. Dancy and A.P. Chandrakasan, "Ultra low power control circuits for PWM converters," Proc. IEEE Power Electron. Specialists Conf., Vol. 1, pp. 21–27, June 1997.

[13] V. Yousefzadeh, T. Takayama, and D. Maksimovic, "Hybrid DPWM with digital delay-locked loop," IEEE Comput. Power Electron., Troy, NY, 2006, pp. 142–148.

[14] A. Syed, E. Ahmed, and D. Maksimovic, "Digital PWM controller with feed-forward compensation," Proc. IEEE Appl. Power Electron. Conf. and Exposition, Vol. 1, pp. 60–66, Feb. 2004.

[15] A.V. Peterchev, Xiao Jinwen, and S.R. Sanders, "Architecture and IC implementation of a digital VRM controller," IEEE Trans. Power Electron., Vol. 18, No. 1, Part 2, pp. 356–364, Jan. 2003.

[16] A.V. Peterchev and S.R. Sanders, "Quantization resolution and limit cycling in digitally controlled PWM converters," IEEE Trans. Power Electron., Vol. 18, No. 1, pp. 301–308, Jan. 2003.

[17] H. Peng, A. Prodic, E. Alarcon, and D. Maksimovic, "Modeling of quantization effects in digitally controlled DC-DC converters," IEEE Trans. Power Electron., Vol. 22, No.1, pp. 208–215, Jan. 2007.

[18] B. Miao, R. Zane, and D. Maksimovic, "System identification of power converters with digital control through cross-correlation methods," IEEE Trans. Power Electron., Vol. 20, No. 5, pp. 1093–1099, Sept. 2005.

[19] B. Johansson and M. Lenells, "Possibilities of obtaining small-signal models of DC-to-DC power converters by means of system identification," Proc. Telecommun. Energy Conf., 2000, pp. 65–75.

[20] M. Hagen, "In situ transfer function analysis, measurement of system dynamics in a digital power supply," Digital Power Forum, 2006.

[21] M. Shirazi, R. Zane, D. Maksimovic, L. Corradini, and P. Mattavelli, "Autotuning techniques for digitally-controlled point-of-load converters with wide range of capacitive loads," Proc. IEEE Appl. Power Electron. Conf. and Exposition, pp. 14–20, 2007.

[22] W. Stefanutti, P. Mattavelli, S. Saggini, and M. Ghioni, "Autotuning of digitally controlled buck converters based on relay feedback," Proc. IEEE Power Electron. Specialists Conf., June 2005.

[23] Texas Instruments. Digital power software: Fusion digital power designer. [Online]. Available: http://focus.ti.com/docs/toolsw/folders/print/fusion_digital_power_designer.html

[24] "Integrated driver – MOS FET (DrMOS)," Renesas, R2J20602NP Datasheet, REJ03G1480-0200, Nov. 30, 2007, http://documentation.renesas.com/eng/products/transistor/rej03g1480_r2j20602npds.pdf

[25] V. Yousefzadeh and D. Maksimovic, "Sensorless optimization of dead times in DC-DC converters with synchronous rectifiers," IEEE Trans. Power Electron., Vol. 21, No. 4, pp. 994–1002, July 2006.