Analog and Mixed-Signal Products

Analog Applications Journal

Third Quarter, 2004

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third–party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Mailing Address:

Texas Instruments
Post Office Box 655303
Dallas, Texas 75265

Contents

To view past issues of the
***Analog Applications Journal*, visit the Web site**
www.ti.com/sc/analogapps

Introduction

Analog Applications Journal is a collection of analog application articles designed to give readers a basic understanding of TI products and to provide simple but practical examples for typical applications. Written not only for design engineers but also for engineering managers, technicians, system designers and marketing and sales personnel, the book emphasizes general application concepts over lengthy mathematical analyses.

These applications are not intended as "how-to" instructions for specific circuits but as examples of how devices could be used to solve specific design requirements. Readers will find tutorial information as well as practical engineering solutions on components from the following categories:

- Data Acquisition
- Power Management
- Interface
- Amplifiers: Op Amps

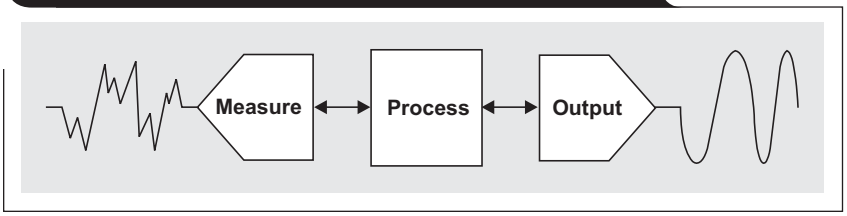
Where applicable, readers will also find software routines and program structures. Finally, *Analog Applications Journal* includes helpful hints and rules of thumb to guide readers in preparing for their design.

Streamlining the mixed-signal path with the signal-chain-on-chip MSP430F169

By Zack Albus (Email: j-albus@ti.com)
MSP430 Applications

With few exceptions, electronic measurement and control system designs incorporate varying degrees of three distinct functions: measurement of the environment, processing of data, and output back into the environment. This process can be anything from the measurement of thermistor resistance and display of temperature on an LCD to digital filtering of an analog signal. The common thread among many processes is that they often perform some level of digital conversion of an analog signal, computation and decision-making in the digital world, and conversion back into the analog domain. Figure 1 illustrates these elements that make up the analog signal chain.

Figure 1. Typical measurement and control system



Over the past decade, advancements in semiconductor integration have facilitated more robust combinations of digital processing and analog peripherals. While such devices are nothing new to today's standards, there are a number of trade-offs that must be evaluated to choose the level of integration most suitable for a given application. Key issues include system performance, size, and cost. As system complexity increases, increased integration can provide a smaller and lower-cost design capable of performance equal to or even better than that of the discrete alternative. Take, for instance, a typical discrete analog signal chain solution as compared to an integrated signal chain solution, both designed to solve the same monitor and control system function. An example for each system is shown in Figure 2.

In both systems, a variable resistance generates a voltage level that is sampled by the ADC. The conversion result is processed and used to determine the update rate of the DAC and, consequently, the analog output signal frequency. The output signal itself is a 12-bit sine wave and is made up of 16 steps or data points. While the analysis and results are based on the specific software implemented for each case, the same approach presented here can be used to determine CPU performance across any application.

Figure 2. Discrete system (top) vs. the integrated MSP430F169

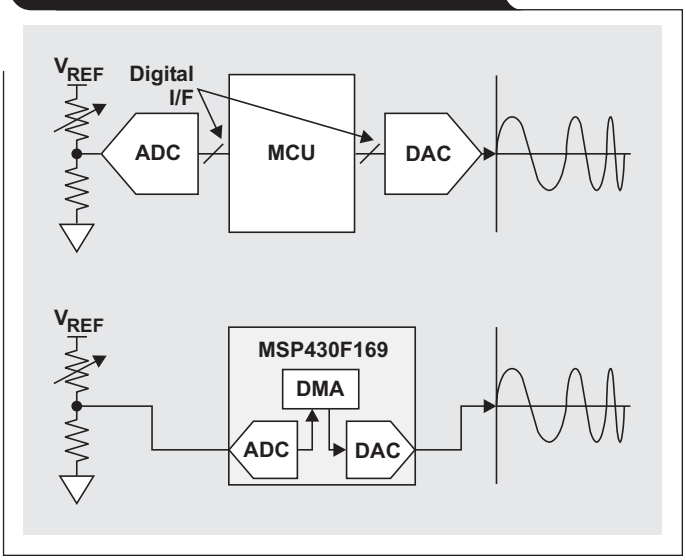
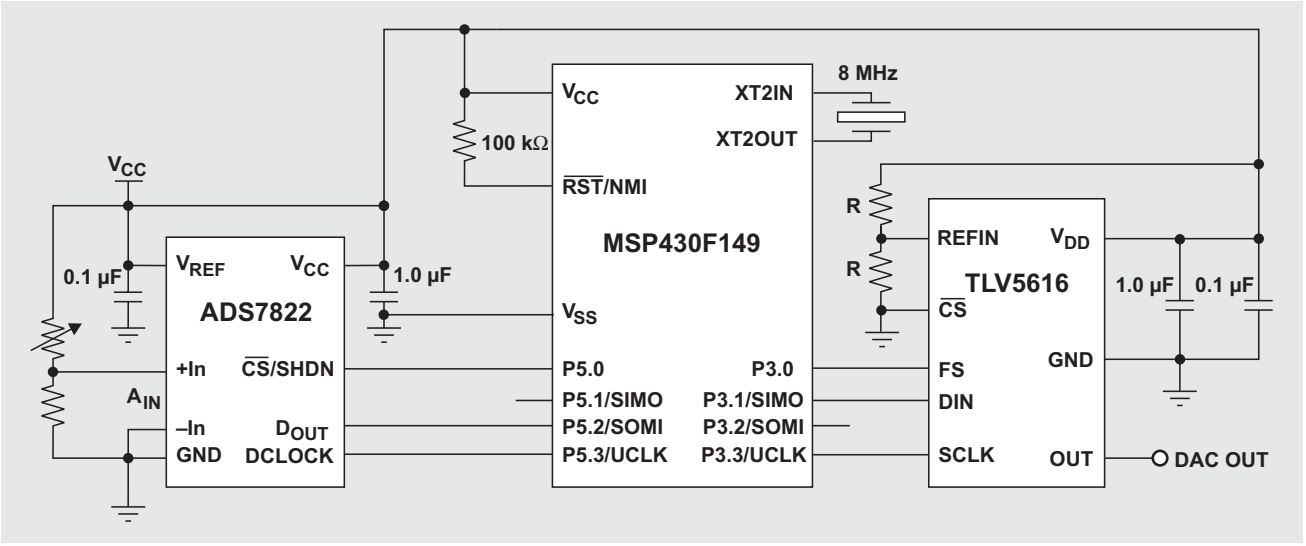


Figure 3. Discrete analog system

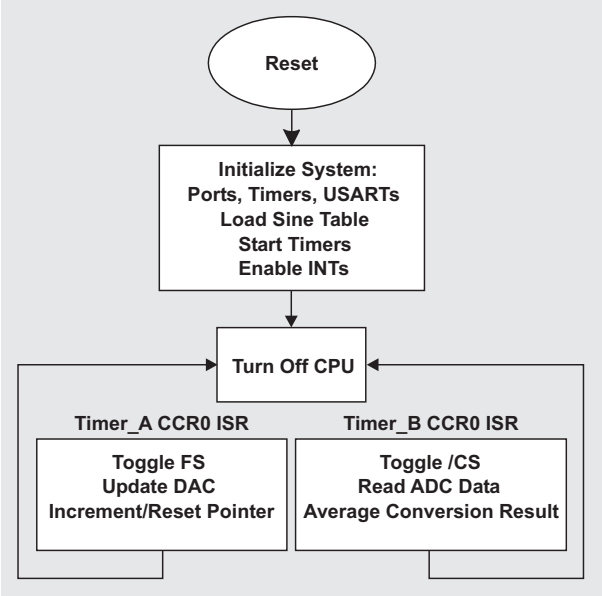


Case 1: Discrete peripheral system

The discrete signal chain solution is envisioned with the MSP430F149, an external ADC, an external DAC, and a serial interface to connect the system. The ADC measures and converts the variable voltage, and the result is used to process and determine the output frequency of a sine wave generated by the DAC. This discrete solution is detailed in Figure 3.

As shown, the external ADC and DAC are controlled via separate SPI interfaces of the microcontroller. Clocking of the MCU is performed at a maximum clock speed of 8 MHz and is input at the XT2 terminals. The external resonator clock source drives two 16-bit timers that provide individually controllable interrupt durations that initiate communication with each of the external peripherals. Sampling of the ADC occurs at a rate of 8 kSPS as set by Timer_B, and data is transferred from the ADC at a 1-MHz serial clock frequency via the hardware USART. Data loading of the external DAC occurs at a variable frequency as determined by Timer_A. The frequency at which Timer_A updates the DAC is determined by an 8-sample average of the ADC conversion result. Using a 16-point sine data table, this configuration provides an adjustable DAC-generated sine wave. Figure 4 shows the software flow for the discrete component system and represents common operation of an MCU using interrupt processing with multiple peripherals.

Figure 4. Discrete system software flow



To characterize the performance of the system, a loading analysis of the CPU is performed. Time spent by the CPU servicing the ADC during execution of the Timer_B interrupt service routine (ISR) requires a minimum of 60 CPU clock cycles (60 MCLKs). This includes toggling of the ADC /CS, receiving 16 bits of data, and averaging 8 conversion samples. The serial data transfer from ADC to MCU takes additional time during which the CPU must wait for the completion of each data byte transfer. The number of bits to be transferred and the ADC serial clock rate used determine this additional time.

The ISR executes in 60 MCLKs and requires an additional handling time of 11 MCLKs (the number of cycles needed to enter and exit each ISR). Every 8th ADC sample, the average result is calculated; and Timer_B CCR0 is updated, triggering the DAC ISR. The average and update operation adds 29 MCLKs to the ADC ISR every 8th time through the routine.

Maximum Timer_B ISR execution time (every 8th sample):

$$\frac{(60+29+11)}{8\ \text{MHz}}+\frac{16}{1\ \text{MHz}}=28.5\ \mu\text{s}$$

Average Timer_B ISR execution time (average time required to receive one conversion from the ADC):

$$\frac{(60+11)\times 87.5\% + (60+29+11)\times 12.5\%}{8\ \text{MHz}} + \frac{16}{1\ \text{MHz}} = 25.3\ \mu\text{s}$$

CPU loading required to complete the total ISR execution time can be estimated in terms of the average MCLK cycles required to service the ADC.

Timer_B ISR average MCLK cycles required:

$$25.3\ \mu\text{s} \times 8\ \text{MHz} = 202.4\ \text{MCLKs}$$

Given the 8-kSPS ADC sample rate, the CPU must enter the Timer_B ISR 8000 times per second, for a total of 1,619,200 MCLKs each second—a CPU loading of ~1.62 MIPS required to communicate with the ADC and handle data. This represents approximately 20% of the CPU's total available instruction execution time (8 MIPS maximum), leaving 6,380,800 CPU cycles to perform other MCU functions such as data transfer to the external DAC.

As mentioned earlier, the DAC is updated at a variable rate depending on the time between interrupts as determined by Timer_A. The DAC update interrupt interval is proportional to the value in Timer_A CCR0. The ISR requires a minimum of 51 MCLKs to complete, including toggling /FS, transferring 16 bits to the DAC, and servicing the sine table pointer.

Since the DAC cannot be updated faster than the Timer_A ISR can be executed, the maximum update rate, and consequently the maximum DAC output frequency achievable, is calculated based on the ISR time required to update the DAC.

In contrast to the time spent executing the ISR servicing the ADC, the CPU does not have to wait for all 16 data bits to be transferred to the DAC before exiting the ISR. After writing the least significant byte to the SPI transmit buffer, the hardware USART module handles transmission of the data to the DAC, allowing the CPU to continue instruction execution. The actual time spent updating the DAC is a combination of the time spent executing the ISR and the time required to complete data transmission to the external DAC. A complete DAC update cycle is shown in Figure 5.

To avoid any possible transmit data overrun, the SPI transmit buffer is tested after the DAC ISR is entered. If data is still being transmitted when the ISR is entered, the CPU will wait until the transfer is complete before sending the next sync pulse to the external DAC.

In addition to the 51 MCLKs to execute the ISR, an additional 4 clocks are required every 16th loop through the routine, which resets the sine table pointer to the first value in the array. This causes every 16th DAC update (or 6.25% of the DAC updates) to require a small additional execution time, which is taken into account when actual CPU loading is calculated. Also included are the 11 cycles required to enter and exit the ISR.

Average Timer_A ISR execution time:

$$\frac{(51+11)\times 93.75\% + (51+4+11)\times 6.25\%}{8\ \text{MHz}} \cong 7.78\ \mu\text{s} \approx 62.2\ \text{MCLKs}$$

Timer_A ISR CPU loading for a given f_{OUT}:

$$62.2\ \text{MCLKs} \times 16\ \text{points} \times f_{OUT} = \text{MIPS}$$

The maximum DAC frequency based on the measured time to transmit each 16-bit data word is given by:

$$\frac{8\ \text{MHz}}{62.2\ \text{MCLKs} \times 16\ \text{points}} \cong 8\ \text{kHz}$$

The maximum frequency represents a 100% loading of the CPU. In reality this is not feasible, as CPU time must also be spent servicing the ADC. The maximum frequency achievable by the DAC is more correctly established when the required ISR execution time needed to sample the external ADC is taken into account. This is a critical requirement if distortion cannot be tolerated in the given application. Figure 6 shows the effect on the DAC output as the update frequency increases beyond the speed of the Timer_B ISR sampling the ADC.

Figure 5. External DAC update (Timer_A ISR) cycle time

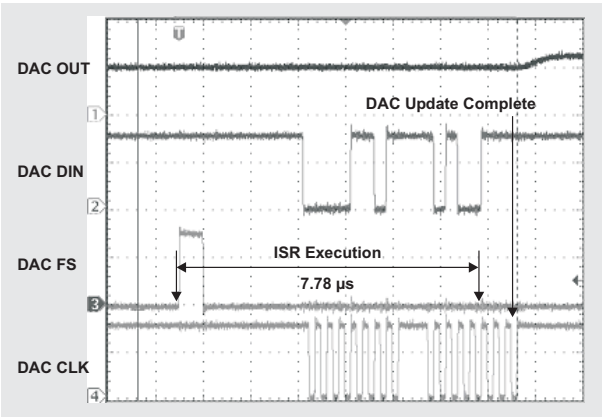
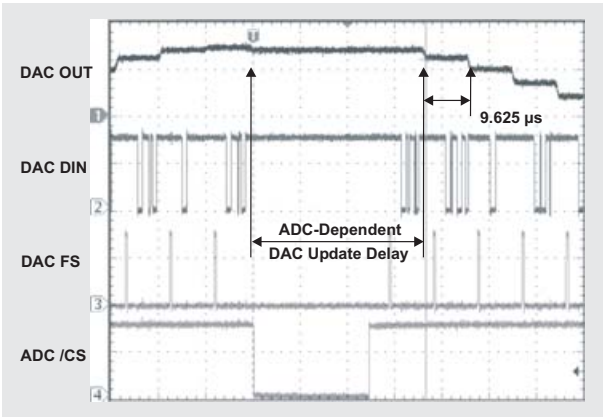


Figure 6. Sine distortion due to ADC ISR service delay



The desired DAC update frequency, or DAC ISR entry frequency, is established by the value in Timer_A CCR0, which for this example is 0x004C = 76 counts. The actual time between DAC ISR entries is 77 counts and includes one additional timer count occurring when the timer rolls from 76 to 0. For the example Timer_A CCR0 value with an 8-MHz Timer_A clock, the expected time delay between DAC updates is 9.625 µs as shown in Figure 6. However, when the ADC is sampled, a much longer delay is incurred; and the shape of the desired sine wave is affected. This effect can be eliminated by establishing a maximum update rate of the DAC and can be estimated by the total time required to execute each ISR. Keep in mind that this is the maximum time, not the average. The total ADC and DAC transfer time is 36.75 µs, providing a maximum DAC output frequency of ~1.7 kHz. The degree to which distortion can be allowed in the DAC output and the methods used to reduce it are application-specific and will determine the maximum acceptable output frequency.

Each complete sine output cycle at this frequency requires 1700 SPS × 16 data points to be transferred to the external DAC each second. With 62.2 MCLKs required to complete each DAC update, the CPU loading due to the DAC ISR at the maximum f_{OUT} is given by:

$$1.7\ \text{kHz} \times 16\ \text{points} \times 62.2\ \text{MCLKs} \cong 1.7\ \text{MIPS}$$

This represents a maximum loading of approximately 21% of the total available instruction throughput of the CPU. As the DAC update frequency is reduced, the CPU loading percentage will decrease accordingly. However, when CPU bandwidth is determined for any application, the maximum potential loading must be taken into account to ensure desired system performance over the entire operational range of the system.

Total CPU loading to service both the external ADC and DAC comes to slightly over 3.32 MIPS, leaving 58% of the CPU's available time to perform additional tasks. With the exception of calculating the ADC average and servicing the sine table pointer, the CPU spends this time doing nothing more than transferring data between the external components of the system. To eliminate the impact of data handling on the CPU, the communication between each element of the system must be streamlined. Integration of the analog functions in the system with the MCU is the step required to achieve this efficiency.

Case 2: Integrated peripheral system

By utilizing the highly integrated analog and digital functionality of the MSP430F169, the system described in the previous section is realized completely with a single-chip silicon solution. Figure 7 shows the integrated MCU system.

Integration of the ADC and DAC functions on-chip with the MCU greatly simplifies the system design in comparison to the discrete case. The serial communication protocols to the ADC and DAC have been removed from the CPU and off-loaded to the ADC and DAC internal modules. The ADC conversion averaging function is still performed by the CPU; but, in the case of the DAC, data transfer and pointer handling are entirely performed by the on-chip DMA module. DAC output frequency adjustment is made by interrupting the DMA instead of the CPU, freeing up resources for other tasks. The software flow for the integrated approach is shown in Figure 8.

The integrated solution uses Timer_B to establish the on-chip ADC12 sample/convert trigger for an ADC sample rate of 8 kSPS. This is given by Timer_B CCR0, which is set to 0x03E7 or 999. Timer_B is clocked with SMCLK = 8 MHz as in the external peripheral case. The ADC12 module is configured to perform a repeat conversion on a single channel: A0. Every 1000 Timer_B counts, or 0.125 ms, the ADC12 is triggered and performs a sample/convert of A0,

Figure 7. Integrated analog system

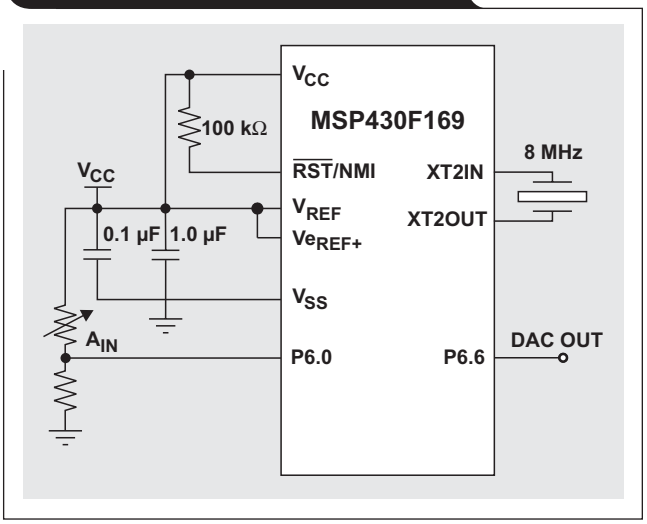
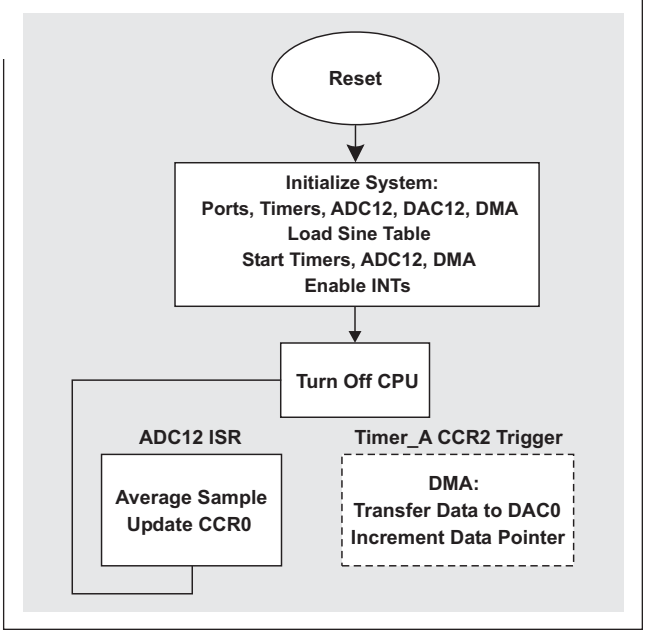


Figure 8. Integrated system software flow



stores the conversion result in the ADC12MEM0 register, and generates an interrupt.

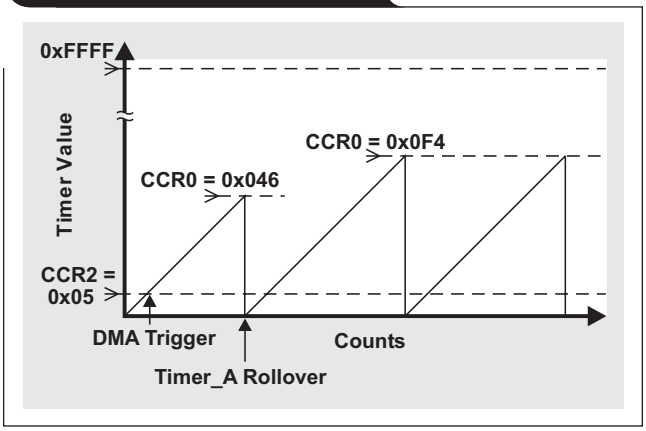
The ADC12 ISR requires a maximum of 50 MCLKs to complete. This is for every 8th interrupt, which includes the averaging of the conversion result. (ISR execution durations for samples 1 through 7 require only 21 MCLKs.) The averaged result is then moved to Timer_A CCR0, which defines the timer count between triggers to the DMA loading data to channel 0 of the DAC12 module. The total MIPS required to perform the ADC conversion handling and averaging is given in the following equation.

On-chip ADC ISR MCLKs per second:

$$\frac{(21+11)\times 7000+(50+11)\times 1000}{1{,}000{,}000}\approx 0.29 \text{ MIPS}$$

Total CPU loading for the ADC12 ISR is 3.6% of the total available CPU—a reduction of greater than five times the CPU loading as compared to the external ADC scenario. While this reduction is impressive, more compelling is the increased performance achieved by bringing the DAC on-chip. Along with the DMA available on the MSP430F169, DAC updating can be performed completely transparent to the CPU.

Figure 9. Timer_A operation



The DAC12 module is configured in 12-bit, unsigned binary mode. As defined earlier, Timer_A is clocked by SMCLK and is set to count up to CCR0 as determined by the ADC12 conversion result average. The DMA trigger moving data from the sine table to the DAC is issued when the value in Timer_A CCR2 is reached by Timer_A. Timer_A CCR2 need not be changed and can range in value from 0 to Timer_A CCR0. When Timer_A crosses the value in Timer_A CCR2, a trigger is sent to DMA channel 0. Since Timer_A CCR0 will be continuously modified to change the DAC update rate, Timer_A CCR2 should be assigned a value less than the smallest value to be written to Timer_A CCR0. The concept of updating the DAC via the DMA is shown in Figure 9.

Channel 0 of the DMA is configured to transfer data from RAM to DAC12_0 upon reception of a Timer_A CCR2 trigger. The source address for the DMA is the first location of the sine table data and is auto-incremented each time a value is moved to DAC12_0. Once all 16 data points are sent to the DAC, the DMA source address resets to the first value of the sine table and repeats. The number of data values transferred by the DMA before address reset is defined in DMA0SZ and is equal to the length of the sine data table.

Triggering of the DMA and the transfer of sine data to the DAC12 module is performed entirely transparent to the CPU. Although the CPU executes no instructions, each DMA transfer does require 2 MCLKs to complete. One MCLK cycle is required to retrieve the source data, and the second moves the data to the destination location. During this time, the CPU is halted for 2 cycles and the DMA transfers the required data. Effective CPU loading can be determined based on the required 2 MCLKs per DMA transfer.

For the same 1.7-kHz maximum DAC f_{OUT} in the external ADC/DAC example, a total of 54,400 MCLKs are required to perform an update cycle through the complete sine table via the DMA.

$$\frac{1.7 \text{ kHz}\times 2\times 16}{1{,}000{,}000}=0.055 \text{ MIPS}$$

This is 0.68% of the total available CPU throughput. Totaled with the time spent calculating the ADC12 conversion result average, the required total CPU loading for the system operating with a DAC output frequency of 1.7 kHz is 0.35 MIPS. This represents a factor of almost 10 times fewer instructions per second than that for the exact same system using external peripherals.

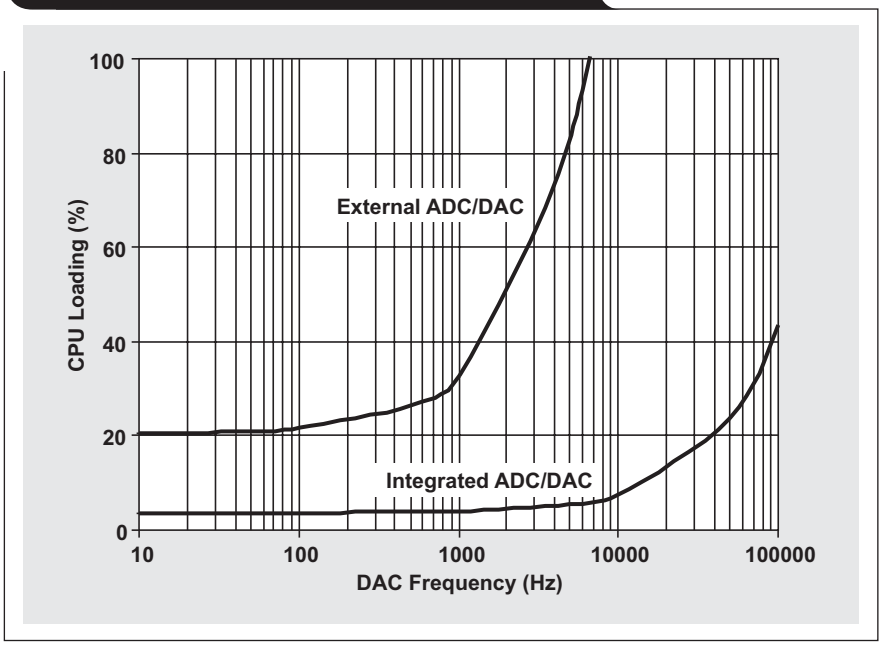
In addition to an increase in available CPU performance, the distortion effects of the sine wave output as shown for the external peripheral case are eliminated. Because the DMA handles 100% of the DAC12 updating duties, the CPU instruction execution time is not dependent on the DAC12_0 output frequency. The timing requirements for the external peripheral case relating to the DAC as a result of external ADC ISR handling are no longer valid and do not apply when determining the maximum DAC12 output frequency. If the update frequency of the DAC12 causes the CPU loading to exceed the remaining 7.65 MIPS available, only the ADC ISR handling will be affected, not the frequency of the DAC output.

As an example, channel 0 of the DAC12 module could theoretically be updated every 2 MCLKs. The maximum theoretical DAC output frequency for a 16-point sine wave is:

$$\frac{2 \text{ MCLKs}\times 16 \text{ points}}{8 \text{ MHz}}=4 \text{ µs or } 250 \text{ kHz}$$

However, triggering the DMA in single-word mode requires a minimum of 4 clock cycles to complete. To transfer the DAC data, 2 clocks are needed; and 2 clocks are required to synchronize the DMA transfer to the timing of the bus. The method of using Timer_A and the DMA to update the DAC12 yields a maximum DAC output frequency

Figure 10. CPU loading performance summary



of 125 kHz. CPU loading is effectively 100%, since this leaves no MCLKs for the CPU to execute instructions and, as a result, ADC12 ISR servicing cannot be executed by the CPU. A summary of CPU loading versus DAC frequency for the discrete and integrated system scenarios is shown in Figure 10.

The two different approaches discussed here are not intended to show an ideal implementation for a given system but rather to compare and contrast the effects of chip-level integration in the mixed-signal world. For any number of reasons, some real-world systems may not be able to take full advantage of such features; but, in general, efficient utilization of MCU on-chip peripherals greatly reduces CPU loading and can also allow for more flexible control of the mixed-signal chain. As has been shown, using an external ADC and DAC can require a significant amount of dedicated CPU cycles simply to move data to and from the data ports of each device. Integration of the analog blocks of a system has a solid impact on driving the total application to a higher level of performance.

Not only do on-chip peripherals facilitate a less complex system, but often a smaller and lower-cost solution can be realized through system integration. In this example, removal of the serial communication links between the ADC and DAC also increases total achievable system performance; and CPU resources that were required for data I/O can now be used for other system control and processing tasks.

The use of integrated analog peripherals is not always feasible in a given application and is ultimately at the discretion of the engineer leading the design. In cases where an MCU such as the MSP430F169 can be used, the benefits of integrated analog become a powerful tool in enabling complex signal-chain applications.

References

For more information related to this article, you can download an Acrobat Reader file at www-s.ti.com/sc/techlit/*litnumber* and replace *"litnumber"* with the **TI Lit. #** for the materials listed below.

Document Title	TI Lit. #
1. "MSP430x1xx Family," User's Guide	slau049
2. "MSP430x13x, MSP430x14x, MSP430x14x1 Mixed Signal Microcontroller," Data Sheet	slas272
3. "MSP430x15x, MSP430x16x, MSP430x161x Mixed Signal Microcontroller," Data Sheet	slas368
4. Mark Buccini, "Using SPI synchronous communication with data converters—interfacing the MSP430F149 and TLV5616," *Analog Applications Journal* (February 2001), pp. 7–10	slyt026
5. Mark Buccini, "Intelligent sensor system maximizes battery life: Interfacing the MSP430F123 Flash MCU, ADS7822, and TPS60311," *Analog Applications Journal* (1Q 2002), pp. 5–9	slyt029

Related Web sites

analog.ti.com
www.ti.com/sc/device/*partnumber*
Replace *partnumber* with ADS7822, MSP430F149, MSP430F169 or TLV5616

Tips for successful power-up of today's high-performance FPGAs

By Jeff Falin, *Applications, Power Management* (Email: j-falin1@ti.com),
and Landa Pham, *Marketing, Power Management* (Email: landa@ti.com)

Introduction

A major issue facing designers of FPGA-based systems is achieving a clean power-up. This article explains FPGA power requirements and the causes of their power-up problems. Then it provides a strategy for powering FPGAs and offers guidance on selecting the appropriate point-of-load dc/dc converter for each power rail.

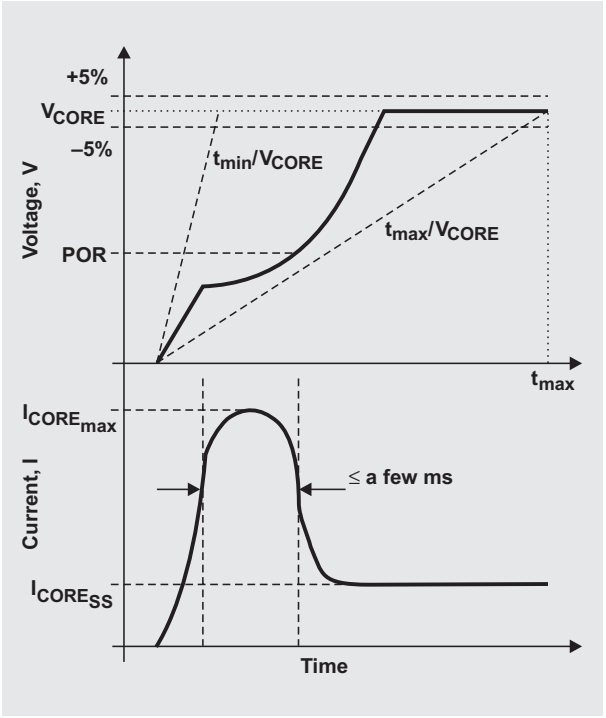
The requirements

Most FPGA manufacturers require the core voltage (V_{CORE}), and possibly the I/O (V_{IO}) or other voltage rails, to ramp up monotonically to a certain voltage with ±5% steady-state tolerance, at a rate between t_{min}/V_{CORE} and t_{max}/V_{CORE} as shown in Figure 1.

The primary reason for the monotonic rise is either an internal power-on-reset (POR) circuit or an external supervisory reset chip that is used to enable certain sensitive analog circuitry, such as a phase-lock loop (PLL) controlled by a voltage-controlled oscillator (VCO). For example, if the voltage were to rise above the POR threshold, then dip below the trip point (outside the POR's hysteresis range) and rise again, the VCO would change frequencies and/or phase, causing the PLL to fall out of phase and lose synchronization.

As shown in Figure 1, further complicating an FPGA power rail's monotonic ramp-up requirement is an inescapable surge current at startup. This surge current, $I_{CORE_{max}}$, which consists of capacitance charging current and the FPGA start-up current, can be much greater than the steady-state current, $I_{CORE_{SS}}$. Each power rail of an FPGA has a minimum amount of bypass/decoupling capacitance that is used to minimize voltage troughs during load transient steps. This decoupling capacitance is usually on the order of several hundred microfarads. Using $i_c = C\ dv/dt$, we can easily see that, depending on the speed at which the converter is attempting to ramp the rail voltage, the start-up current could be on the order of amperes. In addition, FPGA manufacturers specify minimum in-rush current requirements in the ampere range for some of their devices. These in-rush currents are needed not only to charge the capacitances of the millions of internal components of the FPGA but also to momentarily supply current through a low resistance path to ground created by, for example, stacked complementary transistors that are both on.

Figure 1. Voltage and current start-up profile



So, during startup, the power rail's point-of-load dc/dc converter must simultaneously be able to supply a large in-rush current and to maintain a monotonically ramping output voltage with a certain dv/dt. Therefore, in addition to other components it is powering, the converter's input power supply must be capable of supplying large load currents (load transients) for short periods of time. Since these types of power supplies with low source/output impedance and fast response times are usually expensive, more often a less expensive, lower-current, slower supply is chosen; and decoupling capacitors are added to its output to lower its output impedance. Unfortunately, simply adding more capacitance to such a supply can make its response time even worse and can further complicate the ramp time and surge-current issue.

The simplest dc/dc converter: The linear regulator

Interestingly, the simplest of dc/dc converters, the linear regulator, can cause the most problems during powering of the core or I/O voltage rails of FPGAs. Based on the setup shown in Figure 2, Figures 3 and 4 illustrate this point. In Figure 2, a typical bench power supply is followed by a linear regulator to power the V_{CCO} rail of an FPGA that has a specified minimum surge current of 1.5 A (including in-rush current due to charging capacitors).

Figure 3 shows the results of powering the FPGA from Figure 2 with a 3-A current-limited linear regulator and a 5-A current-limited lab supply.

One might assume that simply using a linear regulator with a higher current limit would solve the problem. However, Figure 4 shows that this is not the case when a 9-A current-limited linear regulator, the same 5-A current-limited lab supply, and an FPGA with a specified minimum surge current of 700 mA are used.

Like most linear regulators, these regulators turn on hard at startup and try to provide a regulated voltage within a few hundred microseconds. The regulators quickly reach their current limit and begin to operate like a constant-current source at that current limit. However, the current surge depletes the lab supply's output capacitor quicker than the supply can replenish it. The result is that the input rail falls below the linear regulator's undervoltage lockout (UVLO) circuit. This cycle could repeat indefinitely until the input power rail incrementally increases to a point where the linear regulator does not shut down. Surprisingly, one solution to this particular problem is to

Figure 2. Linear regulator powering an FPGA

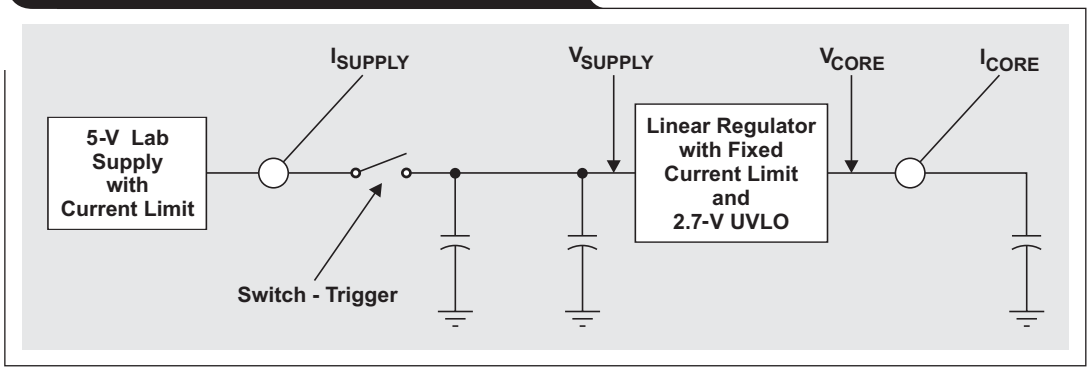


Figure 3. Start-up profile of a 3-A current-limited linear regulator

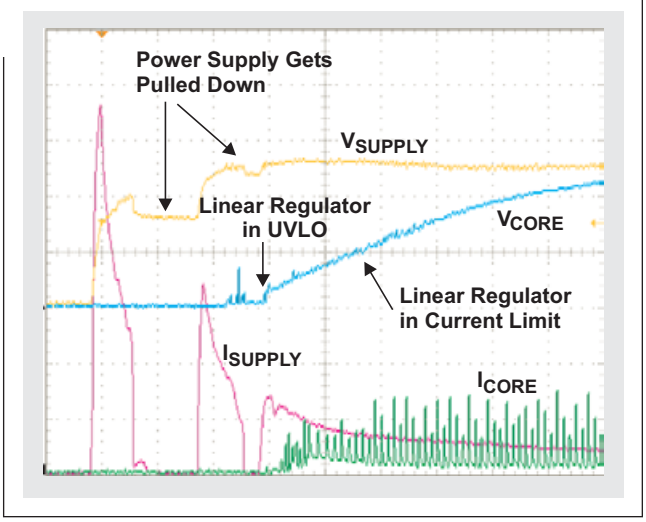
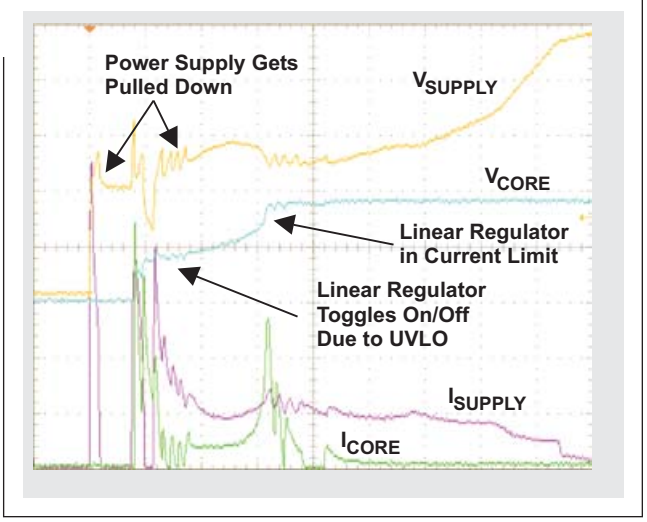


Figure 4. Start-up profile of a 9-A current-limited linear regulator



use a linear regulator with a lower current limit and possibly to increase the input power supply's decoupling capacitance. However, soft starting is the best solution.

Soft start to the rescue

System designers seem to have fewer FPGA start-up problems when using switching regulators instead of linear regulators (assuming that the switching regulator feedback loop has been accurately stabilized). Why is this? The primary reason is that almost all switching regulators have built-in soft-start circuitry that minimizes their input and therefore their output surge currents at startup while providing a monotonic voltage ramp. Figure 5 shows the results of powering the same FPGA from Figure 2 with a bench power supply having a dv/dt of 2 V/ms.

The FPGA in-rush current is significantly reduced when the rail voltage ramps slowly. Most FPGA datasheets specify a minimum and maximum power rail ramp-up time. Therefore, using a point-of-load converter solution that includes ramp-time control is the safest way to power an FPGA.

Why use sequencing?

Most FPGAs do not require sequencing of their power rails; however, FPGA datasheets may specify the amount of time the rails can tolerate a specific voltage difference. Some datasheets state that powering up one rail before the other will minimize in-rush current at startup. Regardless of an explicit requirement, sequentially ramping up power rails one after another, as shown in Figure 6, is recommended if for no other reason than to minimize the demand on the system power supply.

V_{CORE} is usually powered up before V_{IO}. Sequential sequencing is the easiest to implement with the output of the first rail, if the voltage is high enough, or with a supervisory circuit to control the enable pin of the second rail, and so on. Most power ICs have active-high enable signals, and many have integrated supervisory (sometimes called "Power Good" or "Power OK") signals.

In some cases it may be preferable to ramp up all rails simultaneously as shown in Figure 7. Simultaneous sequencing is theoretically the most ideal form of sequencing. It ensures short-term and long-term reliability for powering the IC, since all power rails are at the same voltage during startup; however, it typically requires external circuitry. Some power IC manufacturers are providing power ICs with integrated simultaneous start-up circuitry.

Figure 5. Bench supply powering an FPGA

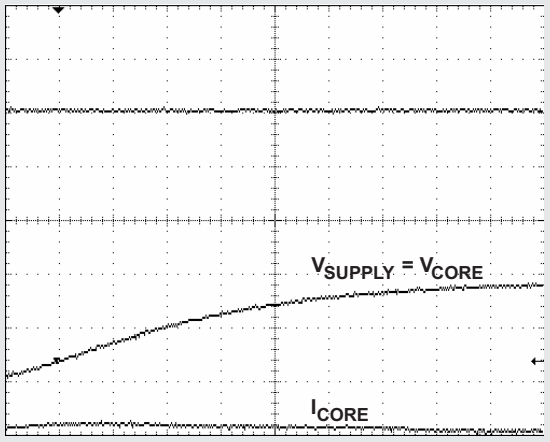


Figure 6. Sequential sequencing

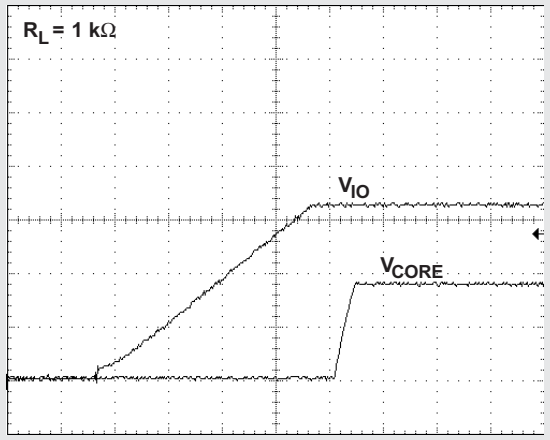
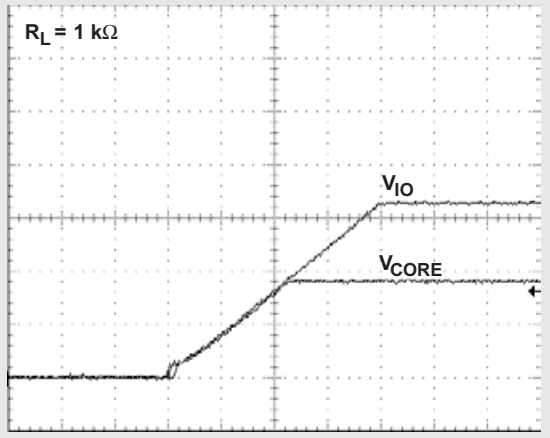


Figure 7. Simultaneous sequencing



Most reliable FPGA power-up strategy

Figure 8 shows a robust FPGA power management system that addresses the previously summarized power-up requirements and start-up issues. Incorporating both soft-start and sequential sequencing, this methodology is especially useful when the system power supply capabilities either are unknown or are being taxed by other components in the system.

The supply voltage supervisor (SVS) on the input rail prevents the V_{CORE} converter from turning on until the input rail is up and its input capacitors are fully charged. This reduces the chances of the in-rush current at startup from pulling down the input rail and tripping the UVLO of the V_{CORE} converter, as seen in Figure 3. The SVS on the V_{CORE} rail may be integrated into the converter. Some type of SVS circuitry is most likely needed since the V_{CORE} voltage is typically too low to drive the enable signal of the V_{IO1} supply directly. Soft starting one or more of the higher current rails further reduces the chances that start-up problems will occur.

The next design step is determining which point-of-load converter to use for each required rail. This decision depends on a number of factors, including the input supply voltage; the FPGA voltage rail and its tolerance; acceptable output voltage noise/ripple; the steady-state current; expected load transients; system size; and, of course, cost.

Which dc/dc converter?

The load current and the voltage difference between the input supply and FPGA power rail determine which dc/dc converter to use. Today's IC process node allows for FPGAs, like the Spartan®-3 line from Xilinx,® to operate with core voltages at 1.2 V or less. Depending on the configuration, they may require greater than 1 A of steady-state current. Meanwhile, input power supply rails have stayed at 3.3 or 5 V to power certain I/O peripherals. So, assuming that the system designer wants to use a low-cost linear regulator to derive the V_{CORE} voltage directly from a 5-V rail, the regulator needs to dissipate $(5 - 1.2)V \times 1\ A = 3.8\ W$. However, even the largest packaged linear regulator on the market cannot handle more than about 3 W without additional airflow or external heat-sinking. Therefore, step-down (buck) switching converters, whether with integrated FETs (TPS54610), with external FETs (TPS40003, TPS64203), or in module form (PTH05050), with their higher efficiency and therefore better power dissipation, are the best solution for high-current but low-value V_{CORE} voltages.

For lower-current V_{CORE} rails or V_{IO} above 2.0 V or so, linear regulators should still be considered for their advantages, including simplicity, small size, and low cost. Because very few linear regulators in today's market offer integrated soft start, they suffer from the surge-current problems described earlier. Adding the external circuitry shown in Figure 9 implements soft starting and eliminates these surge-current problems.

Figure 8. Robust FPGA power management methodology

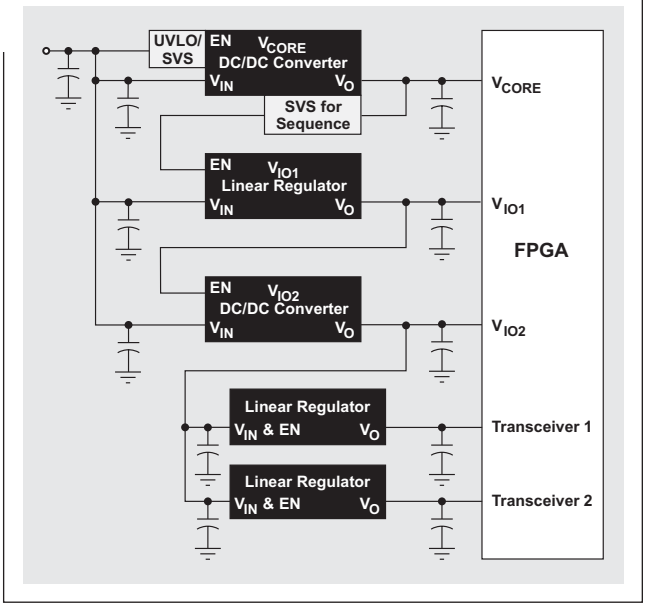
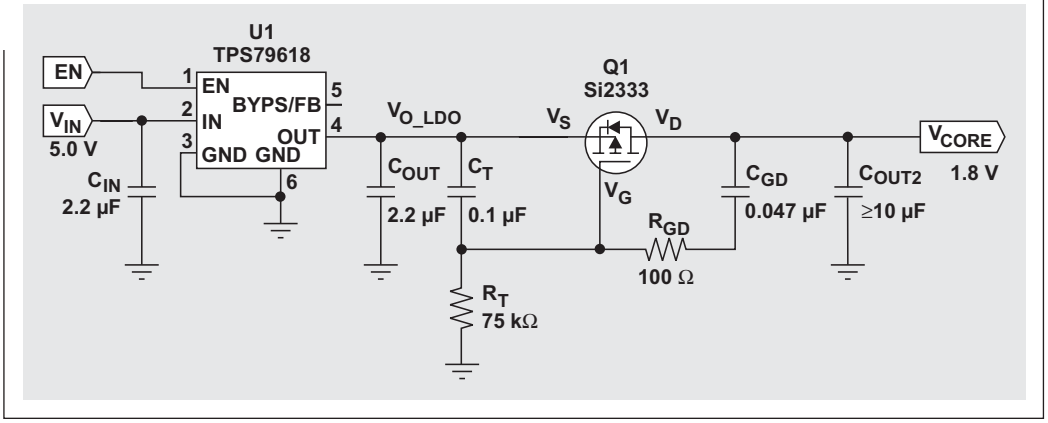


Figure 9. Linear regulator with a PMOS FET for soft start



In Figure 9, V_{CORE} drives the source of the PMOS FET. Choosing a FET with low on-resistance is critical to ensure that there is little voltage drop across the FET at maximum load current. See Reference 1 for further information.

Due to their low-noise performance, linear regulators are the recommended solution for powering analog rails such as PLL, Xilinx's Virtex-II Pro™ RocketIO™ transceiver, or Stratix™ GX transceiver supplies from Altera®. Since these rails are typically low-current, power dissipation is not an issue.

Decoupling capacitors

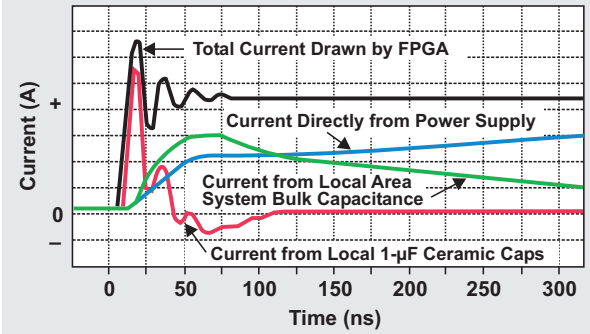
Since FPGAs are digital devices, they have potentially large load-current transient spikes. Most FPGA manufacturers provide guidance to the system designer in selecting the decoupling capacitors for each power rail to minimize load transient effects, such as the power rail being pulled below the –5% tolerance. Figure 10 shows the various sources of steady-state and transient currents on a power rail.

Various "fast-transient-response" point-of-load converters are available. It is generally recommended that a point-of-load converter with a fast transient response be used on fast-switching power rails. Most linear regulators use their output capacitance to set the dominant pole of their feedback loop so that they remain stable with large capacitive loads. However, most switching converters that were optimized for minimal output ripple and/or fast transient response have feedback loops that were compensated to operate within a bounded range of output capacitance and related ESR. So, when stabilizing the feedback loop of the point-of-load switching converter, the system designer must include the decoupling capacitors as part of the converter's output capacitance.

Conclusion

The secret to successful power-up of FPGAs is to use soft starting and sequencing. The amount of load current and the related power dissipation in the point-of-load converter are the determining factors for choosing a dc/dc converter. Linear regulators are appropriate for higher FPGA core voltages but require additional soft-start circuitry. Lower core voltages require switching regulators due to power dissipation limitations in linear regulators. Switching regulators typically have built-in soft start and do not require additional circuitry.

Figure 10. Sources of load transient currents



Reference

For more information related to this article, you can download an Acrobat Reader file at www-s.ti.com/sc/techlit/*litnumber* and replace *"litnumber"* with the **TI Lit. #** for the materials listed below.

Document Title	TI Lit. #
1. Jeff Falin, "Monotonic, Inrush Current Limited Start-Up for Linear Regulators," Application Report	slva156

Related Web sites

analog.ti.com
www.ti.com/xilinxfpga
www.ti.com/alterafpga
www.ti.com/sc/device/*partnumber*
Replace *partnumber* with PTH05050, TPS40003, TPS54610 or TPS64203

Failsafe in RS-485 data buses

By Kevin Gingerich (Email: k-gingerich@ti.com)
High-Performance Linear/Interface

Introduction

After bus-pin electrical overstress, the second most common cause of inquiries to the Texas Instruments (TI) Interface Applications Group is the unanticipated response of differential, and in particular RS-485*, line receivers to loss of input signal. When the response of a circuit is designed to provide a known state under this condition, it is commonly referred to as "failsafe." The intent of this article is to share the answers to many of these questions and to help the reader avoid similar problems.

While there are many different data transmission standards that employ differential signaling, the scope of this article is limited to RS-485-compatible circuits and standard usage. Many of the principles detailed here can be applied to other differential signaling schemes with appropriate adaptation of the system-level constraints.

We will first develop a bus electrical model and analyze receiver responses to and reasons for no input signal. Then we will investigate adding a signal and redefining the bus logic states through external and integrated options to provide failsafe response.

Bus model

A standard RS-485 bus is a single balanced-pair transmission line terminated at each end by a resistance equal to the characteristic impedance of the line. RS-485 line drivers, receivers, or transceivers are distributed along the transmission line to share data over the common bus (see Figure 1).

If the local ground connection of any line driver is selected as the zero potential reference point and noise sources are included, the equivalent circuit of the bus appears as shown in Figure 2. V_{OA} and V_{OB} are the output voltages of the line driver, R_{INEQ} is the equivalent input resistance of all the connected line circuits to their local ground, V_{ncm} is the common-mode noise voltage between the driver ground and receiver grounds (C), and V_{ndiff} is the differential noise voltage. We are assuming that the effect of the transmission line resistance on the node voltages is negligible.

Since we are primarily concerned with the voltage between A and B, it is convenient to make our zero potential at the B node and to substitute the difference voltage source (V_{OD}) for $V_{OA} - V_{OB}$, as shown in Figure 3. Calculation of the receiver differential input voltage, V_{ID}, is then trivial: $V_{ndiff} + V_{OD}$.

*RS-485 is specified in ANSI TIA/EIA-485-A and ISO/IEC-8482:1993.

Figure 1. Standard RS-485 bus

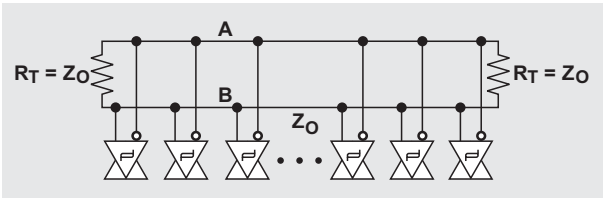


Figure 2. Electrical equivalent circuit for the standard RS-485 bus

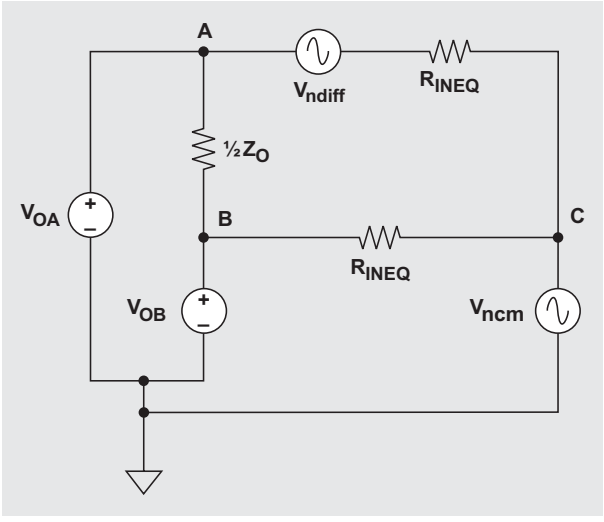
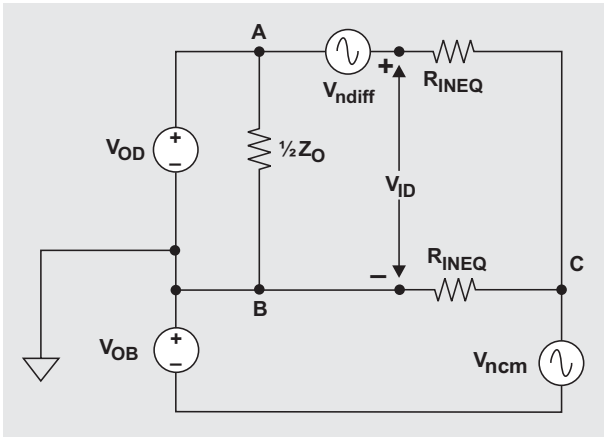


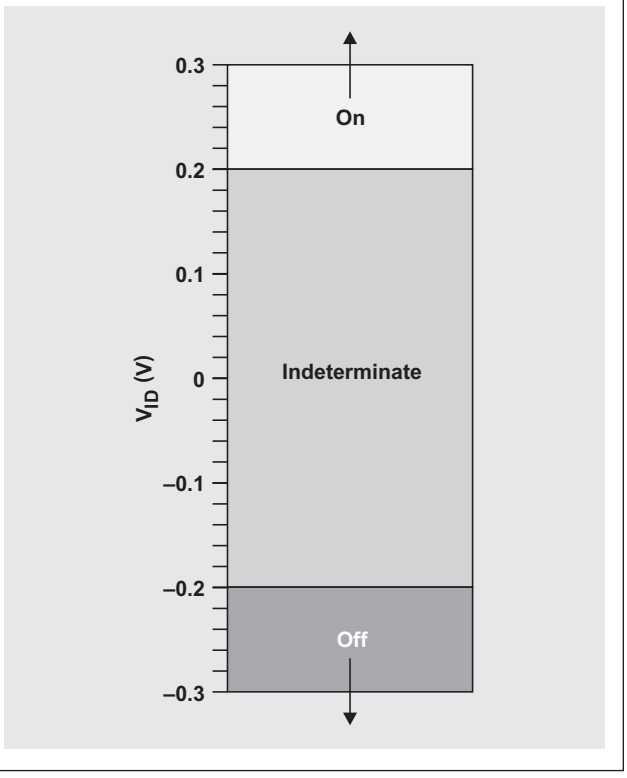
Figure 3. Redrawn RS-485 bus equivalent circuit



Differential receiver responses

The input signal to the transmission line is the driver output voltage; the differential receiver must detect what comes out. The first parameter to consider for a receiver is the differential input voltage threshold, V_{IT}, as it defines the voltage needed to place the receiver output in a high or low state. The specified maximum and minimum V_{IT} establish the limits for V_{IT}, and difference voltages above or below them assure that the receiver will indicate a valid logic state at its output. RS-485 requires a maximum V_{IT} of 200 mV and a minimum of –200 mV under all operating conditions and common-mode input voltages. Figure 4 shows a graphical representation of the differential voltage states transition region between 0.3 V and –0.3 V. The arrows indicate that the valid differential input voltage extends to 6 V or –6 V.

Figure 4. RS-485 differential voltage states



A differential input voltage between the maximum and minimum threshold lies in the transition region and results in an indeterminate output state. The output may be high, low, or on its way between states. Consider a line receiver as a differential amplifier with a gain of about 100,000. V_{IT} is the input voltage where the output would be one-half of the way between high and low; and, with no feedback for hysteresis, it takes only a few tens of microvolts above or below the threshold for the output to switch to a high or low state. There is likely this much in differential noise in the system that, if near the threshold, could cause oscillation of the output.

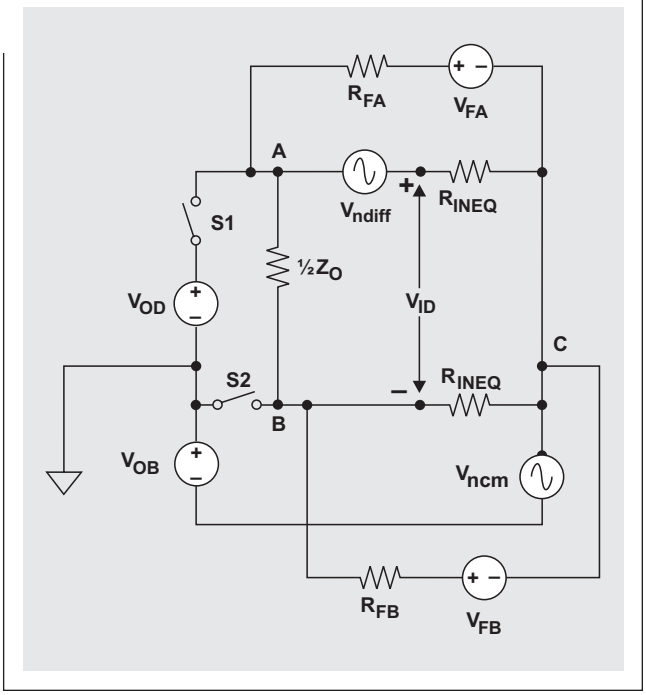
For this reason, it is common to employ some positive feedback to the differential amplifier input stages to provide some hysteresis in the input-to-output transfer function. This makes the positive-going differential threshold higher than the negative-going threshold and requires that the difference be exceeded to switch to the opposite state. This decreases the chance of oscillation from differential noise but does not provide a predictable output state, as the output will remain in the last state prior to input loss.

Adding a steady-state bias signal

Since the V_{OD} source in Figure 3 defines the bus state, disconnecting it or a zero-volt output leaves the standard bus state to be determined by the differential noise in the system or indeterminate. Operating scenarios that may disconnect V_{OD} from the circuit would include physical removal or, much more commonly, disabling the driver while bus access is being granted to another driver on the bus. A zero-volt output may occur due to a short circuit between signal wires or damage to a line circuit.

The problem of keeping the bus in a known state when no active driver is connected may be solved by adding a steady-state differential voltage to the bus. This in effect adds a driver that will not be switched, disabled, or disconnected. This failsafe driver is added to the circuit model in Figure 5 as voltage sources V_{FA} and V_{FB}, along with their respective source resistances R_{FA} and R_{FB}. The failsafe driver is referenced to node C since V_{OD} represents any driver on the circuit and the two grounds may not be at the same potential. Note that S1 and S2 are added to show the driver disconnected from the bus.

Figure 5. Failsafe driver added to equivalent bus circuit



The effect of adding the failsafe driver is determined by shorting all voltage sources other than V_{FA} and V_{FB} and summing the currents at nodes A and B as follows.

$$\frac{V_{FA}-V_A}{R_{FA}}-\frac{V_A}{R_{INEQ}}+\frac{V_B-V_A}{\frac{1}{2}Z_O}=0$$

$$\frac{V_{FB}-V_B}{R_{FB}}-\frac{V_B}{R_{INEQ}}+\frac{V_A-V_B}{\frac{1}{2}Z_O}=0$$

Dispensing with the algebra details and solving for $V_A - V_B$ gives us

$$V_A-V_B=\frac{\frac{V_{FA}}{R_{FA}}\left(\frac{1}{R_{FB}}+\frac{1}{R_{INEQ}}\right)-\frac{V_{FB}}{R_{FB}}\left(\frac{1}{R_{FA}}+\frac{1}{R_{INEQ}}\right)}{\left(-\frac{1}{R_{INEQ}}-\frac{1}{R_{FA}}-\frac{2}{Z_O}\right)\left(-\frac{1}{R_{INEQ}}-\frac{1}{R_{FB}}-\frac{2}{Z_O}\right)-\left(\frac{2}{Z_O}\right)^2}.$$

Since circuit balance requires that $R_{FA} = R_{FB} = R_F$, the *failsafe differential bias voltage* then becomes

$$V_A-V_B=\frac{\frac{V_{FA}}{R_F}\left(\frac{1}{R_F}+\frac{1}{R_{INEQ}}\right)-\frac{V_{FB}}{R_F}\left(\frac{1}{R_F}+\frac{1}{R_{INEQ}}\right)}{\left(\frac{1}{R_{INEQ}}+\frac{1}{R_F}+\frac{2}{Z_O}\right)^2-\left(\frac{2}{Z_O}\right)^2}. \quad \textbf{(1)}$$

Normally, only a 5- or 3.3-V supply is available for the failsafe voltage sources V_{FA} or V_{FB}; and, from Equation 1, we know they cannot be the same and generate a difference voltage. Therefore, we set $V_{FB} = 0$ V in the equation for *single-supplied failsafe differential voltage*,

$$V_A-V_B=\frac{\frac{V_{FA}}{R_F}\left(\frac{1}{R_F}+\frac{1}{R_{INEQ}}\right)}{\left(\frac{1}{R_{INEQ}}+\frac{1}{R_F}+\frac{2}{Z_O}\right)^2-\left(\frac{2}{Z_O}\right)^2}. \quad \textbf{(2)}$$

Note that we could just as well have made $V_{FA} = 0$ V and generated a negative failsafe bias voltage.

Before applying Equation 2, we establish a constraint for R_F. In Figure 5, the added R_F is in parallel with R_{INEQ}, which represents the equivalent resistance of the line circuits attached to the bus. This additional load has the effect of requiring more output current from a bus driver with the common-mode voltage V_{CM}. Since standard RS-485 line drivers are required to handle only 375-Ω common-mode loads, the parallel combination of R_F and R_{INEQ} is limited by

$$\frac{1}{375}>\left(\frac{1}{R_F}+\frac{1}{R_{INEQ}}\right).$$

Applying this constraint gives us the equation for the *maximally loaded-bus failsafe voltage*,

$$V_A-V_B=\frac{\frac{V_{FA}}{R_F}\left(\frac{1}{375}\right)}{\left(\frac{1}{375}+\frac{2}{Z_O}\right)^2-\left(\frac{2}{Z_O}\right)^2}. \quad \textbf{(3)}$$

The failsafe bias voltage $V_A - V_B$ is chosen and the R_F is determined with the appropriate substitutions for V_{FA} and Z_O. For example, a desired 0.25-V failsafe bias on an RS-485 bus segment with a 120-Ω characteristic impedance cable, differentially terminated with 120-Ω resistors at each end, and a 5-V ±5% supply would give us the values of Z_O= 120 Ω, V_{FA} = 4.75 V, and $V_A - V_B$ = 0.25 V. Solving for R_F results in 528 Ω. In application, a 510-Ω pull-up resistor to 5 V on line A and a 510-Ω pull-down resistor to ground on line B are connected somewhere on the bus. It is important to recognize that R_{INEQ} must be greater than 1.4 kΩ to meet the common-mode loading constraint. This limits the bus to eight 12-kΩ, sixteen 24-kΩ, or thirty-two 48-kΩ input resistance nodes.*

Adding a failsafe bias keeps a known state on the bus at all times but does not come without some system-level trade-offs. As already discussed, the number of nodes may be limited and the differential noise margin is reduced compared to the bus without the failsafe bias. Although we have developed a lumped instantaneous model for the bus, in reality it is a distributed-parameter dynamic system; and initial conditions are superimposed upon the response from desired inputs. In other words, when V_{OD} is changed, the failsafe bias voltage is added to the response. This causes an asymmetrical V_{ID} with more differential voltage in one state than the other. Both of these trade-offs call for keeping the failsafe bias as low as possible; but how much is enough?

We know that a standard RS-485 receiver needs a V_{ID} of 200 mV to define the bus state, and V_{ID} with no active driver connected is the failsafe bias $V_A - V_B$ plus the differential noise voltage V_{ndiff}. Therefore, the requirement is that $(V_A - V_B) > (200\text{ mV} + V_{ndiff})$. In well-balanced systems, it is not difficult to keep the differential noise below 50 mV, so a failsafe bias of 250 mV would be sufficient. If there is doubt about the differential noise in a particular environment, measurement is the only recourse to be sure of sufficient failsafe noise margin without unduly limiting the number of bus connections or differential noise margin.

*In RS-485 parlance, these are 1-UL, ½-UL, and ¼-UL circuits, respectively.

Redefining bus states

If the constraints imposed on the system by adding a bus failsafe bias voltage are too great or if shorted lines are included, the user may avoid the problem altogether by defining zero volts as a valid bus state. Since RS-485 defines the bus state with a more than 200-mV difference, the user is free to use lower decision thresholds to avoid unpredictable receiver responses with no input signal.

The SN65HVD08 and other recent RS-485 products from TI offer this option. Figure 6 shows the input thresholds for the SN65HVD08 compared to those of RS-485. It also shows that the SN65HVD08 will indicate the on and off states as required by the standard but is also on with no difference voltage—the failsafe condition. (The arrows indicate that the valid differential input voltages extend beyond the scale shown.)

The maximum V_{IT} for the SN65HVD08 is –10 mV and provides a worst-case 10 mV of differential noise margin with no input signal. Depending upon the user's risk tolerance or the system noise, an additional margin may be added with the steady-state failsafe bias as described earlier in conjunction with the redefined input thresholds. This reduces the required offset voltage and the negative trade-offs associated with that technique.

Since this failsafe approach is implemented in the receiver, it has a few drawbacks. Application to an entire bus requires that the user have configuration control over all the nodes connected to it. The default failsafe output is predetermined by the IC manufacturer, possibly limiting design options; and V_{IT} asymmetry adds jitter. These affect only a small percentage of applications.

Loss-of-signal detection

Another approach to failsafe was introduced with TI's SN65HVD20 family of transceivers. These devices include a differential window comparator in parallel with the RS-485 receiver to detect loss of signal. Once detected, the output of the receiver is forced to a high level after a time delay. This results in the state thresholds shown in Figure 7.

Separation of the window comparator from the main line receiver allows symmetry of V_{IT} about zero on the main receiver, avoiding addition of jitter for high-speed applications. This separation, as implemented in the SN65HVD2x devices, offers 40 mV of differential noise margin with no signal. This is the only approach that creates a unique signal indicating signal loss that could be brought out of the line circuit for customized user processing.

The propagation-delay time of the window comparator can be a system issue since it is set by a timer in the SN65HVD2x and the user must account for the output state during the time between signal loss and activation of the failsafe output.

Figure 6. SN65HVD08 and RS-485 input voltage thresholds

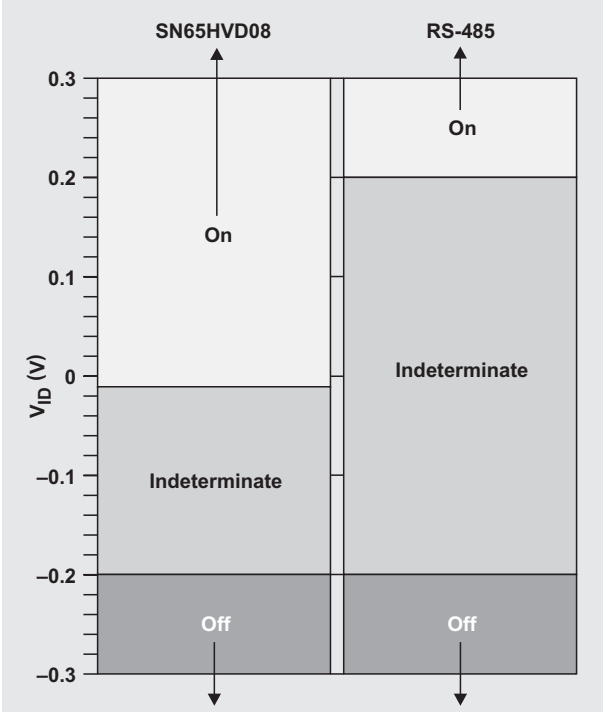
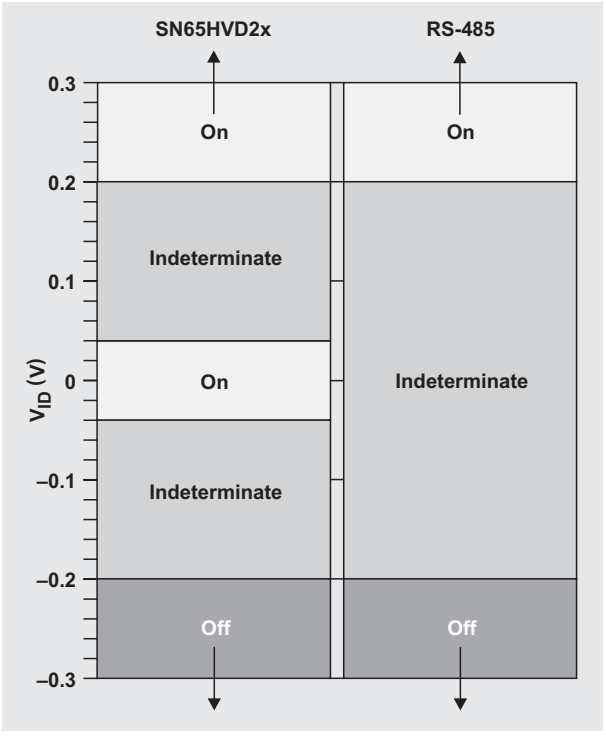


Figure 7. SN65HVD2x and RS-485 input voltage thresholds



Summary

Unexpected differential receiver behavior due to the loss of input signal is a common source of application problems. Obtaining a predictable output under this condition is called failsafe and three approaches were presented; adding a failsafe bias voltage to the bus, redefining the bus states, and loss-of-signal detection. The advantages and disadvantages of these are summarized in Table 1.

Related Web sites

analog.ti.com
www.ti.com/sc/device/SN65HVD08
www.ti.com/sc/device/SN65HVD20

Table 1. Summary of failsafe techniques

FAILSAFE TECHNIQUE	ADVANTAGES	DISADVANTAGES
Adding a bus bias voltage	• Uses standard receivers • Adaptable to application	• Reduces maximum number of nodes • Reduces differential noise margin • Adds jitter • Requires two resistors • Does not handle shorted lines
Redefining zero volts	• Available integrated with the receiver • Handles shorted-line condition	• Requires nonstandard receiver • Fixed failsafe output state • Adds jitter
Loss-of-signal detector	• Available integrated with the receiver • Uses standard receivers (with stand-alone window detector) • Handles shorted-line condition • Does not add jitter • Can create information for use by application	• Delay time from signal loss to forced output

Active filters using current-feedback amplifiers

By Randy Stephens (Email: r-stephens@ti.com)
Systems Specialist, Member Group Technical Staff

Introduction

The use of filters in circuit design is very common. They can be found in circuits such as anti-aliasing for analog-to-digital converters (ADCs), image rejection for digital-to-analog converters (DACs), intermediate frequency (IF) stages in communications systems, and even simple bandwidth limiting. However, when the frequency of interest is higher than a megahertz or two, using a current-feedback (CFB) amplifier to perform the task of filtering may be a good choice.

A CFB amplifier has some attributes that make it especially suited as a very high-frequency filter. These include the essentially limitless gain-bandwidth product, an inherently low voltage noise at the expense of more current noise, and the capability for exceptionally high slew rates.

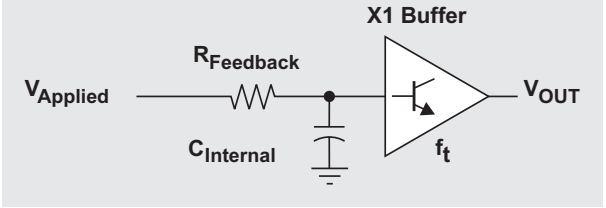
CFB to VFB compensation

One key difference between a voltage-feedback (VFB) amplifier and a CFB amplifier is that the VFB amplifier can use a very wide resistance in the feedback path and maintain a common frequency response characteristic. A CFB amplifier relies on the feedback-path impedance to compensate the amplifier and thus does not have nearly the flexibility in resistance that a VFB amplifier inherently possesses. With a VFB amplifier, placing a capacitor in the feedback path will simply cause a pole to form—hence creating a first-order filter. Doing this to a CFB amplifier, however, will cause the amplifier to oscillate. Why is this? Exploration of this subject can be found in several application reports (see References 1–3) and will not be repeated in this article.

The most simplistic explanation of how a CFB amplifier is compensated is shown in Figure 1. There is much more to the CFB amplifier than a simple RC filter and a buffer, but this is an easy way to see how the amplifier behaves. The capacitor (C) is internal and fixed, while the resistor (R) is external and can be visualized as the feedback resistance. Just like a true RC filter, as R increases, there is more compensation and the bandwidth decreases; but if R is decreased too much, the bandwidth will increase to such a point that the buffer's own transistor frequency effects (f_t) will come into play and cause instability. It is easy to see that adding a capacitor in parallel with R cancels out the C compensation, leading to oscillations.

Reference 4 discusses how to make a CFB amplifier work even though there may be a capacitor in the feedback loop.

Figure 1. Simplified compensation of a CFB amplifier



This same technique allows for the construction of any type of active filter required. To see the advantages of the CFB amplifier more clearly, a set of second-order, 0.5-dB-ripple Chebyshev filters was constructed with a corner frequency of 40 MHz at gains of ±2 and ±5.

The Sallen-Key filter with gain = +2

The Sallen-Key filter is inherently suitable for CFB amplifiers due to the feedback loop essentially being isolated from the filter loop. This circuit allows for any resistance to be used in the feedback loop without directly affecting the frequency characteristics. A detailed analysis of the Sallen-Key filter can be examined in Reference 5. Realizing a design that has proper component values can be quite time-consuming. To help simplify the process and to minimize the chance of miscalculations, Texas Instruments (TI) has developed a filter design program called FilterPro™. FilterPro is available for download at no cost from TI's Web site (go to www.ti.com and enter *FilterPro* in the Search Keyword field). The program allows for an easy realization of the desired filter with industry-standard component values and shows the expected frequency response of the filter.

One drawback of this tool is that it uses ideal amplifiers. In the real world, amplifiers have their own bandwidth limitations that can alter the frequency response of the filter. As a general rule of thumb, it is desirable to have the bandwidth of the amplifier at least 10 times higher than the filter's corner frequency. Some of the test results presented later on in this article will show how the amplifier bandwidth can alter the filter's response.

The 40-MHz, second-order, 0.5-dB Chebyshev filter with a forward gain of +2 V/V (+6 dB) was constructed with

the THS4271 (see Figure 2). The THS4271 is a unity-gain stable VFB amplifier with a 390-MHz bandwidth at a gain of +2. The same filter was also realized with the THS3201—a CFB amplifier with a 725-MHz bandwidth at a gain of +2 (also shown in Figure 2).

All tests were run with ±5-V power supplies and were constructed on printed circuit boards (PCBs) with proper high-speed layout techniques and bypassing. One issue with all of these filters is that the input impedance is not fixed and can change drastically from dc up to well beyond the filter's response. It is easy to see that, at dc, the input impedance is determined by the amplifier's input impedance in parallel with the 100-Ω termination resistor. For both the VFB and CFB amplifiers, the noninverting input impedance is typically much greater than several megohms and can effectively be ignored at low frequencies.

At extremely high frequencies, the capacitances of the amplifier input (a few pF), the PCB (a few pF), and the package (about a pF) reduce the amplifier's effective input impedance; but the external 27-pF and 47-pF capacitors should be the dominant capacitors in the system. Since the impedance can never be properly matched, it was determined to set the termination for all of these filters to give approximately 50 Ω of impedance at the corner frequency of 40 MHz.

A very important consideration when capacitor component values are chosen is to keep the capacitance above a minimum of 10 pF. This is to reduce the effects of all the stray capacitances in the system. It will be shown that it is not always possible to meet this goal. In these cases it is best to try to measure the amount of stray capacitance on the PCB and to readjust the capacitor value accordingly to reach the proper design value. Another generally accepted method is simply to use the next lower standard capacitor value, such as 3.3 pF instead of 3.9 pF.

The results of testing the VFB and CFB amplifiers in Figure 2 are shown in Figure 3.

Both of these responses show that there is about a 0.75-dB ripple rather than the desired 0.5-dB ripple. However, this is a fairly reasonable response considering component tolerances, parasitics, and the sheer fact that this is at 40 MHz with a gain of 2.

Since both amplifiers behave similarly, using either one seems reasonable; but the key difference between the two is not actually seen here. The CFB amplifier's capability of very high slew rates compared to the VFB amplifier will translate to a better high-amplitude response and a better distortion characteristic at the higher amplitudes. Using the simple formula

$$f = \frac{\text{Slew rate}}{2\pi V_{PEAK}}$$

shows that with a 5-V_{PP} signal at 40 MHz, the slew rate must be at least 628 V/µs. Both the THS4271 (1000 V/µs) and the THS3201 (5200 V/µs) certainly have the capability to reproduce this type of signal; but with the CFB amplifier's 5× slew-rate advantage, its odd-order distortion should be better.

Figure 2. Sallen-Key 40-MHz filter with gain = +2 V/V

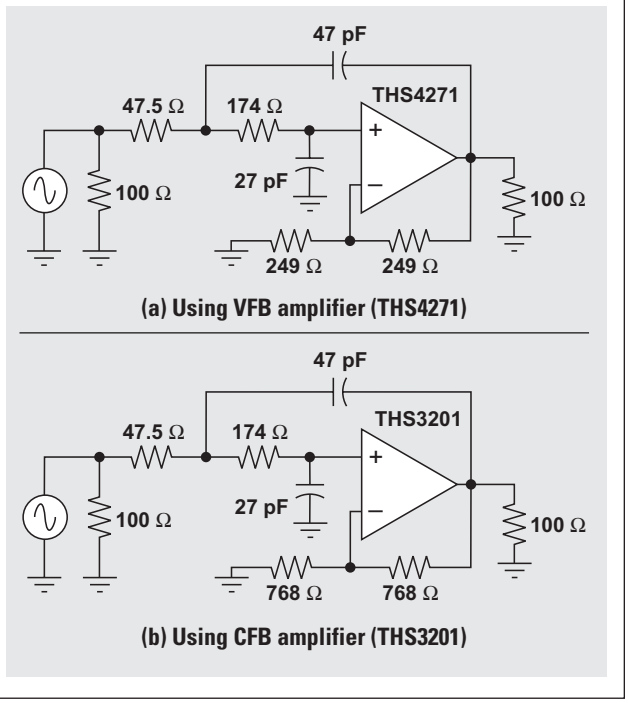
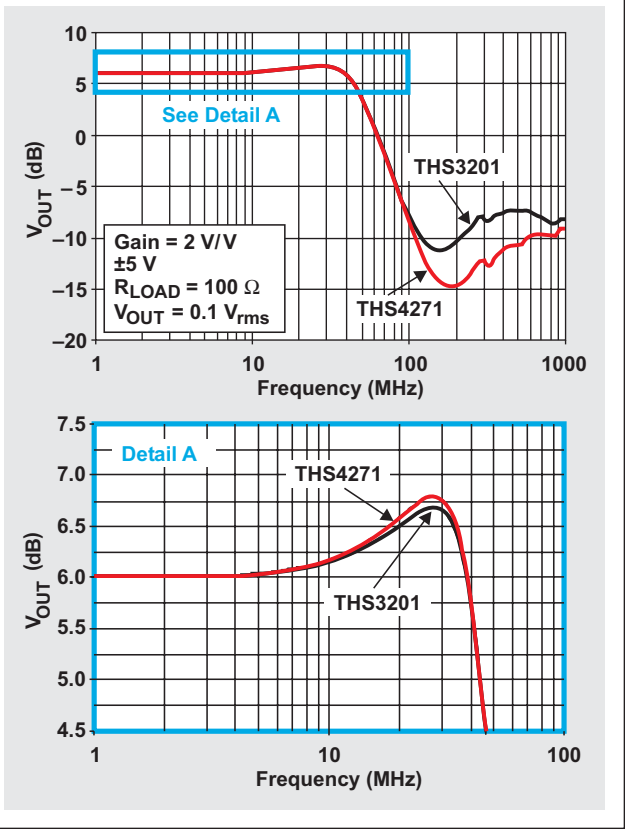


Figure 3. Sallen-Key filter responses with gain = +2 V/V



Another interesting result is that there is at best an 11-dB rejection (input referred) with the THS3201 and almost a 15-dB rejection with the THS4271. These responses are actually within reason for a Sallen-Key filter. The Sallen-Key filter topology requires the amplifier to have a very low output impedance within the rejection band. The 47-pF capacitor connecting between the input and the amplifier output is also an excellent high-frequency path for the input signal. If the amplifier does not have the bandwidth and low output impedance to perform well at these very high frequencies, it cannot reject the high-frequency content and the signal passes through the system.

The Sallen-Key filter with gain = +5

The next step was to see how the responses change with a gain change from +2 V/V (6 dB) to +5 V/V (14 dB). A VFB amplifier's bandwidth will decrease as the gain increases and should start showing effects on the frequency response of the filter. The THS4271 at a gain of +5 V/V has a bandwidth of about 85 MHz, only about 2× the desired corner frequency of the filter. On the other hand, a CFB amplifier can have a much higher bandwidth with a simple reduction of the feedback resistance; think of it as decompensating the amplifier. The THS3201 at a gain of +5 V/V will have about a 540-MHz bandwidth. FilterPro realized the filters shown in Figure 4.

Notice that the component values for the negative feedback path can be chosen arbitrarily from the filter components. Only the ratio must be maintained to achieve the proper filter feedback. This allows the CFB and VFB amplifiers to utilize their own "optimum" resistor values in the system. The responses, shown in Figure 5, have considerably more peaking than the desired 0.5-dB ripple. Again, some of this peaking is caused by the component tolerances; but a big influence is the amplifier's reduced bandwidth capability. The VFB amplifier (THS4271) shows almost 2 dB of peaking along with a –0.5-dB point at 34 MHz, deviating considerably from the target specification.

The CFB amplifier (THS3201) has about 1.25 dB of peaking, but its –0.5-dB point is at about 44 MHz. The fact that the bandwidth is about 13× the filter's corner frequency strongly suggests that component tolerances are a highly influential factor in the filter response.

Once again, the amount of rejection remains about the same between the design with a gain of +2 V/V and that with a gain of +5 V/V. This means that the amplifier's frequency characteristics are a major factor in the design, as originally expected.

Figure 4. Sallen-Key 40-MHz filter with gain = +5 V/V

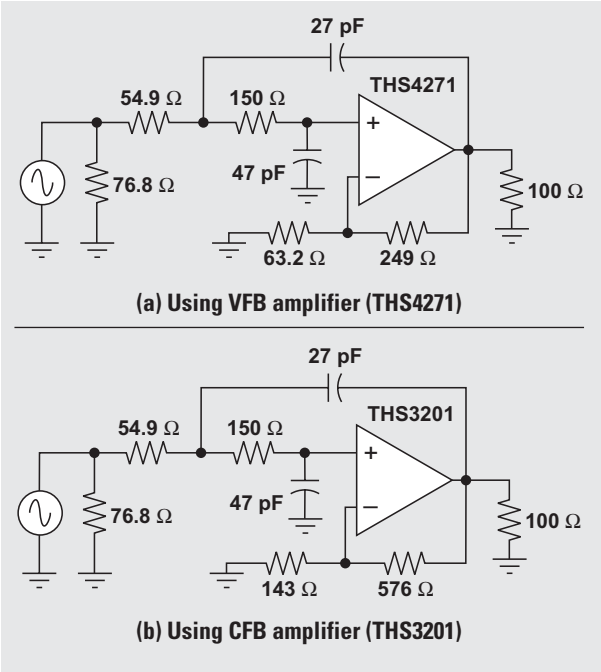
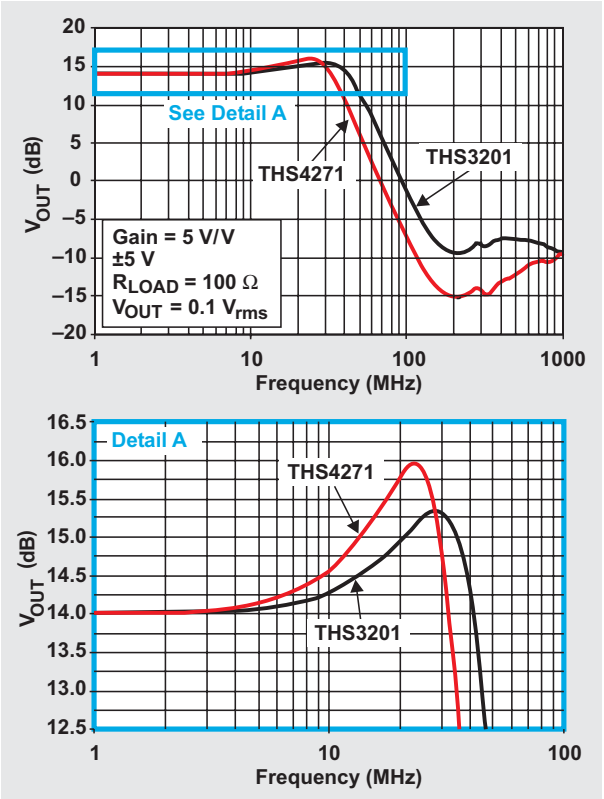


Figure 5. Sallen-Key filter responses with gain = +5 V/V



Another key point that should be made is that the noise of a CFB amplifier at gains typically higher than 3 V/V can easily be lower than the noise of a VFB amplifier. This is because the CFB amplifier inherently has low voltage noise and because, as the optimum feedback resistance decreases at higher gains, the noise contribution from the CFB's inverting current noise also decreases. On the other hand, a VFB's dominant noise component stems from the voltage noise, which becomes directly multiplied by the gain of the amplifier.

One last issue that must be addressed is that there are decompensated VFB amplifiers available that may be better suited for higher-gain systems—such as the OPA843, OPA846, and OPA847, with a minimum gain of +3 V/V, +7 V/V, and +12 V/V, respectively. These amplifiers typically have a reduced voltage noise as the minimum gain is increased, along with a higher slew rate and a higher gain-bandwidth product. They are certainly viable alternatives that may find success in many higher-gain filter designs; but, for an all-in-one amplifier solution, a CFB amplifier is very versatile.

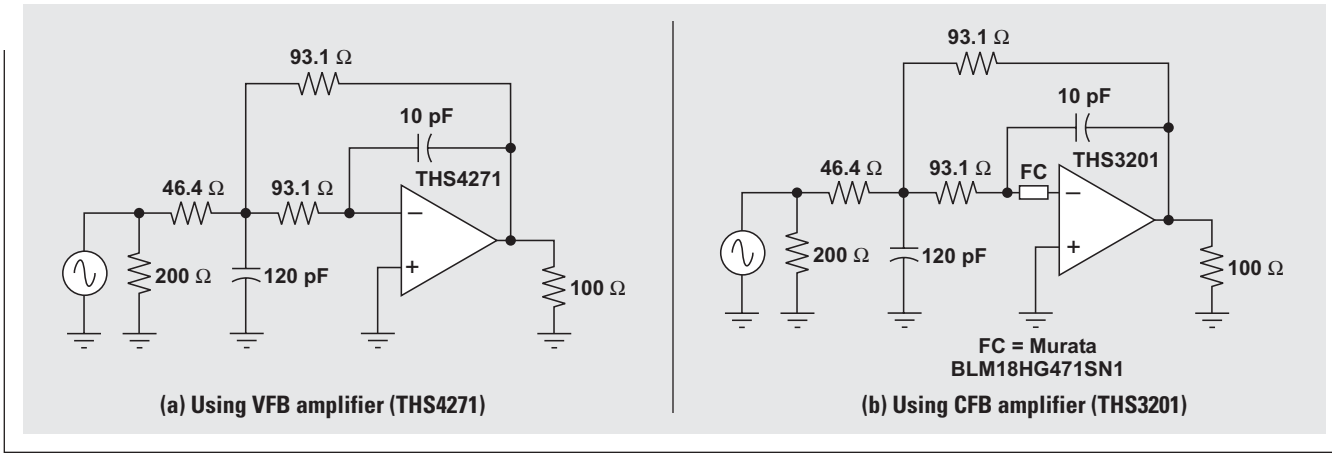
The multiple-feedback filter with gain = –2 V/V

The next most commonly used filter is the multiple-feedback (MFB) filter, also known as the Rauch filter. One advantage of the MFB filter is its reduced sensitivity to component variation. This is important when real-world capacitors can easily have ±15% temperature variances, +5/–10% variances over frequency, and ±10% variances over operating voltage. The same 0.5-dB-ripple Chebyshev design was done with FilterPro with a gain of –2 V/V (+6 dB with a 180° phase shift) and is shown in Figure 6a for the THS4271.

There are some drawbacks to the MFB design. First is the obvious fact that the capacitor value in the feedback path is very small—10 pF. Even using very small resistor values around the amplifier did not help increase the capacitor value very much. Another possible issue is that, in the inverting configuration, the amplifier's noise gain is +1 higher than before. What this means is that the bandwidth of the amplifier is approximately equal to the gain-bandwidth product divided by the noise gain. Noise gain is always referred to the noninverting terminal and thus has a gain of $1 + R_f/R_g$. For this design with a gain of –2 V/V, the noise gain is +3 V/V, reducing the effective bandwidth even more than the non-inverting Sallen-Key design. Note that when operating well above the filter's corner frequency, the feedback capacitor essentially becomes a short, resulting in an amplifier noise gain of +1. For this reason, using a decompensated VFB amplifier is not suggested without special techniques.

Can this topology be used with a CFB amplifier? There is a capacitor directly in the feedback loop, and traditional thought suggests that there is no way to use a CFB amplifier. The method outlined in Reference 4, however, makes it possible to use an MFB circuit with a CFB amplifier. For this test a Murata BLM18HG471SN1 ferrite chip was chosen, as its impedance is about 650 Ω at 700 MHz and about 600 Ω at 1 GHz. The key reason for using a ferrite chip is its low impedance—and hence low noise—at low frequencies while still maintaining enough impedance at high frequencies to keep the amplifier stable. Keep in mind that this chip has about 200 Ω of impedance at 20 MHz and about 100 Ω at 6 MHz, so the contribution of inverting current noise must be considered in the total noise of the system. The final design of the MFB circuit with the THS3201 CFB amplifier is shown in Figure 6b.

Figure 6. MFB 40-MHz filter with gain = –2 V/V



The frequency responses of these systems, shown in Figure 7, reveal an excellent reason to use the MFB filter—the out-of-band attenuation is far superior to the Sallen-Key response, by about 20 dB. This superiority is due to the inherent RC filter (46.4 Ω and 120 pF) to ground at the input of the MFB filter. The filter shunts the input signal to ground even if the amplifier is running out of bandwidth.

Figure 7 also shows that the VFB amplifier has a much better filter response with a 0.5-dB ripple and a corner frequency of 36 MHz. The CFB amplifier has nearly a 2-dB ripple and an extended corner frequency of 44 MHz. Since exactly the same components were swapped from one test PCB to the other, component variations can be ruled out as the main contributor to this excess peaking. Additionally, the bandwidth of the THS3201 is over 600 MHz with a gain of –2 V/V, minimizing the effects of amplifier bandwidth interactions; and the THS4271 has only about a 220-MHz bandwidth at a gain of –2 V/V. This leaves the ferrite chip's interaction with the amplifier as the only probable cause of this peaking.

The impedance of the ferrite chip increases with increasing frequency. Since the CFB amplifier's compensation is dictated by the feedback impedance, this will have a direct impact on the amplifier's characteristics. The true test of this impact was to replace the ferrite chip with pure 750-Ω and 249-Ω resistors, whose responses are shown in Figure 8. It is interesting to note that as the frequency increases, the response of the ferrite chip approaches that of the 750-Ω resistor. This makes sense, since the ferrite chip's impedance is about 700 Ω at its peak.

It is also noteworthy that the 249-Ω resistor allows the amplifier to remain stable. We would expect the THS3201 to need at least 600 Ω of impedance to be stable with a gain of –1 V/V. The reason it remains stable with 249 Ω of impedance is that the 10-pF capacitor has some impedance at 750 MHz. Remember that in the real world, a capacitor has an associated inductance that causes its impedance to increase at high frequencies. Add this impedance to the amplifier's output impedance, and the real resistor value in an inverting configuration does make the amplifier stable. The response at about 400 MHz shows some aberrations, which implies that stability is starting to be a bit of a concern. With –30 dB of attenuation at this point, however, the system will remain stable; as one of the conditions for instability is that the amplitude must be greater than 0 dB.

Figure 7. MFB filter responses with gain = –2 V/V

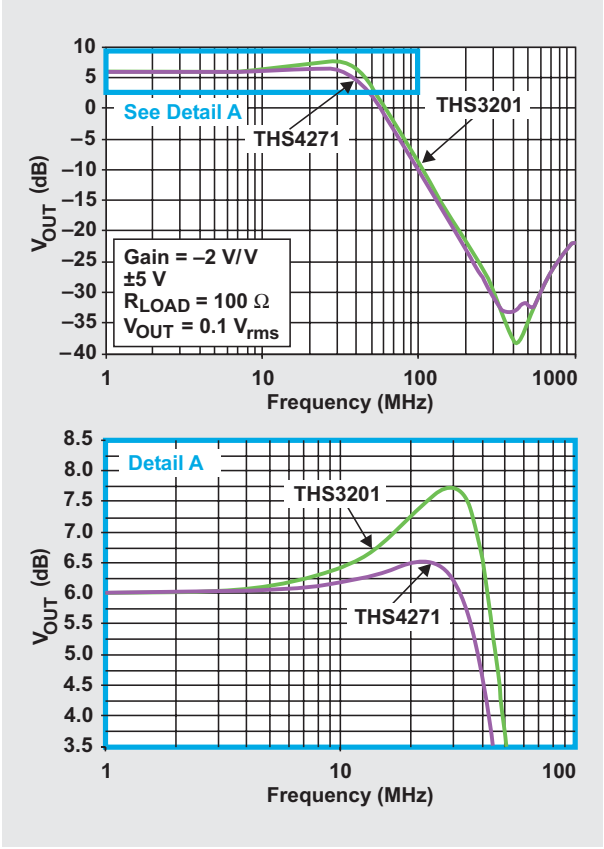
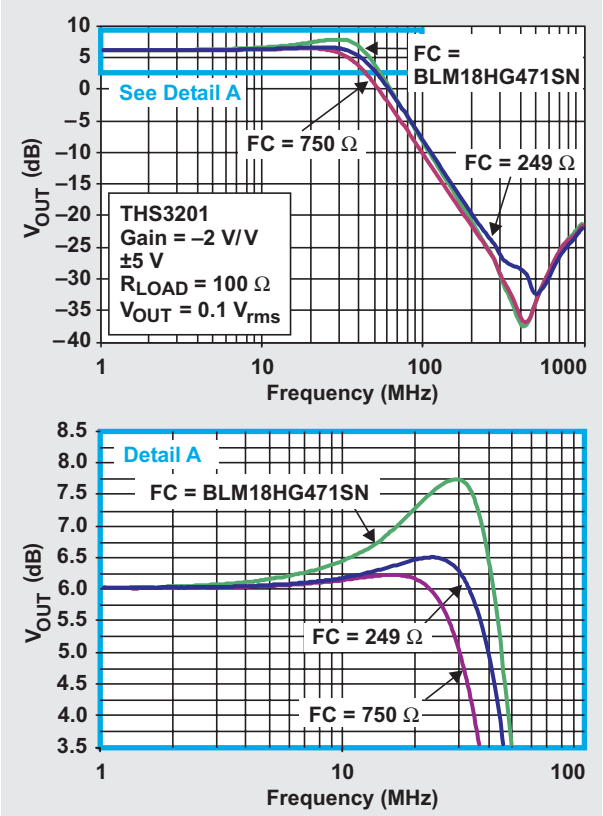
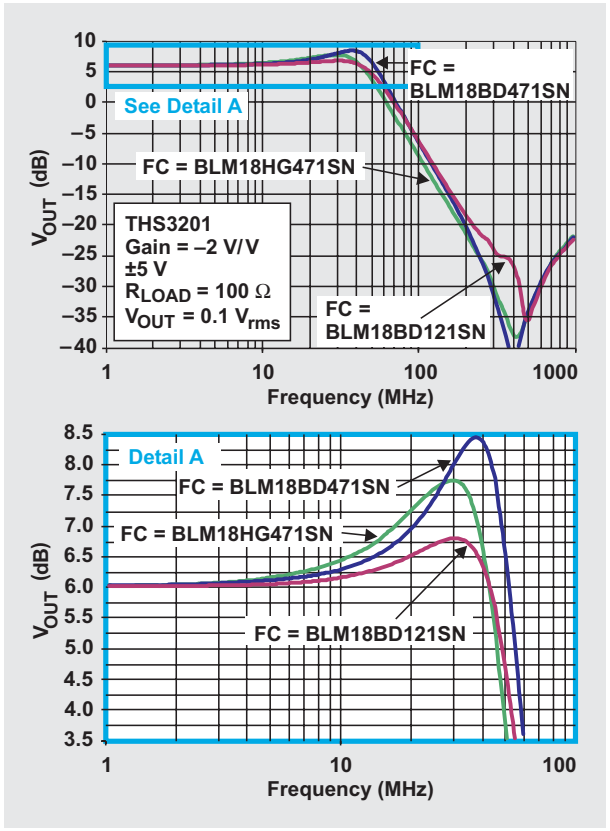


Figure 8. CFB amplifier MFB responses with resistors and gain = –2 V/V



It is obvious that the ferrite chip's impedance characteristics directly impact the system. The use of other types of ferrite chips was explored, and their responses are shown in Figure 9. From this plot it appears that using the BLM18BD121SN chip with only 120 Ω of impedance at 100 MHz performs the best. This chip has a maximum impedance of 300 Ω at 800 MHz, which resembles the 249-Ω resistor's high-frequency characteristics at 400 MHz; but, again, having substantial attenuation will help keep the system stable.

Figure 9. CFB amplifier MFB responses with other ferrite chips and gain = –2 V/V



The multiple-feedback filter with gain = –5 V/V

The last circuit explored was the same MFB topology except with a gain of –5 V/V. As we know, this implies a noise gain of +6 V/V; but the impact on amplifier bandwidth is negligible between these two gains and should not be a concern. Figure 10 shows the circuits used to test this configuration.

The obvious concern about this design is the 3.3 pF in the feedback path. Stray capacitance can easily influence the circuit relative to the capacitance value. It is possible to lower the resistor values around this, but a value of 10 pF would cause the input resistance to change from 53.6 to 15.8 Ω, placing too large a load at the rejection band of the filter. Needless to say, the filter was used as shown in

Figure 10. MFB 40-MHz filter with gain = –5 V/V

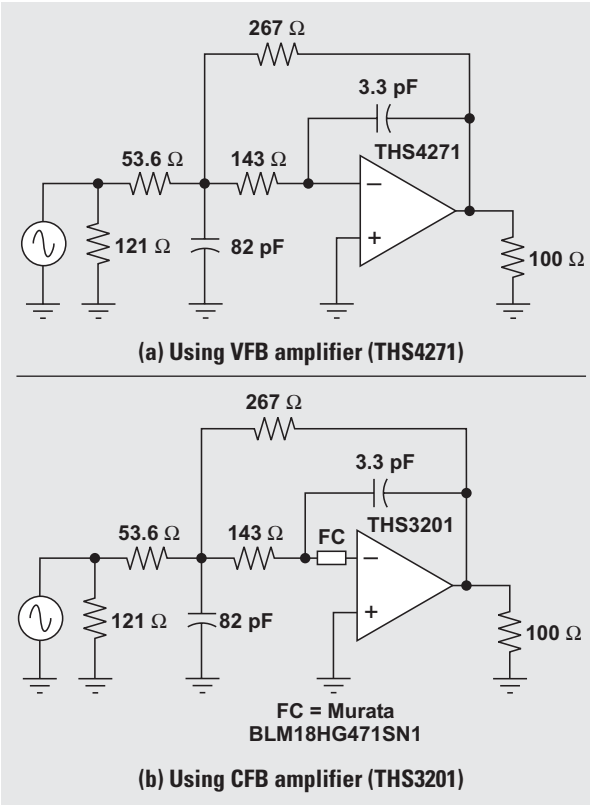


Figure 10. Figure 11 shows the responses, which are pretty much what is expected. The VFB amplifier does not have enough bandwidth to create a 40-MHz filter properly and rolls off prematurely, resulting in a corner frequency of 23 MHz.

The CFB amplifier exhibited peaking of only about 1.25 dB and a corner frequency of 34 MHz. As stated previously, the interaction of the ferrite chip and the amplifier comes into play. Just as the response of the CFB amplifier MFB design with a gain of –2-V/V showed an interaction with the ferrite chip's impedance, so did the design with a gain of –5 V/V, which exhibited similar results as shown in Figures 12 and 13. The best response was achieved when the impedance of the chip was very low at the pass-band corner frequency and increased to at least 250 Ω at 400 to 500 MHz.

Figure 11. MFB filter responses with gain = –5 V/V

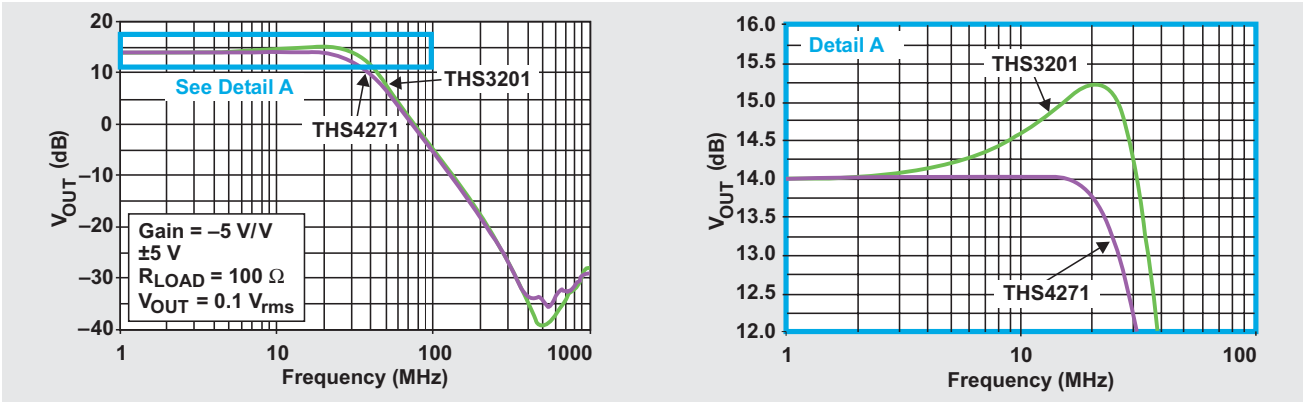


Figure 12. CFB amplifier MFB responses with resistors and gain = –5 V/V

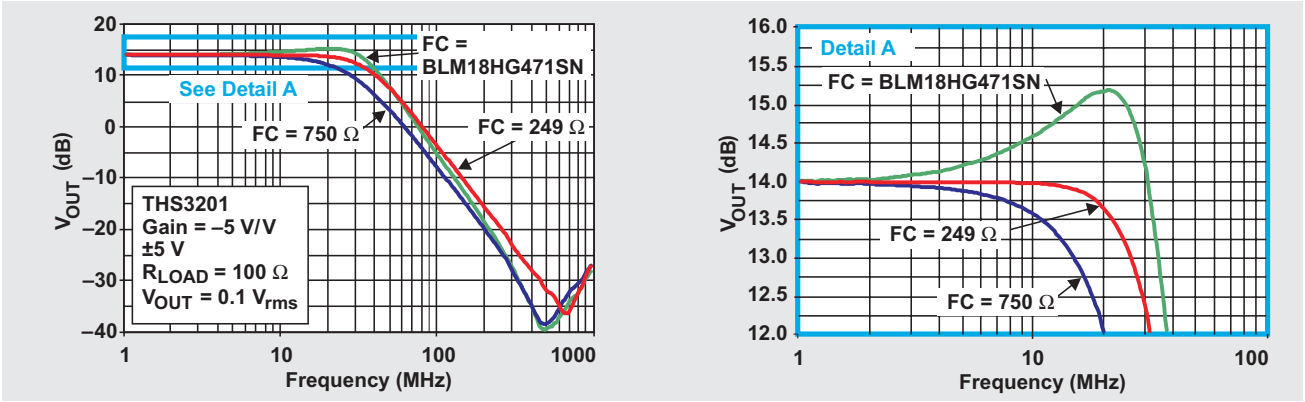
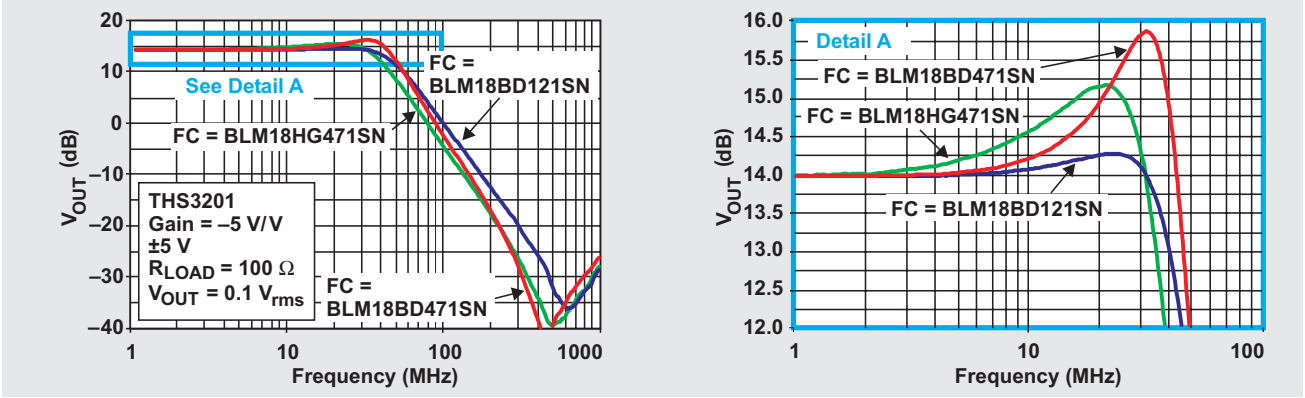


Figure 13. CFB amplifier MFB responses with other ferrite chips and gain = –5 V/V



Conclusion

As any circuit dictates, there will always be trade-offs in any design, and filtering is no exception. The MFB filter certainly shows excellent performance in the rejection band, over 20 dB better than the Sallen-Key filter; and its sensitivity to tolerance is much better. However, the use of very small capacitors in the system may limit its overall usefulness as a filter.

The use of a CFB amplifier as a filter certainly has been proven to be functional. The enhanced bandwidth and slew-rate capabilities show even better potential than a VFB amplifier; but the trade-off in the MFB design may be a hindrance to its acceptance as a good high-frequency filter. Nevertheless, independent testing at frequencies lower than 10 MHz has shown excellent results. This makes sense, as the impedance of the ferrite chip typically does not become too high until this point and can be negligible in the system. Coupled with the reduced output noise as the gain increases, the CFB amplifier can be a good choice for the proper application. It is not perfect, and the use of VFB amplifiers may make more sense in the circuit.

Of course, the unity-gain stable VFB amplifier can work with any topology without issue; but the gain-bandwidth-product limitation exists, and the slew-rate capabilities are not nearly as good as those of a CFB amplifier. The VFB amplifier also may not be the perfect building block if multiple gains are required due to the bandwidth reductions at higher gains. Instead, the use of decompensated amplifiers at higher gains for the Sallen-Key filter would be suggested, as this allows higher performance with low noise; but this approach can be used with only limited gain ranges and cannot be used with the MFB filter without the need for its own special compensation techniques.

References

For more information related to this article, you can download an Acrobat Reader file at www-s.ti.com/sc/techlit/*litnumber* and replace *"litnumber"* with the **TI Lit. #** for the materials listed below.

Document Title	**TI Lit. #**
1. James Karki, "Voltage Feedback Vs. Current Feedback Op Amps," Application Report	slva051
2. Anthony D. Wang, "The Current-Feedback Op Amp: A High-Speed Building Block," Application Bulletin	sboa076
3. John Austin, "Current Feedback Amplifiers: Review, Stability Analysis, and Applications," Application Bulletin	sboa081
4. Randy Stephens, "Expanding the Usability of Current-Feedback Amplifiers," *Analog Applications Journal* (3Q 2003), pp. 23–28	slyt047
5. James Karki, "Effect of Parasitic Capacitance in Op Amp Circuits," Application Report	sloa013

Related Web sites

analog.ti.com
www.ti.com/sc/device/*partnumber*
Replace *partnumber* with THS3201, THS4271, OPA843, OPA846 or OPA847

Index of Articles

TI Worldwide Technical Support

Internet

TI Semiconductor Product Information Center Home Page
support.ti.com

TI Semiconductor KnowledgeBase Home Page
support.ti.com/sc/knowledgebase

Product Information Centers

Americas

Phone	+1(972) 644-5580
Fax	+1(972) 927-6377
Internet/Email	support.ti.com/sc/pic/americas.htm

Europe, Middle East, and Africa

Phone	
Belgium (English)	+32 (0) 27 45 55 32
Finland (English)	+358 (0) 9 25173948
France	+33 (0) 1 30 70 11 64
Germany	+49 (0) 8161 80 33 11
Israel (English)	1800 949 0107
Italy	800 79 11 37
Netherlands (English)	+31 (0) 546 87 95 45
Spain	+34 902 35 40 28
Sweden (English)	+46 (0) 8587 555 22
United Kingdom	+44 (0) 1604 66 33 99
Fax	+(49) (0) 8161 80 2045
Internet	support.ti.com/sc/pic/euro.htm

Japan

Fax	International	+81-3-3344-5317
	Domestic	0120-81-0036
Internet/Email	International	support.ti.com/sc/pic/japan.htm
	Domestic	www.tij.co.jp/pic

Asia

Phone	
International	+886-2-23786800
Domestic	Toll-Free Number
Australia	1-800-999-084
China	800-820-8682
Hong Kong	800-96-5941
Indonesia	001-803-8861-1006
Korea	080-551-2804
Malaysia	1-800-80-3973
New Zealand	0800-446-934
Philippines	1-800-765-7404
Singapore	800-886-1028
Taiwan	0800-006800
Thailand	001-800-886-0010
Fax	886-2-2378-6808
Email	tiasia@ti.com ti-china@ti.com
Internet	support.ti.com/sc/pic/asia.htm

A111103

SLYT060