Engineering journal volume Sixty-Eight

| LETTER FROM THE CEO | | 2 |
|---------------------|--|----|
| | | |
| IN-DEPTH ARTICLES | | |
| | Propagation delay measurements using TDR (time-domain reflectometry) | 3 |
| | Use thermal analysis to predict an IC's transient behavior and avoid overheating | 9 |
| | Selecting high-linearity mixers for wireless base stations | 18 |
| | How to use high-voltage and high-current-drive op amps in ±20mA or 4–20mA | |
| | current-loop systems | 21 |
| | | |



A VI converter transforms the DAC output to load current. The circuit uses two MAX9943 op amps. (See article on page 21.)



Letter from the CEO

Getting Your Systems to Market Faster—It's Our Job

Each new IC generation that we develop is usually more highly integrated than the previous. So a new IC typically offers more features and functionality, and thus should simplify the design of the final system. But we all know that those new features and enhanced functionality add complexity. Thus evaluating a new chip for your application becomes harder as device complexity increases.

In past years a typical evaluation scenario involved inserting the IC into a breadboard and connecting a few wires to make the chip "wiggle." However, today that scenario is very different. Many mixed-signal chips now contain registers or other digital interfaces that require some form of serial interface so that various control parameters can be downloaded to the chip from a host computer. Additionally, some functions on the chip can be so specialized that it is difficult to quickly breadboard a test circuit to exercise the chip. Furthermore, at high operating frequencies considerable attention must be paid to the board layout so that performance is not compromised. This extra time and effort can mount up and quickly impact a tight product-development schedule.

We know that your time is critical when evaluating a new product. To make the best use of your time, we offer a broad range of evaluation kits. These EV kits are inexpensive tools that let you evaluate the functionality of the chip, apply inputs and control functions, and view the results—all in a matter of a few hours, not days or weeks.

Of course, not all chips are simple to set up, and for those devices we offer a system-level evaluation solution. For example, we build full systems on a board for our MAXQ[®] microcontroller family and some of those EV kits even include evaluation installations of the software development tools. These tools include a free assembly project-development environment (MAX-IDE), the Rowley Crossworks IDE, and the IAR Embedded Workbench[®] assembly tool, plus emulators to help you jumpstart your program development.

And we don't stop with system-level evaluation solutions. We go a step further and offer reference designs that are complete system solutions to showcase how chips really perform. Reference designs cover a broad swath of applications, including our new MG2580 H.264 codec in an HD-resolution IP camera; proprietary 1-Wire[®] solutions for sensing, data collection, and identification; automotive lighting and telematics systems; battery management, like a design that derives 5V and 3.3V supplies from a USB port; microcontroller designs like using the MAXQ3222 as a digital thermostat or using the MAXQ314 for single-phase electric power monitoring; digital TV terrestrial tuners; and many other solutions.

We augment these EV kits, software tools, and reference designs with one of the industry's largest collections of application notes and strong in-house and field applications support teams. All our focus is on helping you get your design to market.

We are always open to new ideas, so let us know what else we can do to help you. Drop me a note at tunc@ maxim-ic.com.

Respectfully,

Doluca

Tunç Doluca President and Chief Executive Officer

1-Wire and MAXQ are registered trademarks of Maxim Integrated Products, Inc IAR Embedded Workbench is a registered trademark of IAR Systems AB.

Innovation Delivered is a trademark and Maxim is a registered trademark of Maxim Integrated Products, Inc. @ 2010 Maxim Integrated Products, Inc. All rights reserved.

Propagation delay measurements using TDR (time-domain reflectometry)

Bernard Hyland, Applications Engineer and Senior Member of Technical Staff

Maxim Integrated Products, Inc.

Abstract: As clock speeds increase, it is more difficult for traditional methods of measuring delays with active probes connected to high-speed oscilloscopes to obtain accurate results. These probes become a part of the high-speed signal path and distort the signal being measured, thus introducing errors. The probes must also be placed directly on device pins to remove delay errors caused by PCB (printed circuit board) run lengths, and that placement is a difficult and complex procedure. This article will demonstrate how to use TDR (time-domain reflectometery) measurements to minimize probing errors and improve the accuracy of propagation delay measurements.

Analytical approach

This article proceeds from three basic premises.

- 1. TDR (time-domain reflectometry) minimizes probe errors. TDR is normally used to measure the length vs. impedance changes along a signal path. TDR is also a valuable tool for measuring propagation delays.
- 2. Avoid direct probing. Due to loading, active probes can complicate the measurement and introduce errors.
- 3. Use a real example to demonstrate the method. The example in this article will be the MAX9979, a chip that contains high-speed pin-electronics circuitry for ATE systems. Included on the chip are dual high-speed drivers, active loads, and window comparators that operate in excess of 1Gbps.

The approach presented here applies to any high-speed device.

TDR basics

TDR is a method in which a high-speed signal edge is propagated down a signal path and the reflection is observed. The reflection will show both the impedances along that signal path, and the change in signal delay for each of those impedance changes. A simple tutorial for TDR is shown in **Figure 1**.



Figure 1. TDR fundamentals. TDR measurements are based on the reflection coefficient, ρ , where $\rho = (V_{REFLECTED}/V_{INCIDENT})$. Finally, $Z_{\rho} = \rho \times (1 + \rho)/(1 - \rho)$.

There are two important concepts to note from Figure 1:

- 1. T_{DLY} is the delay of the PCB (printed circuit board) run that we will be measuring.
- 2. Z_0 is the impedance of the PCB run under measurement.

Instrumentation and EV board

To measure delays in the order of a nanosecond, we need very fast pulse generators, high-speed scopes, and very high-speed probes. Alternatively, we can use the TDR measurement capabilities of the Tektronix[®] 8000 series scopes (TDS8000, CSA8000, or CSA8200) joined with the 80E04 TDR sampling module. The MAX9979EVKIT (evaluation kit), Hewlett Packard 8082A Pulse Generator, and the TDS8000/80E04 were used to demonstrate this approach. **Figure 2** shows part of the MAX9979EVKIT



Figure 2. MAX9979EVKIT (partial).

board. Any high-speed scope with TDR capability and any high-speed differential pulse generator can be substituted; similar results will be obtained.

The measurements that will be made in this analysis are:

- The delay from the DATA1/NDATA1 SMA edge connectors on the PCB, to the input pins DATA1/ NDATA1 of the MAX9979 IC.
- The delay from the MAX9979's DUT1 (device under test) output through the SMA connector J18.
- The delay in the test cable that connects the DUT1 output to the CSA8000.
- The entire delay from the DATA1/NDATA1 inputs to the DUT1 output through the cable and to the CSA8000.
- Finally, the MAX9979's true delay will be calculated.

Modeling the DATA1/NDATA1 inputs

Because TDR responses can be confusing, we will first model the input delays using a SPICE simulator. Then we will compare this simulation to the real measurements taken. See **Figure 3**.

Note from Figure 3

- The PCB run delays are modeled as 6in long with 65Ω impedance. This is, in fact, the real impedance of the DATA1/NDATA1 PCB runs. Ideally, these should have been 50Ω , but as will be seen on the TDR measurement, they are 63Ω .
- The NDATA1 output has been terminated to ground. Since DATA1 and NDATA1 are symmetrical and have identical lengths to the pins of the MAX9979, we will only measure the DATA1 PCB run.
- A 12in cable from the generator is modeled, but it will prove unnecessary in the actual propagation delay measurement.

Simulation of DATA1/NDATA1 input

Figure 4 displays the SPICE simulation of the waveform at TPv3.

Several observations can be made from the data in Figure 4.

• The input signal is a step function. The step amplitude in this simulation is 0.5V. This emulates the TDR signal generated by the CSA8000.



Figure 3. Equivalent input schematic and final simulated model.

- Time represents the delay of the various elements in our model:
 - a. Step 1 represents the 12in cable from the generator. The delay time is about 3ns, which represents twice the actual delay. The delay of this cable is 1.5ns.
 - b. Step 2 represents the DATA1 PCB run. The delay is approximately 2ns. The PCB delay is half of this, or 1ns.
 - c. The rest of the delays are the reflections of the pulse through the DATA1 PCB run.
- The Y axis represents the impedance of these various elements. The unit is in volts and can be translated into impedance.
- The X axis is the time for the simulated signal reflections due to the single-input step signal. Refer to Figure 1 to compare the signals. The length of these signals represents the delays through the various elements.

Propagation delay measurements of the MAX9979

Follow these six steps to make the propagation delay measurements.

Step 1. Measure the delay of a 2in SMA cable that attaches the DUT1 node to the CSA8000 vertical input (**Figure 5**).

To make this measurement:

- Connect the 2in SMA-to-SMA cable to one input of the 80E04 TDR module; leave the other end open.
- Use the TDR pulldown menu to make the measurement.
- Notice that this looks exactly like the "open" example in Figure 1. The delay measured here is 804ps. As this is twice the delay of the cable, the cable length is 402ps.



Figure 4. SPICE simulation of the model shown in Figure 3 (node TPv3). Data were gathered at the DATA1 input of the MAX9979EVKIT.

- Also note that the second step is exactly halfway between the top and bottom steps. Recalling the TDR basics, this indicates that our 2in cable is a true 50Ω .
- This 2in cable is one of the delay paths in our measurement.

Step 2. Measure the delay/impedance of the DATA1 input signal's PCB run.

There are several observations to make from this data.

- **Figure 6** is identical to the simulated plot shown in Figure 4. This confirms the accuracy of our model.
- The cursors are set to measure the impedance of the line. The first step is 49.7Ω, which is the cable from the CSA8000. We expect this.
- The second cursor shows 97.8Ω, and this is the impedance of the MAX9979's internal 100Ω resistor across DATA1/ NDATA1 (see Figure 3). We also expect this.
- The impedance in the second step is not 50Ω . This step is the DATA1 PCB impedance, and about 63Ω . This tells us that the PCB run for DATA1 and NDATA1 was not designed to be 50Ω , as we would have expected.



Figure 5. CSA8000 TDR of a 2in SMA cable.



Figure 6. DATA1 PCB TDR impedance measurement.

• The large amplitude is 150Ω which is the addition of the 50Ω cable and the 100Ω resistor. This only occurs for the third reflection.

To make this measurement, simply:

- Connect one end of the 12in SMA cable to the CSA8000. Connect the other end of the cable to the DATA SMA input connector on the MAX9979EVKIT.
- Ground the NDATA1 SMA connector with an SMA ground. This is depicted in Figure 3. The length of the 12in SMA cable is irrelevant to the propagation delay measurement, but should be as short as possible.
- It is not necessary to power up the MAX9979EVKIT. This measurement was made with the MAX9979 soldered onto the board but with no power applied. Some users prefer to make this measurement without the device soldered onto the board. Disconnecting the MAX9979 results in a cleaner three-step signal, simulating an open condition as shown in Figure 1. The actual time measurement is the same in both configurations.

With **Figure 7**, we are measuring the second step—the delay of the DATA PCB run. Note:

- The first step was the cable, and we are not interested in its delay.
- The measurement is 1.39ns and our PCB delay is half this, or 0.695ns. This delay is admittedly larger than our modeled delay, but we were only estimating the delay in the model to be close for comparison purposes.
- The measurements are made between dips in the signal. These dips represent capacitances created by the board SMA and the DATA1 pin of the MAX9979. Consequently, the measurement is made between these dips to ensure that we get the SMA and the PIN delays. Also note that there is a bump: the inductance of the SMA connection to the board. So again, the



Figure 7. The same plot as Figure 6, but expanded and measuring delay.

measurement is made before this bump to ensure that we also capture the full board delay. Further reading of TDR measurements will highlight these capacitance and inductance dips and bumps.

Step 3. Measure the delay/impedance of the DUT1 output signal's PCB run.

The scope trace in **Figure 8** data was generated from the same setup as in Figures 6 and 7. We are now using a 2in SMA cable connected to the CSA8000 80E04 module and the DUT1 SMA on the MAX9979EVKIT. Note:

- The first step represents the 2in cable. The TDR signal is 0.5V and the first step is 250mV. This indicates that our cable impedance is 50Ω , as expected.
- The measurement for the DUT1 delay is made between the two dips, as was explained in the DATA1 measurement above. Note, however, that the level between these dips is also 50Ω . This value now tells us that the short DUT1 PCB metal run is very close to the ideal 50Ω .
- Recall that the DATA1 run had an impedance of 63Ω and that the DUT1 node had an impedance of 50Ω. This means that the metal width on the DATA1 inputs was narrower than the DUT1 output. Ideally, they should be the same. The TDR measurement found this difference, which may not be an error. The slightly higher impedance of the DUT1 run, created by the narrower metal run, also reduces the capacitance of the DATA1 metal run. The data run is the longest run, and to ensure maximum bandwidth this capacitance needs to be as low as possible.
- It is difficult to measure the DUT1 PCB delay. Its impedance appears the same as the cable. If the MAX9979 was not soldered to the board, we would have seen the "open" three-step signal. But we still can measure this delay with the MAX9979 soldered in place.



Figure 8. DUT1 PCB TDR delay and impedance measurement.

Examining the capacitive dips reveals both where the SMA connector is soldered to the board, and where the dip of the MAX9979's DUT1 pin is. We also look for the inductive bump of the SMA connector and ensure that this is within the two dips. Once we resolve these issues, we see that the delay was 360ps. Now we halve this value to obtain the actual DUT1 PCB board delay. This delay is 180ps.

Step 4. Set up the differential signal generator with two identical SMA cables, and measure the baseline delay on the CSA8000.

In **Figure 9**, C1 and C2 are two complementary PECL signals with amplitudes of approximately 450mV. These are our DATA1 and NDATA1 signals fed directly from the external generator to the inputs of the CSA8000. We are using the 20GHz sampling heads of the CSA8000. Several observations can be made from this data:

- M1 is a mathematical calculation of the differential signals C1 C2. The amplitude is 900mV, and the 10%/90% rise and fall times are close to 700ps. So this means that we have a clean set of DATA1/ NDATA1 signals.
- We are also measuring the Crs or zero-crossing point of the differential signal M1. This is noted as 29.56ns. The scope is triggered, and all we want is one of these crossing points. We will power up the MAX9979 and measure the same crossing point, as it is delayed through the entire board.
- This delay also includes the delay of our two input cables. Because these cables will also be used to measure the signal's delay through the board, their delays cancel out. Nonetheless, it is still best to use the smallest length cables, but their delays are not important for the propagation delay measurement.



Figure 9. Measurement of the DATA1/NDATA1 signal from the generator.

Step 5. Power up the MAX9979EVKIT.

Take the DATA1 and NDATA1 signals and connect them to the powered-up MAX9979EVKIT DATA1/NDATA1 inputs. Use the same cable as in Step 4. Set the MAX9979 to the specified 0V to 3V signal and terminate the output into 50Ω , as specified in the data sheet for the propagation delay measurement. In this case, the 50Ω load is the input to the CSA8000. Data points taken from **Figure 10** then show:

- The output signal amplitude is now 0V to 1.5V. This is expected and is divided down by a factor of two by the 50Ω load.
- The rise and fall times are well within the specifications for the MAX9979. We are, therefore, assured that we have a good, clean, valid output driven by a very clean and valid DATA1/NDATA1.
- The setup for the CSA8000 remains as in Step 5. The trigger is the same, as in Step 4. Now we see that the zero-crossing point is 33.77ns.

Step 6. Calculate the MAX9979's propagation delay.

The total delay through the MAX9979EVKIT was:

33.77ns - 29.56ns = 4.21ns.

To make this measurement:

- Subtract the DATA1 PCB run of 0.695ns, and our delay is now 3.515ns.
- Subtract our DUT1 PCB run of 0.18ns, and our delay is now 3.335ns.
- Subtract the delay of the 2in cable to the CSA8000. This delay was 402ps, so our delay is now 2.933ns.

The nominal delay in the MAX9979 specifications for this setup is 2.9ns. Here we have established that the delay for the MAX9979 soldered to the board of this EV kit is 2.933ns, which is very close to what was expected.



Figure 10. MAX9979 powered up and generating a 3V signal into the 50 Ω load of the CSA8000.

Summary

This analysis has demonstrated that using TDR measurements for propagation delay offers several advantages:

- Very accurate propagation delay measurements.
- No active probes (avoids the inaccuracies that they introduce) are needed.
- Simple techniques can be used for most propagation measurements.
- Impedance measurements ensure correct impedances for connectors and PCB runs.
- Excess capacitance and inductance in the signal path can be analyzed with the TDR signals, and then used as feedback to redesign the board if necessary.

- Simplified modeling and simulation tools ensure correct interpretations and confirm the measurement setups.
- Use of good practices in measuring critical specifications.

As signal speeds rise, the errors and mistakes in timing measurements can cause incorrect planning decisions, device selections, and system design. The use of good practices in high-speed measurements is always a good subject to revisit. This article emphasizes these good practices.

Tektronix is a registered trademark of Tektronix, Inc.

Use thermal analysis to predict an IC's transient behavior and avoid overheating

Milind Gupta, Member of Technical Staff, System & Power Management Business Unit

Da Weng*, Product Definer, System and Power Management Business Unit

Maxim Integrated Products, Inc.

This article presents a method for predicting thermal behavior in ICs. This information will be especially helpful for the PMICs (power-management ICs) used in automotive applications and other high-temperature environments. After characterizing thermal behavior, we formulate a mathematical model that simulates transient temperatures within the chip. We introduce physical laws governing thermal behavior and evaluate them for use in the thermalbody models defined for an IC. Based on that analysis, we then propose an equivalent passive RC network for modeling an IC's transient thermal behavior. To illustrate an application for the proposed analysis, we devise an RC network for an LED driver (the MAX16828). We conclude with insights on the use and usefulness of this approach, and suggest ways to speed the creation of the RC models.

Designers often need to know the thermal behavior of an IC, especially for the PMICs (power-management ICs) used in automotive applications. When a particular IC operates at a high temperature (such as +125°C), does it trigger the thermal-shutdown circuitry or exceed the product's safe operating temperature? Without a definite method of analysis, we cannot offer a reliable answer. Therefore, when defining a new IC, we need a way to predict thermal shutdown or excessive die temperature based on complex internal functions.

For operation in DC mode, you can often determine the junction temperature using data-sheet parameters such as θ_{JA} (thermal resistance) and θ_{JC} (thermal junction temperature).¹ However, to predict how high the junction temperature will peak for modes other than DC (such as a power MOSFET driven by a PWM signal to control LEDs or a switching regulator), you need transient thermal data. Although useful, that data is not typically found in data sheets. You might also ask how long the chip can operate at a given power-dissipation level before encountering trouble? That question is also difficult to answer.

This article derives equations that use power dissipation and the ambient temperature to predict the junction temperature of a chip as a function of time. The article begins by introducing the physical laws upon which the analysis is based. The discussion continues by defining an IC system as a complex, layered thermal body. The thermal-body model is then analyzed theoretically, and equations to govern transient thermal behavior are derived. Based on these equations, the article proposes an equivalent RC passive network that represents the IC's thermal characteristics. Finally, to demonstrate the usefulness and accuracy of this analysis, experimental results are shown for a high-voltage, linear HB LED (high-brightness LED) driver with PWM dimming, the MAX16828.

Laws of thermal dynamics

For any object we can derive the required relations for temperature vs. time by using two principal laws.

Newton's law of cooling:

$$\frac{dT_B}{dt} = -k_A(T_B - T_A)$$
 (Eq. 1)

Where:

T_B is body temperature.

 T_A is ambient temperature. k_A is a constant of proportionality (> 0).

t is time.

Law of conservation of nonlatent energy:

$$mc\Delta T = Energy = P\Delta t$$

$$\Rightarrow \frac{\Delta T}{\Delta t} = \frac{dT}{dt} = \frac{P}{mc}$$
(Eq. 2)

Where:

P is constant power generated or imparted to the body. m is the mass of the body.

c is the specific heat capacity of the body.

Combining these laws, we have:

$$\frac{dT_{\rm B}}{dt} = \frac{P}{mc} - k_{\rm A}(T_{\rm B} - T_{\rm A})$$
(Eq. 3)

The data sheet for an IC normally lists thermal data for the package, such as θ_{JA} . That data lets us analyze the steady-state thermal equilibrium for a package to see if it agrees with Equation 3:

$$\frac{dT_{\rm B}}{dt} = 0$$
 at the steady state

Therefore:

$$P = mck_A(T_B - T_A)$$
(Eq. 4)

Equation 4 can be rewritten as:

$$P = \frac{(T_B - T_A)}{\theta_{BA}}$$
(Eq. 5)

Where:

 θ_{BA} is thermal resistance - body to ambient. T_B is the temperature inside the package. T_A is the ambient temperature outside.

Thus:

$$\theta_{BA} = \frac{l}{mck_A}$$
(Eq. 6)

Defining the chip as a thermal system

A clear definition of the system is very important because the thermal result depends on that definition. In the cross section of a chip mounted on a PCB (**Figure 1**), we see at least three different materials in the path from die to environment: the die itself, the mold epoxy, and the package. Thermal models are based on one of two patterns of heat flow, depending on the location of the dominant heat source: flow from an external source to the die (when the external source is the dominant heat generator), and flow from the die to the environment (when the die is the dominant heat generator). We will discuss each of these patterns of heat flow in turn.

Heat flow from an external source to the chip

Consider the system of **Figure 2**, which shows a uniform body gaining energy (heat) from a power source and losing energy to the environment.

Heat reaches the internal die through the package and the mold compound. Therefore, this system also models thermal transients in the chip for heat sources outside the package. The package normally has a much higher thermal resistance than the die itself because the die has



Figure 1. A cross section of a chip mounted on a PCB shows the layers of material between the die and the environment.

lots of metal on it. The die, therefore, tracks the package temperature with almost no lag, thus causing the chip to behave as a single body. We can define this one-body system immediately by using Equation 3. Solving for T_B , we have:

$$T_{\rm B} = \frac{P}{mck_{\rm A}} + T_{\rm A} + k_{\rm o}e^{-k_{\rm A}t}$$
 (Eq. 7)

Where k_o is the constant of integration, which is solved according to the initial conditions. In general, this equation is useful for defining the thermal transient of a chip when the heat source is outside the chip.

We can illustrate this model with an example. To determine the thermal transient for a chip whose initial temperature is T_i , substitute t = 0 and $T_B = T_i$ in Equation 7:

$$k_{o} = T_{i} - \frac{P}{mck_{A}} - T_{A}$$
 (Eq. 8)

Therefore:

$$T = \frac{P}{mck_{A}} + T_{A} + (T_{i} - \frac{P}{mck_{A}} - T_{A})e^{-k_{A}t}$$
(Eq. 9)

Considering the special case for which $T_i = T_A$:

$$\Gamma = T_A + \frac{P}{mck_A} (1 - e^{-k_A t})$$
 (Eq. 10)

Using Equation 6, we can rewrite Equations 9 and 10:

$$T = \theta_{JA}P + T_A + (T_i - \theta_{JA}P - T_A)e^{-k_A t}$$
(Eq. 11)

$$T = T_A + \theta_{JA} P(1 - e^{-k_A t})$$
 (Eq. 12)

Equations 11 and 12 are useful for predicting chip temperature (either package or die) when the heatgenerating source is outside the package. One example could be a nearby high-current MOSFET that dissipates lots of heat.



Figure 2. This thermal model illustrates the flow of heat from an outside power source to the chip (BODY 1) and then back out to the environment.

When we know k_A and θ_{JA} we can calculate the temperature at different times. Alternatively, if P is a complex function of time, we can use the above equations to evaluate temperature as a time-based simulation and use MATLAB[®] software to write a program that plots temperature as a function of time.

The θ_{JA} value is provided in data sheets. However, when a setup imposes conditions other than those of the JEDEC standard, that published θ_{JA} value for these calculations can cause errors. The JEDEC standard 51-3 states, "It should be emphasized that values measured with these test boards cannot be used to directly predict any particular system application performance, but are for the purposes of comparison between packages."² Thus, to properly estimate temperature, you should either measure θ_{JA} for the prototype board or estimate it directly as explained below.

Heat flow from a die to the environment

Consider the system of **Figure 3**, in which a three-body system (similar to a chip) generates heat on the die and dissipates it through the epoxy and package to the environment. Body 1 is the die, Body 2 is the epoxy, and Body 3 is the chip package.

To solve for θ_{JA} in this system, we must define the equations for all three bodies.

Body 1:

$$\frac{dT_{B1}}{dt} = \frac{P_G}{m_1 c_1} - k_1 (T_{B1} - T_{B2})$$
(Eq. 13)

Body 2:

$$\frac{dT_{B2}}{dt} = \frac{P_{12}}{m_2 c_2} - k_2 (T_{B2} - T_{B3})$$
(Eq. 14)

Body 3:

$$\frac{dT_{B3}}{dt} = \frac{P_{23}}{m_3 c_3} - k_3 (T_{B3} - T_A)$$
(Eq. 15)

Where:

 T_{B1} , T_{B2} , T_{B3} are the instantaneous temperatures of Bodies 1, 2, and 3.

 P_{12} is power in the form of heat transferred from Body 1 to Body 2.

 P_{23} is power in the form of heat transferred from Body 2 to Body 3.

 P_G is the power generated on Body 1 or directly transferred to Body 1.

Power generated by the die (P_G) minus power absorbed by the die is:

$$P_{12} = P_{G} - m_{1}c_{1}\frac{dT_{B1}}{dt}$$
(Eq. 16)

Power received by the epoxy minus power absorbed by the epoxy is:

$$P_{23} = P_{G} - m_{I}c_{1}\frac{dT_{B1}}{dt} - m_{2}c_{2}\frac{dT_{B2}}{dt}$$
(Eq. 17)

Substituting Equations 16 and 17 in Equations 13, 14, and 15:

$$\frac{dT_{B1}}{dt} = \theta_{12}k_1P_G - k_1(T_{B1} - T_{B2})$$
 (Eq. 18)

$$\frac{dT_{B2}}{dt} = \frac{\theta_{23}}{\theta_{12}} k_2 (T_{B1} - T_{B2}) - k_2 (T_{B2} - T_{B3})$$
(Eq. 19)

$$\frac{dT_{B3}}{dt} = \frac{\theta_{3A}}{\theta_{23}} k_3 (T_{B2} - T_{B3}) - k_3 (T_{B3} - T_A)$$
(Eq. 20)

The solution of this three-body system in Equations 18, 19, and 20 can be complicated, but the use of Laplace transforms makes it easier. The form of the solution is:

$$T_{B1} = T_1 e^{m_1 t} + T_2 e^{m_2 t} + T_3 e^{m_3 t} + T_A + (\theta_{12} + \theta_{23} + \theta_{3A}) P_G \quad (Eq. 21)$$

Where:

 $\begin{array}{l} \theta_{12} \text{ is the thermal resistance from Body 1 to Body 2.} \\ \theta_{23} \text{ is the thermal resistance from Body 2 to Body 3.} \\ \theta_{3A} \text{ is the thermal resistance from Body 3 to the environment.} \\ T_1, T_2, \text{ and } T_3 \text{ are the constants of integration.} \\ m_1, m_2, \text{ and } m_3 \text{ are functions of } k_1, k_2, \text{ and } k_3. \end{array}$

Equation 21 predicts die temperature in a very accurate way when the die is generating power. To use this equation, however, we must know all the constants of integration



plus m_1 , m_2 , and m_3 , which are complicated functions whose solution is difficult. To avoid this difficult exercise, we use a tool for solving differential equations: SPICE.

RC network models' thermal-transient differential equations

We will now propose a circuit modeled by similar differential equations, and we will then simulate the circuit and read out temperatures from the simulation.

The differential Equations 18, 19, and 20 can be modeled by a simple RC network (**Figure 4**) that represents the power generated on the die.

In Figure 4 initial voltages on the capacitors represent the initial temperatures of the die (C₁), the epoxy (C₂), and the package (C₃). V_A represents the ambient temperature of the environment, and I_S (the current going into capacitor C₁) represents the power generated on the die. The differential equations representing voltages on the capacitors are:

$$\frac{dV_{C1}}{dt} = \frac{I_s}{C_1} - \frac{(V_{C1} - V_{C2})}{R_1 C_1}$$
(Eq. 22)

$$\frac{dV_{C2}}{dt} = \frac{(V_{C1} - V_{C2})}{R_1 C_2} - \frac{(V_{C2} - V_{C3})}{R_2 C_2}$$
(Eq. 23)

$$\frac{dV_{C3}}{dt} = \frac{(V_{C2} - V_{C3})}{R_2 C_3} - \frac{(V_{C3} - V_A)}{R_3 C_3}$$
(Eq. 24)

These three equations correspond to Equations 18, 19, and 20, with the following substitutions of variables:

 $V_{C1} \leftrightarrow T_{B1}, V_{C2} \leftrightarrow T_{B2}, V_{C3} \leftrightarrow T_{B3}, I_{S} \leftrightarrow P_{G}$

The capacitor voltages correspond directly to the temperatures of the die, epoxy, and package. Any SPICE package can simulate the RC circuit easily. When we know the proper values of R_1 , R_2 , R_3 , C_1 , C_2 , and C_3 modeled for a particular chip, we can then simulate the circuit and directly read out die temperature as the voltage on capacitor C_1 .



Figure 4. This RC network models the transient-thermal behavior of a chip when heat is generated internally.

Now we can determine the passive component values for a particular chip (R_1 , R_2 , R_3 , C_1 , C_2 , and C_3). Use Equation 5 (repeated below as Equation 25) to obtain the thermal resistance for the system (θ_{JA}) by measuring the die's steady-state final temperature:

$$\theta_{JA} = \frac{T_J - T_A}{P_G}$$
(Eq. 25)

Where:

T_J is the steady-state junction temperature of the die.

T_A is the ambient temperature.

 P_G is the power dissipated on the die.

Operating with the same dissipated power (P_G) as in Equation 25, you can create a data set for the chip's transient temperature variation by measuring the die temperature at different times starting at time 0. Then, based on the following constraint, do a curve-fitting exercise on the measured data to determine the values of R_1 , R_2 , R_3 , C_1 , C_2 , and C_3 :

$$\theta_{\rm JA} = R_1 + R_2 + R_3 \tag{Eq. 26}$$

Measuring the die temperature

There are a couple of practical methods to measure the die temperature of an integrated circuit.³ Here we will use the ESD diode forward-drop measurement method to determine the chip temperature, since it is easy and will not introduce a large amount of error. However, to ensure that the accuracy levels of the measurement remain within acceptable limits, always choose the die-temperature measurement technique for a particular chip carefully. The following guidelines will prove helpful.³

- Make sure that the ESD diode chosen for measurement does not have a large parasitic resistance, nor a large current flow that would offset the diode drop read-out. It is best to discuss this with the IC manufacturer to determine the estimated maximum internal bond-wire and metallization resistance.
- 2. Also make sure that the ESD diode is near the hotspot of the chip or in the area where you are actually concerned with the temperature. This configuration will provide better estimation of the temperature and deliver more accurate results.
- 3. If you are choosing a FET's on-resistance as a temperature indicator, make sure that the FET is fully on and in dropout mode at the measurement point.

To use the ESD diode forward-drop approach, we need a diode on the chip to which we can apply forward bias and measure its voltage. That can easily be done on most chips with an ESD diode connected between a pin and the supply voltage. Because the measured data gives us the diode



Figure 5. The forward voltage for a diode biased at constant current varies with temperature.

voltage, we must also consider the relationship between a diode voltage and temperature.⁴

Diode voltage decreases with a nearly constant slope and negligible deviation. If plotted with respect to temperature, the result would look like the plot in **Figure 5**.

In Figure 5, T_A is the ambient temperature and V_{DA} is the diode voltage at ambient temperature. We, therefore, know one point on the graph and its slope. Slope can be derived by measuring the diode voltage at different temperatures in a temperature-controlled oven. Alternatively, you can use a number like 2mV/K, which is valid with minimal error for a wide range of diode currents.⁴ These numbers should apply to any other chip as well, but for accuracy it is always better to measure the slope associated with the current intended for biasing the diode. Any temperature can now be represented in terms of the diode voltage:

$$T = \frac{V_{\rm D} - (V_{\rm DA} - sT_{\rm A})}{s}$$
(Eq. 27)

Where:

T is the temperature for which the diode voltage is V_D . s is the slope of the graph (s < 0).

Substituting this expression in Equations 11 and 12 yields the following:

$$V_{\rm D} = s \,\theta_{\rm JA} P + V_{\rm DA} + (V_{\rm Di} - s \theta_{\rm JA} P - V_{\rm DA}) e^{-k_{\rm A} t}$$
(Eq. 28)

$$V_{\rm D} = V_{\rm DA} + s\theta_{\rm JA}P(1 - e^{-k_{\rm A}t})$$
 (Eq. 29)

Substituting in Equations 18, 19, and 20 also yields:

$$\frac{dV_{D1}}{dt} = s\theta_{12}k_1P_G - k_1(V_{D1} - V_{D2})$$
(Eq. 30)

$$\frac{dV_{D2}}{dt} = \frac{\theta_{23}}{\theta_{12}} k_2 (V_{D1} - V_{D2}) - k_2 (V_{D2} - V_{D3})$$
(Eq. 31)

$$\frac{dV_{D3}}{dt} = \frac{\theta_{3A}}{\theta_{23}} k_3 (V_{D2} - V_{D3}) - k_3 (V_{D3} - V_{DA})$$
(Eq. 32)

To apply our RC network properly for curve fitting the measured voltage-transient data for the diode, now we only need to set the magnitude of the current source as:

$$I_s = sP_G$$

(Eq. 33)

Because s < 0, you can realize Equation 33 by reversing the current source direction and setting its magnitude to $|sP_G|$.

Experimental determination and verification of the RC network

We can demonstrate a practical application of the RC simulation model using the equations derived above and a linear LED driver like the MAX16828/MAX16815. These chips operate up to 40V using few external components, and the MAX16828 supplies an LED string with up to 200mA (**Figure 6**). The MAX16815 is pin-compatible with the MAX16828 and similar in function, but maximum output current is 100mA instead of 200mA.

Both LED drivers are suitable in automotive applications such as side lighting, automotive exterior rear combination lights, backlighting, and indicators. The MAX16828 can dissipate considerable heat if the internal MOSFET sees high current combined with a large dropout voltage. (The MOSFET does this when the LED string's forward voltage is low.) The voltage across R_{SENSE} is regulated to 200mV ±3.5%, which allows that resistor to set the LED current. The chip's DIM input provides a wide range of PWM dimming for the LEDs and, because it also withstands high voltages, it can connect directly to the IN pin.

To obtain a direct indication of the die temperature, we measure the forward-bias voltage of an internal ESD diode connected between the DIM and IN pins. This diode is biased at ~100 μ A, causing its forward voltage to vary 2mV/K. (This can be confirmed by heating the part in a temperature-controlled oven.) **Figure 7** shows the setup for these experiments. The 5V source and 56k Ω resistor provide 100 μ A for forward biasing the ESD diode.



Figure 6. Typical application circuit for the MAX16815/MAX16828 HB LED drivers.



Figure 7. The test setup shown lets you measure transient die temperatures using an on-chip ESD diode. *EP indicates an exposed pad.

The driver is programmed to deliver 200mA of output current for the LEDs.

In this state the part carries a lot of current and our ESD diode measurement is in the path of that measurement. Consequently, we will get some error due to the parasitic resistance of the bond wire and internal metallization. From the internal layout and calculation of the length of bond wire, the worst-case parasitic resistance is estimated to be $50m\Omega$. With 200mA, this parasitic resistance will cause an error of around $\pm 10mV$ (max) in our diode reading. Our accuracy error will be larger than $\pm 5^{\circ}$ C. Additionally, the ESD diode on the die is placed adjacent to the on-chip power MOSFET device and thermal-shutdown circuitry. This configuration makes the diode the best indicator of that region's temperature.

System definition 1

This next section describes how you can use a test setup to capture transient-thermal diode voltages for use in the system-definition equations presented above in Equations 7 and 21.

To calculate k_A and θ_{JA} (for substitution in Equation 11), we heat the chip using a hot-air gun. The chip should be powered off because we do not want to generate internal heat. Heating the part with a hot-air gun causes the temperature of the package and die to rise together. You can monitor the die's temperature change by measuring the diode voltage on a scope (**Figure 8**).

When the chip is heated, the diode voltage decreases with an exponential rate of change as the equation predicts. Near the center of the curve the hot-air gun is switched off, causing the package and die to begin cooling. The diode voltage rises, again following an exponential curve.

We do not know exactly how much heat is imparted from the heat gun to the chip. Therefore, to eliminate that unknown we first adjust Equation 28 to fit only the rising



Figure 8. This diode-voltage transient includes exponential curves that represent heating with an external heat gun (falling curve) and cooling by removal of the heat gun (rising curve).

(cooling) part of the curve (Figure 8). This curve-fitting exercise lets us estimate the best value for k_A . With no heat power transferred to the package during cooling, the package is simply cooling down with P = 0. Equation 28, therefore, simplifies to:

$$V_{DB} = V_{DA} + (V_{Di} - V_{DA})e^{-k_A t}$$
 (Eq. 34)

We know the values for V_{DA} (643mV from the initial measurement at room temperature) and V_{Di} (the reading for t = 0 reference). To determine k_A , we must just adjust



Figure 9. Equation 34, fitted to a couple of diode-voltage measurements, closely tracks all the diode measurements for a chip that is cooling after being heated with a heat gun.





the equation so that it includes a couple of readings on the rising curve. This exercise yields $k_A = -0.0175$. A graph of the readings (diode voltages in mV, with respect to time in seconds) and Equation 34 with the above k_A is shown in **Figure 9**.

As we can see in Figure 9, Equation 34 closely follows the measured data for $k_A = -0.0175$. To verify that our equations are correct, we try to fit the falling curve on Equation 28 with the value determined for k_A . The equation fits very accurately (**Figure 10**). Thus, we see that Equation 34 for the system discussed in *System definition 1* closely matches the experimental data.

System definition 2

Verification of the system 2 Equations 30, 31, and 32 is more difficult. We must generate heat on the die, measure the die temperature using the diode forward voltage, and fit that temperature value to a simulated value for the C1 voltage of the proposed RC network. This task is accomplished by writing a program using MATLAB.

It is important to record the thermal transient at a time for which the initial temperature of the whole chip is known. In that way we also know the initial capacitor voltages for solving the RC network. Using the same test setup (see Figure 7), we now turn on the current and capture the diode voltage on a scope (**Figure 11**).

Similar transients are recorded for three different powerdissipation levels, and one curve is fitted to that data. The circuit in **Figure 12** is the result of fitting from the first set of data in which the power dissipation is 1.626W. The graph in **Figure 13** compares the measured and fitted data. Similarly, the graph in **Figure 14** shows how the RC network fits the second set of readings (power dissipation



Figure 11. A forward-voltage transient from the MAX16828's internal diode signals that an on-board MOSFET has turned on and is generating heat.

of 2.02W). The graph in **Figure 15** shows how it fits the third set of readings (power dissipation of 1.223W).

These experimental results show that the measured results closely match the theoretical model. Such modeling is useful for simulating the transient temperature of an IC, once you have modeled the RC network for that particular chip. The model can also be used for chips of similar size to determine their thermal characteristics during the definition phase. That capability can indicate the operational limitations of the chip. That information, in turn, helps you define the chip's operational modes to prevent overheating.

Conclusion

This article has described a way to model the thermal behavior of a chip as an RC network, which can then



Figure 12. With component values as shown, this RC network models the chip's thermal transient when heat is generated on the die.



Figure 13. Measured vs. curve-fitted results for the chip's heating curve when the die is dissipating 1.626W.

be simulated easily using a SPICE tool. The following measures can improve the accuracy of this model:

- Take data at both extremes of power dissipation and at one level in the middle. Fitting the RC network to all three levels simultaneously makes the model usable for most practical power-dissipation levels.
- Improve model accuracy by collecting data points at different ambient temperatures.



Figure 14. Measured vs. curve-fitted results for the chip's heating curve when the die is dissipating 2.02W.



Figure 15. Measured vs. curve-fitted results for the chip's heating curve when the die is dissipating 1.223W.

These exercises should improve accuracy when necessary, but most applications do not require that you know the temperature with high accuracy. Applications and design engineers, as well as system designers may find the method useful. For more detailed chip information, a company can create RC network models for its ICs and make them available with the chips' corresponding SPICE models.

References

1. "Package Thermal Resistance Values (Theta JA and Theta JC) for Dallas Semiconductor Temperature Sensors," Maxim application note 3930.

- 2. "Package Thermal Characteristics," Actel Corporation application note AC220, (February 2005).
- 3. Rako, Paul, "Hot, cold, and broken: Thermal-design techniques," *EDN* online (3/29/2007).
- Pease, Bob, National Semiconductor, "The Best of Bob Pease. What's All This VBE Stuff, Anyhow?" (11/5/2008).

A similar article appeared in EDN Magazine in January 2010.

MATLAB is a registered trademark of The MathWorks, Inc.

*Da Weng is no longer with Maxim.

Selecting high linearity mixers for wireless base stations

Stephanie Overhoff, Field Applications Engineer

Maxim Integrated Products, Inc.

Today's communication systems like wireless base stations make strong demands on receiver sensitivity and largesignal performance. This article focuses on mixers, and describes key mixer performance issues and the basic parameters specified in data sheets. The article explains how to select the best mixer to optimize the receive channel.

Introduction

The communication standards for wireless base stations—e.g., GSM, UMTS, and (now) LTE—define minimum specifications for various parameters including receiver sensitivity and performance in the presence of large signals. These key requirements make heavy demands on every functional block of the radio in a wireless base station. In the receive signal path, mixer performance has a major impact on the receiver's sensitivity and large-signal performance. This article describes key mixer performance issues and parameters to help you select the best mixer for your receive channel.

A wireless base-station receiver

To start, we first analyze the block diagram of a typical receiver used in wireless base stations (**Figure 1**). These receivers are called superheterodyne receivers

because the received signal undergoes two consecutive downconversions to lower frequencies. As shown, the signal is received by the antenna and then filtered by RF filter 1, which is normally used to filter out trash signals. This filter output is then amplified by an LNA (low-noise amplifier), which normally has a very low noise figure.

The amplified signal is again filtered, this time by RF filter 2, which limits the frequency range while filtering out unwanted signals that can limit the mixer's performance. The filtered and band-limited signal is then fed to the first mixer, where it is downconverted to an IF frequency by mixing with an LO's (local oscillator's) signal. Depending on the receiver's architecture, this IF signal can be further downconverted to a second, lower-IF frequency, and then demodulated for processing in the baseband.

We now examine the mixers in this receiver chain. The mixers' parameters should be investigated because they have a major effect on the receiver's sensitivity and largesignal performance.

Mixer parameters

The noise figure of a mixer describes the degradation in SNR (signal-to-noise ratio) from input to output. That ratio is normally expressed in the logarithmic measure of decibels (dB), as shown in Equation 1:

$$NF = 10 \log \frac{SNR_{RF}}{SNR_{IF}} \quad [dB]$$
 (Eq. 1)

A second important parameter is the conversion gain (or alternatively, conversion loss). Conversion gain gives an important indication whether the mixer configuration is active or passive. Passive mixers have insertion loss (called conversion loss) because they include no components for amplifying the signal. Active mixers, however, have active components and provide conversion gain.



An active mixer can be realized in either of two configurations: as an integrated mixer based on a balanced (Gilbert cell) design; or as a passive mixer that is combined with an IF amplifier stage to provide gain instead of loss. Because the integrated mixer has gain, it requires no external IF amplifier stage to make up for insertion loss.

Conversion gain / loss = G =
$$\frac{P_{RF}}{P_{IF}}$$
 [dB] (Eq. 2)

Conversion gain (or loss) is a logarithmic measure expressed in dB, as seen in Equation 2. It is frequency dependent, and should be specified over the mixer's entire operating frequency range. To ensure optimal receiver performance, the variation of conversion gain/loss over the specified frequency range should be as small as possible.

Because wireless base stations usually operate in a fluctuating temperature environment, the conversion gain/ loss should also be specified over the operating temperature range, again with as small a variation as possible. This temperature range is important because under normal conditions a small variation with temperature allows a smaller headroom, which is useful in system planning.

The large-signal behavior of a mixer is described by a mixer parameter called the "1dB compression point," also called compression point (IP1dB), and the secondand third-order intercept points (IP₂ and IP₃). The IP1dB compression point predicts the level of input power at which the mixer gain is reduced by 1dB, with respect to the linear expression in Equation 3:

$$P_{OUT} = G \times P_{in}$$
 (Eq. 3)

A mixer should also be able to convert a weak signal when two large signals of nearly the same frequency are applied to the mixer's input. This behavior is normally described by the third-order intercept point (IP₃), which together with the noise figure describes the dynamic range of the mixer. A large IP₃ indicates a high-linearity mixer. The mixer's data sheet should also specify intercept points for the mixer's input and output. Using Equation 4, you can calculate the OIP₃ (output intercept point) from the IIP₃ (input intercept point), and vice versa:

$$OIP_3 = IIP_3 + G \tag{Eq. 4}$$

Where OIP_3 is the intercept point at the mixer's output, IIP_3 at the input, and G is the conversion loss or gain. The OIP_3 for a passive mixer is, therefore, reduced by the mixer's conversion loss. This insertion loss requires compensation in either the RF or IF gain stages in order to establish the receiver's desired overall noise figure. (The noise figure is an additional parameter that must be accounted for in the receiver design.)

Passive vs. active mixers

A major advantage of passive mixers is that they can also be used as frequency upconverters. In other words, their input signals can be converted to a higher frequency. An upconverter is normally employed in a transmitter chain, where it converts an IF signal to the final transmit frequency. Because a passive mixer can be used in the transmit chain as well as the receiver chain, you need to order and stock only one component.

A "direct downconversion receiver" directly downconverts input signals to the baseband without requiring an IF signal. For these receivers, the mixer's data sheet should specify another important parameter called port-to-port isolation. This parameter measures the amount of isolation between the LO's signal and the mixer's input signal. If port-to-port isolation is not large enough, the LO can mix with itself, producing a DC offset at the mixer output that degrades the receiver's performance.

Because a mixer converts frequencies, it generates new frequencies called mixer spurs. Spurs should be investigated thoroughly, especially those at (2RF - 2LO), (3RF - 3LO), and higher orders that affect the receiver by coinciding with its IF frequencies. This behavior is usually described in a mixer's data sheet by the 2 x 2 and 3 x 3 parameters.

Besides these various parameters, you must also consider the level of integration. Some applications can benefit by integrating the mixer core with an LO amplifier, baluns, and LO switch.

A common PCB receiver layout yields design flexibility

Today the development effort can be reduced by using one layout for different frequency ranges. A receiver designed for a 900MHz GSM system can then be used for an 1800MHz GSM system, just by changing a few key components.

A family of pin-compatible mixers is ideally suited for applications in which a common PCB layout accommodates multiple frequency bands for the wireless infrastructure. The ultimate goal is to develop a single layout for a multistandard wireless base station that handles GSM, UMTS, WiMAXTM, and LTE.

A passive mixer like the MAX2029 in the receiver chain, for example, can downconvert the receiver signal, while another identical mixer in the transmitter can upconvert the IF signal to the final transmit frequency. The circuit in **Figure 2** integrates all the external components: LO buffer amplifier, baluns, and LO switch.

Used as a downconverter, the MAX2029 delivers 36.5dBm of IIP₃, 27dBm of IP1dB, 6.5dB of conversion loss, and a



Figure 2. Block diagram of a passive mixer.

6.7dB noise figure. Because the MAX2029's SiGe process technology enables impressive performance, it is well suited for base-station applications in which high linearity and a low noise figure are critical.

The 2RF - 2LO rejection (72dBc with a -10dBm RF input signal) enables simpler and more cost-effective filters by easing the requirements for filtering the close-in harmonics. The MAX2029 expands the frequency range at the lower end from 815MHz to 1000MHz. As one member of a pin-compatible mixer family, which includes the MAX2039 and MAX2041, the MAX2029 allows the creation of a single PCB layout for receivers that handles different frequency ranges and different communication standards.

Active mixers can take the form of either a balanced (Gilbert cell) design, or a passive mixer combined with an IF amplifier stage. The MAX9986, for example, represents the second configuration. Its low noise figure allows less RF gain ahead of the mixer stage, which in turn enables better overall linearity for the receiver. Again, as more gain is added in front of the mixer to minimize the cascaded noise figure, the mixer's linearity must be higher to maintain overall receiver linearity.

A similar article appeared in German in *Elektronik Informationen* in July 2008, and in English in *High Frequency Electronics Magazine* in October 2009. WIMAX is a trademark of the WIMAX Forum.

Choose the right mixer

When searching the Internet for a mixer, the challenge is to sift through all the specifications listed for the various mixers. Then you must make an optimum choice. Fortunately, a web-based parametric search tool will help you do just that. The parametric search enables design engineers to quickly find the right IC for an application. A single page shows all the search criteria for filtering information and all the corresponding parts. Changing any of these criteria updates the parts list immediately. Search features include single-click filtering, sliding filter controls, multilevel sorting, and abundant tool tips. There is no easier way to find the right part for an application.

Figure 3 shows the search results for an active mixer with 10dB gain, designed especially for base stations. The part proposed is a MAX9986. An additional click on that component leads the user directly to the component's QuickView homepage, where the associated data sheet, application notes, and other information can be found.

A parametric search with this web tool from Maxim reveals the number of products that match a specific combination of filter settings—before the user makes the first click. The "smart" search algorithm shows only valid criteria. The user cannot make selections that eliminate all parts. Built using the latest Web 2.0 technologies, this parametric search requires no plug-ins on the user's system



Figure 3. This web tool reveals the number of products matching the filter settings—before a user clicks!

How to use highvoltage and highcurrent-drive op amps in ±20mA or 4–20mA currentloop systems

Maurizio Gavardoni, Sr. Product Definer, Amplifiers and Sensors

Maxim Integrated Products, Inc.

This article shows how to use a high-voltage, high-currentdrive operational amplifier to convert a voltage signal into a ± 20 mA or 4–20mA current signal for use in processcontrol industrial applications. The MAX9943 op amp serves as the example device. Experiments are described and test results presented.

Introduction

Current loops have long been used in process-control industrial applications. Current loops enable the transfer of information from remote sensors to central processing units, or from those central units to remote actuators. Although 4–20mA current loops are quite common for the application, \pm 20mA current loops can be used as well. For low-impedance loads, the use of high-voltage op amps with high-current drive simplifies the circuit design by eliminating the need for external power FETs.

This article shows how to use a high-voltage, highcurrent-drive op amp in a ± 20 mA or 4–20mA current-loop application. The op amp converts a voltage signal from a DAC into a ± 20 mA or 4–20mA current output. Application experiments feature the MAX9943 op amp. Test data are presented.

Basics of a current loop

A current loop typically includes a sensor, transmitter, receiver, and an ADC or microcontroller (**Figure 1**). The sensor measures a physical parameter (such as pressure or temperature) and provides a corresponding output voltage. The transmitter converts the sensor's output into a proportional 4mA-to-20mA current signal. The receiver then converts the 4–20mA current into a voltage. An ADC or a microcontroller digitizes the receiver's voltage output.

In current loops the information is transmitted by a currentmodulated signal. For 4–20mA systems, 4mA normally represents the sensor's zero-value output while 20mA represents the full-scale output. A broken loop (0mA, a fault condition) can be easily distinguished from the sensor's zero output (4mA).

Current loops are inherently more immune to noise than voltage-modulated signals. This is why they are ideal for use in noisy industrial environments. The signal can travel over long distances and information can be sent to, or received from, remote locations. Typically the sensor is remote from the control center where the system microcontroller is located.

A more complex system includes a second current loop from the microcontroller or DSP to an actuator (**Figure 2**). A DAC converts the digital information into an analog voltage signal. A current-loop transmitter converts the DAC's output voltage into a 4–20mA or \pm 20mA current signal that drives the actuator. An example of such a system can be found in power-grid monitoring systems where sophisticated algorithms determine the current state of the system, predict the direction of changes in the system, and implement a control loop to dynamically adjust the system.

Use the op amp as a VI converter with a high-current drive

The circuit of **Figure 3** shows how a simple VI (voltageto-current) converter can be designed with two op amps and a few external resistors. When powered with $\pm 15V$, the



op amp (here, the MAX9943) delivers more than ± 20 mA output current to small impedance loads.

The MAX9943 is a 36V op amp with a high-output current drive. It is stable with up to 1nF load capacitance. The device is ideal for industrial applications where a voltage signal coming from a DAC needs to be converted into a proportional 4–20mA or \pm 20mA current signal.

The relationship between the input voltage, V_{IN} , and the load current is given by Equation 1:

$$V_{IN} = (R2/R1) \times R_{SENSE} \times I_{LOAD} + V_{REF}$$
(Eq. 1)



Figure 2. A more complex system uses a second current loop for controlling an actuator.



Figure 3. A VI converter transforms the DAC output to load current. The circuit uses two MAX9943 op amps.

In this circuit example the component values are:

$$\begin{split} R1 &= 75 k \Omega \\ R2 &= 750 k \Omega \\ R_{SENSE} &= 12.5 \Omega \\ R_{LOAD} &= 600 \Omega \end{split}$$

A typical load value would be in the order of several hundred ohms. However, significantly smaller impedance loads can occur either from short-to-ground faults, or to allow long-distance signal transmission by reducing the voltage burden requirements at the receiver.

 V_{REF} can be synchronized with the reference voltage used by the DAC. In that case all voltages (V_{IN}) are ratiometric with V_{REF} , and errors from variation in V_{REF} can be eliminated.

Create a ±20mA current drive from a ±2.5V range

The circuit in Figure 3 can also be used to create a ± 20 mA current drive. With V_{REF} = 0V, the input range between -2.5V and +2.5V produces a nominal ± 20 mA current output, as shown in **Figure 4**.

The relationship between the input voltage (V_{IN}) and the output voltage (V1) of the "forward" op amp is given by:

$$V_{\text{IN}} = (\text{R2/R1}) \times (1 - \alpha/\beta) \times \text{V1} + V_{\text{REF}} \times (1 - (\text{R2/R1}) \times 1/(\beta \times (\text{R2+R1}))))$$
(Eq. 2)

Where:

$$\alpha = (1/R_{\text{SENSE}}) + R2/(R1 \times (R1 + R2))$$
 (Eq. 3)

$$\beta = (1/R_{SENSE}) + (1/R1) + (1/R_{LOAD})$$
(Eq. 4)

Using the component values specified in Equations 2, 3 and 4:

$$V1 = 4.897 \times V_{IN} - 4.896 \times V_{REF}$$
 (Eq. 5)

The relationship in Equation 5 helps to avoid saturating the output devices. In fact, when $V_{IN} = +2.5V$, the output of the lower op amp (V1) reaches approximately 12.2V. If the input voltage increases beyond 2.5V, eventually the output device reaches its saturation point and the output voltage can no longer increase. The Figure 4 curves flatten and no longer follow the ideal profile. A similar process happens when the negative input is lowered below -2.5V.

The Figure 4 data show that the MAX9943 still operates in the linear range when sourcing and sinking up to approximately ± 21.5 mA, which corresponds to the input of ± 2.68 V and of ± 13 V at the output of the forward (lower) op amp. The negative current could actually be a much larger magnitude because the MAX9943's output voltage can operate very close to the negative supply voltage.



Figure 4. A $\pm 20mA$ output current range is produced by a $\pm 2.5V$ input voltage range. The blue line is the ideal gain curve; the red line is the measured data. $V_{CC} = +15V$; $V_{EE} = -15V$.



Figure 5. A $\pm 24mA$ output current range is produced by a $\pm 3V$ input voltage range. The blue line is the ideal gain curve; the red line is the measured data. $V_{CC} = +18V$; $V_{FE} = -18V$.

The device is limited to approximately 2V from the positive supply. (The 2V value depends on the load, and is given as a worst-case specification vs. process and temperature.)

Some applications require higher output current, either to satisfy margin concerns or to provide room for calibration. For those applications the Figure 3 circuit can be operated with dual $\pm 18V$ supply voltages (instead of $\pm 15V$). Now the op amp can drive up to ± 24 mA (corresponding to $\pm 3V$ inputs) and remain in the linear zone. This performance is illustrated in **Figure 5**.

Create a 4–20mA current drive from a 0 to +2.5V range

Referring back to Equation 5, when $V_{REF} = -0.25V$ the input range between 0 and +2.5V produces a 2mA-to-22mA current output (**Figure 6**). Normally in 4–20mA current loops, designers want extra "room" in the dynamic range (e.g., from 2mA to 22mA) to allow for software calibration. If more current is required, then the MAX9943 circuit can be powered with a dual ±18V supply voltage, as explained earlier.

Conclusion

Current loops are very popular for industrial applications that need to transfer information from remote sensors to central processing units, or from those central units to remote actuators.

The MAX9943 op amp has been shown to be ideal for control-loop applications where a voltage from a sensor or a DAC needs to be converted into either a 4–20mA or \pm 20mA current range. The MAX9943 offers a precision, high-current drive over the temperature range. It is stable with capacitive loads up to 1nF, as are found in long transmission lines.



Figure 6. A 4–20mA output current range is produced by a 0 to 2.5V input voltage range. The blue line is the ideal gain curve; the red line is the measured data. $V_{CC} = +15V$; $V_{EE} = -15V$.