Digital Power-Conversion for the Analog Engineer

By
Bryan Kris
Staff Architect, Architecture & Applications
Digital Signal Controller Division
Microchip Technology Inc.

It is no secret that, in the past, analog engineers have struggled with complexity when designing power supplies that require multiple outputs, coordinated load sharing, hot-swap capability, or extensive fault handling. Using analog circuitry to implement system control functions is not always cost-effective or flexible.

Designing power supplies using analog techniques requires components to be “oversized” to compensate for component variation and component drift. Even after overcoming these design challenges, the power supplies need manual tuning at the end of the manufacturing line. So, what options do analog engineers have for designing power supplies? The engineering answer to this problem lies in being able to implement intelligent digital control of the power-conversion feedback loop.

Recently, microcontrollers have enabled analog designers to implement monitoring, control, communications and even deterministic functions, such as power sequencing, soft start and topology control in power supplies. However, digital control of the complete power-conversion loop has been impractical, due to the lack of cost-effective, high-performance technology.

DSCs in Switch-Mode Power Supply Design

A new breed of digital signal controllers (DSCs) is now making digital conversion possible with features such as intelligent power peripherals that combine counter-based pulse-width-modulation (PWM) modules, analog comparator-based feedback and coordinated analog-to-digital converter (ADC) sampling, coupled with fast multiplication in a single clock cycle. The combination of these features allows DSCs to handle the high execution rates needed for control-loop software.

Before starting a power-supply design, the following decisions need to be made:
1. Choose a topology that fits the application’s needs:
   a. Step-up or -down (Boost or Buck)
   b. Isolated (Forward, Half or Full-Bridge)
2. Select a switching technique:
   a. Hard
   b. Soft-switching. Soft-switching techniques such as Resonant Mode or Quasi-Resonant Mode, offer lower switching losses at the expense of more complex circuitry and control.
3. Select a Control Methodology:
   a. Voltage-mode
   b. Current-mode

Voltage-mode control and current-mode control are the two methods of control based on classic analog switch-mode power supply (SMPS) control techniques. In voltage-mode, the difference between desired and actual output voltage (error) controls the time that the supply voltage is applied across the inductor, which indirectly controls current flow in the inductor.

In current-mode control, the difference between the desired and actual output voltage (error) creates a threshold for an analog comparator to set the peak inductor current, in an effort to control the average inductor current.

Voltage-mode can provide more stability in a noisy environment or over a wide operating range. Current-mode control offers cycle-by-cycle current limiting and faster transient response. It also prevents “ratcheting-up the inductor current,” which can result in inductor saturation and lead to catastrophic MOSFET failure.

4. Select a PWM operating frequency. A higher-frequency PWM enables the use of smaller inductors and capacitors, but at the cost of additional switching losses.

5. Determine the required control bandwidth. This is largely determined by the expected load transients for the application.

6. With an estimated control bandwidth requirement, the next step is to allocate processor resources. There are a wide variety of control algorithms, but a common technique is the Proportional, Integral, and Differential (PID) method. Using the common PID algorithm, the control-loop will need to operate at a rate of approximately eight times the required control system bandwidth to ensure sufficient phase margin. When estimating the control-loop delays, all delays within the control-loop must be considered (see “Computing Control-loop Delay” section).

The next step is to choose a DSC that meets all or most of your chosen design requirements.
To demonstrate a digital SMPS design, we have chosen to design a synchronous buck converter based on one of Microchip Technology’s SMPS dsPIC® DSCs, the dsPIC30F2020. This DSC features a hard-switching, voltage-control mode that offers a complementary PWM mode. This buck converter (see Figure 1) uses synchronous switching, and a MOSFET replaces the rectifier in the circuit because it has a much lower forward-voltage drop than the standard rectifier. By lowering the voltage drop, the overall efficiency for the buck converter can be improved by as much as 5 to 10 percent. Synchronous switching with Q2 requires a second PWM signal that is the complement of the primary PWM signal. Q2 is on when Q1 is off, and vice-versa. In addition, “dead band” control is required to prevent both Q1 and Q2 from conducting at the same time during the rise and fall edges of the PWM signals.

The input to output voltage relationship in a Buck converter is represented by:

\[ V_{OUT} = V_{IN} \cdot D, \quad \text{where } D = \text{PWM duty cycle} = \frac{T_{ON}}{T_{ON} + T_{OFF}} \]

The ideal output voltage for a Buck converter is the product of the input voltage multiplied by the duty cycle of the transistor. By inspection (see Figure 1), the output voltage will equal the input voltage, if transistor Q1 is always turned on. If Q1 is always off, then the output voltage will be zero. In reality, there are voltage drops across the transistor and the inductor that increase when the load current rises.

We can now begin to design a digital SMPS control system using DSCs (see Figure 2).
In this typical SMPS control system for a synchronous buck converter, note carefully that there are delays associated with each block in this diagram.

The sample-and-hold (S & H) circuit typically samples every 2 to 10 microseconds. The ADC requires about 500 nanoseconds (ns) to convert the analog feedback signal to a digital value.

The PID controller is a program running on a DSC with a computation delay of about 1 to 2 microseconds. The controller output is converted to a PWM signal, which drives the switching circuitry. The PWM generator can introduce significant delays if it cannot immediately update its output when given a new duty cycle. The transistor drivers and the associated transistors also introduce delays from approximately 50 ns to about 1 microsecond, depending upon the devices used and the circuit design.

**Computing Control-Loop Delay**

Total control-loop delay is the sum of the ADC sample and convert time (500 ns), the PID calculation time (1 µsec), the PWM output delay (0), the transistor switching time (50 ns) and the PID execution-rate period (2 µsec). The total loop delay in this example is 3.65 µsec, which implies the maximum effective control-loop sampling rate of 274 kHz.

While the Nyquist Theorem requires a 2x sampling rate to reconstruct a signal, digital control loops must sample at a 6x to 10x rate. The reason for this is that, with only a 2x sampling rate, the phase-lag is 180 degrees. With a 2x sampling rate, we have already used up our “budget” of 180 degrees for phase-lag without considering any other delays in the system. A system with an 8x sampling rate introduces 45 degrees of phase-lag from the sampling process alone -- a much better sampling rate. To maximize phase margin, many digital control systems oversample the analog signals by 10x or more. Assuming a maximum effective sample rate of 274 kHz, the effective control bandwidth is one eighth of this, or approximately 34 kHz.
**Importance of PWM in SMPS Design**

Different power-supply specifications drive the need for different power-supply topologies. These different topologies require different PWM modes, which each support a wide variety of SMPS designs including standard, complementary, push-pull, multi-phase, variable-phase, current-reset and current-limit PWM modes.

The most basic PWM mode is the standard edge-aligned PWM, where the ratio of the on time versus the off time controls the flow of power. Only one of the PWM outputs per output pair is used in these Asynchronous Buck, Boost and Flyback converter circuits.

Our Synchronous Buck Converter uses a complementary PWM mode, where the complementary output controls a “synchronous switching” rectifier that is implemented with a MOSFET, instead of the typical rectifier. The complementary PWM mode can also be used in other circuits that use synchronous rectification to improve system efficiency.

Push-pull converters are commonly used in DC/DC converters and AC/DC power supplies. The term “multiphase PWM” describes multiple PWM outputs that are not edge-aligned. Multiphase converter circuits are often used in DC/DC converters that must supply high current in applications where the load may change very rapidly.

Phase-shifted PWM mode is becoming more common because of its wide use in personal computer power supplies. Microchip’s dsPIC DSC SMPS family supports all of the known PWM modes currently in wide use by the power-supply industry.

**Understanding PWM Resolution**

Power-supply designers and customers must correctly understand the term “PWM resolution.” PWM resolution is NOT how wide a particular counter is, but how many counts (the minimum possible PWM time slices) that can occur within a PWM cycle period. In the power-supply industry, PWM resolution is specified by the smallest time increment achievable within the PWM duty cycle. This resolution is often specified in ns.

If a PWM module does not have enough resolution, the control system (hardware or software) will dither the PWM outputs to achieve the desired average output. In power-supply applications, PWM dithering can create problems with ripple currents, and cause the control to enter a bad mode of operation called “Limit Cycling.”
For example, consider a PWM that can output values of 3 and 4 while the control-loop needs an output value of 3.25. In this case, the PWM dithers between the values 33343334. This is easily put into perspective – many DSCs offer PWM counters that operate in a range from 40 to 150 MHz, yielding PWM resolutions of 6 to 25 ns. The SMPS dsPIC DSC family features a duty cycle resolution of 1 ns.

In a control-loop, the time from acquisition of voltage and current measurements until the PWM outputs a new duty cycle value is called “latency.” As latency decreases, the control-loop becomes more stable and responsive. Some DSCs feature PWM modules that accept new duty-cycle data only on a PWM cycle-to-cycle basis. The time lag from the software computation of a new duty cycle value until the PWM module accepts the data increases the control-loop latency, while decreasing its stability. Therefore, it is best to use a DSC with a PWM module that immediately accepts and processes new duty-cycle data.

**SMPS ADC Requirements**

You can apply your analog insights to intelligent power-supply design using DSCs. The on-chip ADC provides the system status (feedback) to the control-loop. Conventional ADCs are designed with an assumption that the ADC values are collected and processed in a “group.” ADCs in audio processing and industrial control systems typically function in this manner. Group sampling causes the processor workload to peak in groups, and this increases control-loop latency.

Often in SMPS circuits, the analog signal to be sampled and converted does not exist, or may not be significant at all times. The signal may only be important at specific points in the PWM cycle. Standard ADC modules may, therefore, miss the desired data due to imprecise sample timing.
Figure 3: The importance of ADCs with individual sample-and-holds (S & Hs).

Figure 3, above, shows an example circuit where a current-sense resistor is used to monitor current flow. In this circuit, the current can be sensed only when the transistor is turned on. Typical ADC modules cannot accurately command the sample-and-hold circuit to take a sample at the appropriate time. If the application has multiple circuits to sense, then the ADCs are inadequate.

The ADC module onboard the SMPS dsPIC DSC provides independent sample-and-hold circuits that can sample independently of each other. It can therefore monitor voltages or currents at precise times, which enables event transitory signals to be sampled. This ability to can reduce system costs. In addition, the ADC onboard the SMPS dsPIC devices can sample asynchronously, enabling it to support multiple control loops operating at different frequencies, such as PFC (70 kHz) and DC/DC (250 kHz).

**Analog Comparators Improve Digital SMPS Designs**

As ADCs cannot continuously monitor signals, samples can only be processed up to the ADC’s mega samples per second (MSPS) rating. Some DSCs feature analog comparators that free-up the processor and ADC to perform other valuable tasks. For example, analog comparators can be used for current control in a manner similar to how traditional linear power-supply controllers directly control the PWM duty cycle. Analog comparators can also provide independent monitoring of over-voltage or over-current conditions.
Let us illustrate this important design issue with a practical example. The reference DACs and analog comparators on Microchip’s SMPS dsPIC DSCs can achieve latencies from current measurement to a PWM update of approximately 25 ns. Typically, the time when the analog voltage is sensed until the PWM output is modified by the comparator is about 25 ns. This response time is the quickest, when compared to other DSCs that must use “polling” techniques with their ADC and processor to modify the PWM outputs in response to changing conditions. In fact, this is how the DSC implements cycle-by-cycle current limiting, which is required for current mode control. Because the reference DAC tied to the analog comparator is also 16-bits, the same as the PWM resolution, the same control resolution is available for both voltage and current modes.

**PID Algorithm**

Using the PID algorithm, the proportional, integral and derivative errors of the actual versus the desired output voltage are combined to control the PWM duty cycle. The PID algorithm can be used in both voltage- and current-mode control loops.

Many analog engineers think that complex DSP programming skills are required to handle the DSP-like architecture involved with these DSCs. However, this is not the case. In fact, DSP skills are not needed to handle Microchip’s DSCs (see code listing in Figure 5, below). The central “core” of the control software (Figure 4) is the PID loop. The PID software is typically small -- one or two pages of code -- but its execution rate is very high, often hundreds of thousands of iterations per second. This high iteration rate requires the PID software routine be as efficient as possible to maximize performance. Using an assembler is a good method to insure “tight code”.

The PID control-loop is interrupt-driven by the ADC on a fixed-time basis. Any system function that can be executed in the “Idle Loop” should be, in order to reduce the unnecessary workload within the PID control software. Functions such as voltage ramp-up/down, error detection, feed-forward calculations and communication support routines are candidates for the idle loop. Any other interrupt-driven processes, such as communications, must be at a lower priority than the PID loop.
The idle loop is entered after completion of the system and peripheral initialization tasks. Typically, the idle loop monitors temperature, calculates the "feed-forward" terms, and checks for fault conditions.

The SMPS software implements the control algorithm, with the ADC interrupt-driven PID loop as its most time-critical portion. The PID software should use no more than approximately 66% of the available processor bandwidth, so that the remainder of the computing resources can be allocated to the idle loop software.

Assuming 30 MIPS operation with the PID loop (comprising 30 instructions), the execution time is approximately 1 µsec. If the iteration rate is 500 kHz (2 µsec), then the PID workload consumes one-half of the available processor bandwidth, or 15 MIPS.
CALCULATE_PID:
push.s ; Save SR and W0-W3
bclr.b IFS0+1, #3 ; Clr IRQ flag in interrupt controller
mov #PID_REG_BASE, w8 ; Init pointer to PID register block
mov #PID_GAIN_REG_BASE, w10 ; Init pointer to PID gain register block
mov ADBUF1, w0 ; Read ADC to get voltage measurement
mov COMMANDED_VOLTAGE, w1 ; Get commanded output voltage
sub w1, w0, w0 ; W0 = proportional voltage error
mov PROPORTIONAL_ERROR, w1 ; Get previous voltage error
sub w0, w1, w2 ; diff error = new verr - old verr
mov w0, PROPORTIONAL_ERROR ; Store New Proportional Voltage Error
mov w0, PREINTEGRAL_TERM ; Store copy PERR as pre integral term
mov w2, DERIVATIVE_ERROR ; Store new Derivative Error

; These registers are reserved for PID calculations
; w6, w7 = contains data for MAC operations
; w8, w10 = pointers to error terms, and gain coefficients

SUM_PID_TERMS:
cr A, [w8]+=2, w6, [w10]+=2, w7 ; clr A, prefetch w6, w7
mac w6\*w7, A, [w6]+=2, w6, [w10]+=2, w7 ; MAC proportional term and gain
mac w6\*w7, A, [w6]+=2, w6, [w10]+=2, w7 ; MAC derivative term and gain
mac w6\*w7, B, [w8]+=2, w6, [w10]+=2, w7 ; Update Integrator
add ACCA ; Add ACCB (Integrator) to ACCA
sftac A, -#8 ; scale accumulator (shift)
mov ACCAH,w0 ; Read MSW of acca (result)
bstt ACCAU,#7 ; Check sign bit of ACCA
bra z, OUTPUT_PWM ; Branch if acca PWM value is positive
cr w0 ; Clear negative PWM values

OUTPUT_PWM:
mov w0, DC1 ; Output new duty cycle value
pop.s ; Restore SR, w0-w3
retfie ; Return from Interrupt

Figure 5:
This code listing shows an example of the PID software for a digital-mode Buck converter. Although the PID software would not change for most SMPS topologies, the initialization code may need to be changed for the peripheral modules.

An evaluation board enables designers to test, modify the SMPS control software and understand SMPS design principles. In this case, you may consider Microchip’s dsPICDEM™ SMPS Buck Development Board (see Figure 6) -- a rugged, low-power, DC/DC Buck converter to evaluate the DSC device and control software. The board derives its input power by the standard AC/DC 9V, 0.75A power-supply. Two independent buck converters are available on this board, and the demonstration software is set-up to provide +5V and +3.3V output.
This board accepts an input voltage ranging from 8 – 14V DC. The output loads should be limited to 0.75A each, and input power can be supplied through either the coaxial input-power connector J2 or the test clip connections P1 and P2.

This board also provides a dynamic load for the +5V output. This load is driven by a 1 kHz square wave signal that is generated by the output compare module. The onboard dynamic load uses a FET to connect a resistor load to the converter output to ground and enables converter stimulation so that the converter’s dynamic behavior can be measured. The use of dynamic loads is user-selectable via jumper blocks and/or software.

Users can choose to operate this board as a standard buck converter or a synchronous buck converter via a jumper setting. The software supplied can operate the board in voltage control mode, and current monitoring is provided. The board also measures current via a sense-resistor on the output of the converter, with the voltage is amplified and presented to the ADC inputs on the dsPIC30F2020 device. The board also provides three uncommitted variable resistors that are readable via the ADC inputs. These "pots" can be used to simulate signals for prototyping purposes.
Conclusion

Power-supply designers can easily add new features and capabilities to their designs using the latest DSCs available for digital loop control. Complex digital signal processor (DSP) processing skills do not need to be learned in order to do this -- rather, using familiar analog components and software, designers can quickly and cost-effectively build more intelligence into their power supplies using DSCs.

Note: The Microchip name and logo, dsPIC, PIC, and MPLAB are registered trademarks of Microchip Technology Inc. in the USA and other countries. dsPICDEM is a trademark of Microchip Technology Inc. in the U.S.A. and other countries. All other trademarks mentioned herein are property of their respective companies.