primary ones are shown in the Endnote at the end of this chapter. These formulas can be very difficult to work with and therefore are not particularly useful.

Fortunately, many alternatives have become available for designers to use for calculating the impedance of their traces. UltraCAD has made available a freeware calculator for doing these calculations (Figure 10-8). Refer to Appendix H for information about UltraCAD's freeware calculators. Polar Instruments has a particularly good commercially available calculator available in several versions from their Web site (also shown in Figure 10-8). Many of the higher-end PCB design packages include built-in trace impedance calculators. Numerous other Web-based or Web-obtainable design aids have been developed in the last few years. Do a search on the Web for "PCB trace impedance" or variations on this.

Both propagation time and characteristic impedance can be impacted by capacitive loading along the trace. Every receiving device on the trace adds a small increment of lumped capacitance to the trace. Even vias can contribute small increments of capacitance.

The adjustment factor commonly used is shown in Equation 10-3:

$$\sqrt{1 + \frac{Cd}{Co \times l}} \qquad [10\text{-}3]$$

where Cd is the total amount of all such capacitances added, Co is the characteristic capacitance of the line, per unit length, and l is the length of the trace. The application of this adjustment factor to characteristic impedance and to propagation delay is shown in the Chapter Endnote.

Despite any claims to the contrary, impedance calculations are by their very nature approximate. The calculations are based on approximations that are reasonable, but cannot be exact. That is because, in part, the nature of the physics involved is just too complex to model exactly without any error. Therefore, it is expected that calculations based on tools from different sources might reach slightly different results. The better tools, however, should agree within about 2%.

Your board fabricator, however, will usually be doing well to hit the impedance target within about 5% because of the many variables (times, temperatures, pressures, material dimensions, and even humidity) that exist in the chemically based board fabrication process itself. If you want impedance control tighter than

INTEL-1010
U.S. Patent 7,151,430

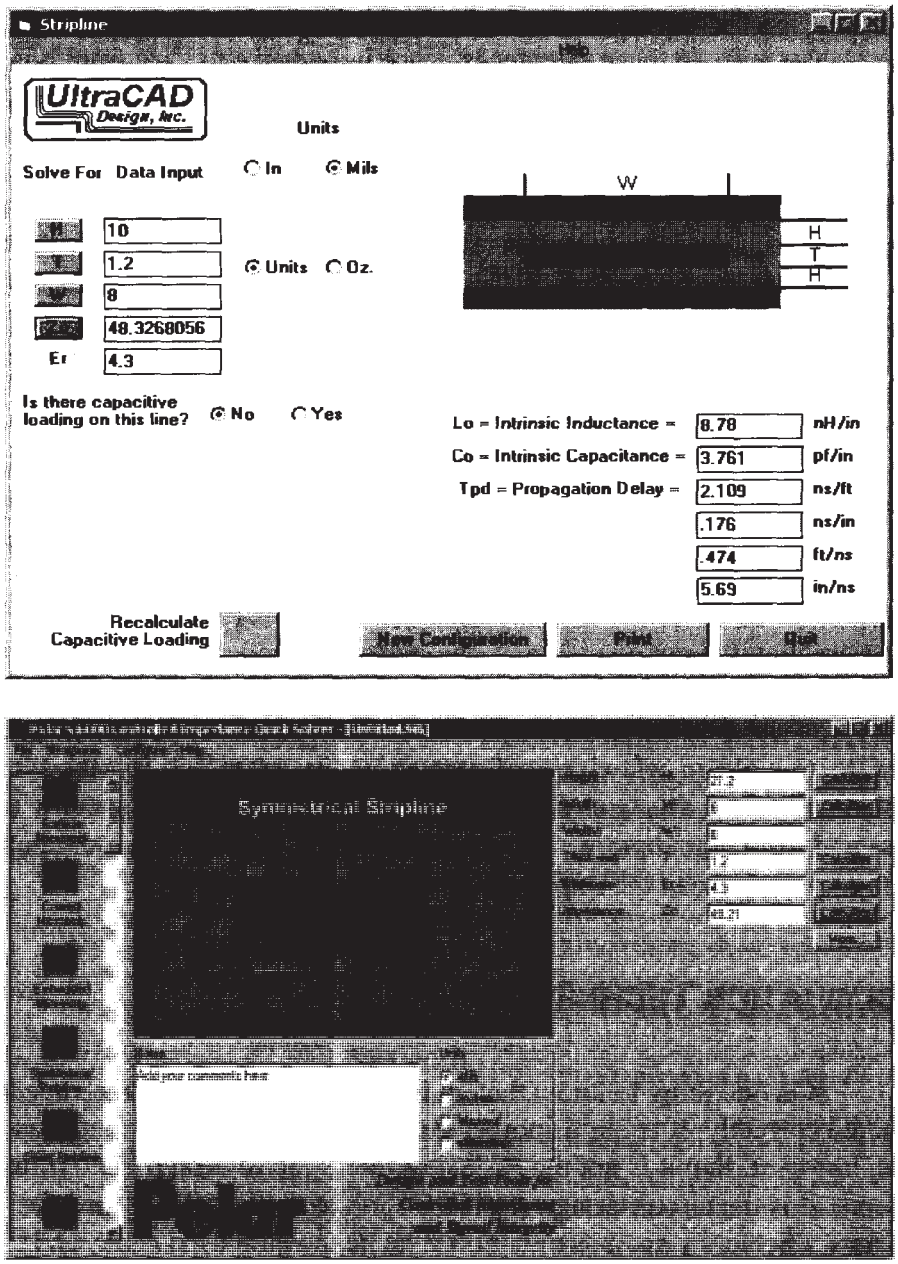



**Figure 10-8** Two impedance calculators: UltraCAD's freeware calculator, UltraCLC, and Polar Instruments SI6000 QuickSolver, both available from their respective Web sites.

about 5%, your selection of board fabricators that can do the job will shrink significantly and the prices will start going up.

Not only do these tolerances cause impedance to vary from board to board, they will cause impedance variations from place to place on the same board. In 1994 UltraCAD partnered in a study to determine the effects of vias on traces. An unexpected by-product of that study was information regarding the variation of impedance between boards fabricated in the same batch process and even along traces on the same board. Figure 10-9 is taken from that study. It shows a TDR pattern of one of the traces on a test board. The trace is 15 inches long and has three vias along its length, but the trace does not pass through the via. Instead, the trace stays on the same layer for the entire 15 inches. The impedance varies by about 3% along this trace.[2]

There are several reasons for this kind of result along a trace: temperature gradients during fab, pressure gradients during fab, nonuniform thicknesses and etching, material nonlinearities, and so on. Polar Instruments has done some interesting work to show that another effect can also come into play. One of the most common board materials used in board fabrication is FR4. FR4 is a blend of resin and glass, but it is not a uniform, homogeneous blend. The glass tends to form in clumps and the resin fills in around it (see Figure 10-10). The glass has an effective $\varepsilon_r$ of approximately 6.0 and the resin an $\varepsilon_r$ of approximately 3.0. These combine, on average, to yield an effective $\varepsilon_r$ of approximately 4.0 to 4.3. But the $\varepsilon_r$ can vary dramatically from this at any specific point, depending on the specific mixture of glass and resin at that point. Since the effective $\varepsilon_r$ can change so dramatically from point to point, so can the impedance that depends on that $\varepsilon_r$.

## TERMINATION TECHNIQUES

It is one thing to know that an impedance-controlled trace needs to be terminated correctly. It is quite another thing to know how to do that. It is important to note, however, that it is not the PCB designer's job to design the termination scheme. It is the circuit design engineer's job to do that. The board designer's job is to know whether or not a termination scheme is being used at all, and to raise the question if it appears that a termination scheme has been inadvertently overlooked.

[2]See "The Effects of Vias on PCB Traces," 1994, available from UltraCAD's Web site. This article also appeared in *Printed Circuit Design* magazine in August 1996. Also see Appendix I for information on how TDRs measure the impedance of traces.

INTEL-1010
U.S. Patent 7,151,430

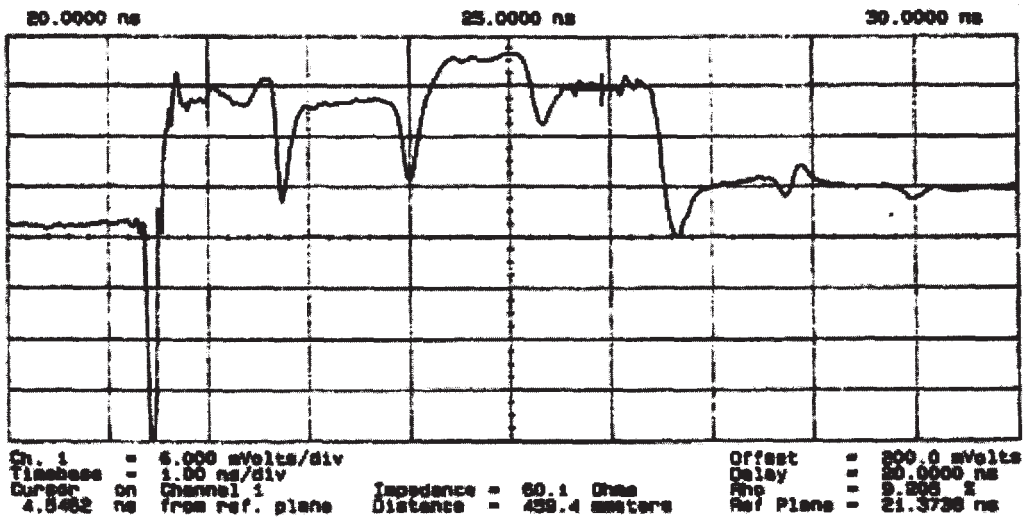



**Figure 10-9** TDR results of a PCB trace illustrating impedance variations at different points along the trace (courtesy UltraCAD Design, Inc.).



**Figure 10-10** FR4 is made up of glass and resin. The apparent combination of these can vary from point to point on the board. This is a copy of a microsection cut from a test board (courtesy Polar Instruments).

I can tell you from personal experience that everyone wins when you raise the question about terminations. If no change is necessary, you look intelligent simply for knowing about transmission lines, impedances, and terminations. If a change

is necessary, you become a hero for being the one to call it to your engineer's or customer's attention. Either way, you gain in stature.

In this chapter we discuss the five most common forms of transmission line terminations.[3]

*Parallel* Parallel termination (Figure 10-11) is the most intuitive to understand. A single resistor equal to the characteristic impedance of the line (RL = Zo) is attached to the end of the line. All the energy flowing down the trace is absorbed in the resistor and there is no reflection. The terminating resistor can be connected to either Vcc or ground.

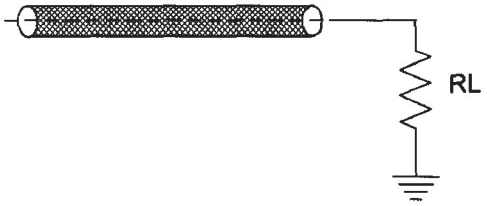



**Figure 10-11** Parallel termination.

This technique has several advantages: (a) the value of the resistor is relatively easy to determine, (b) there is only a single component, (c) it is easily connected, and (d) it performs well with distributed loads (i.e., loads that are distributed along the trace). There is only one drawback to this type of termination: It provides a continuous DC path to ground. Therefore, continuous DC current can flow through it at all times. This may or may not be an issue for a single trace. But if your design has 1,000 or so impedance-controlled nets, the total power dissipation in 1,000 terminating resistors can become quite significant.

*Thevenin* A closely related variation of the parallel termination strategy is the Thevenin termination (Figure 10-12). This consists of a pair of resistors, one going to ground and one going to Vcc. The pair of resistors provides a pull-up/pull-down function as well as a termination function. Therefore, it can improve noise margins in certain situations and performs as well as the parallel termination with distributed loads.

[3]For a good discussion of trace terminating techniques see Ethirajan and Nemec, "Termination Techniques for High Speed Buses," EDN, February 16, 1998, p. 135.

INTEL-1010
U.S. Patent 7,151,430

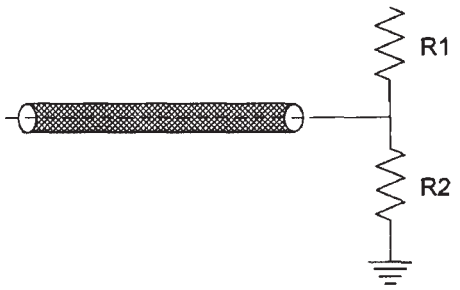



**Figure 10-12** Thevenin termination.

On the one hand, it would appear that the selection of resistor values would be relatively straightforward. The parallel combination of the two resistors must simply equal Zo, or (c.f. Equation 5-9):

$$Zo = \frac{1}{\frac{1}{R1} + \frac{1}{R2}}$$

But Ethirajan and Nemec showed that there are optimum values for R1 and R2 based on the specific characteristics of the driver. The optimum values for R1 and R2 primarily optimize (minimize) the power dissipated in the circuit. This optimum value can be difficult to determine. Other drawbacks to this strategy include the addition of an additional component, and the fact that DC current still flows through the resistor pair at all times. Finally, this strategy is only well-suited for bipolar (two-state) devices, not for tristate logic families.

Some people have reported that there can be an EMI problem with this termination strategy. Note that there are different currents flowing between Vcc and ground, through R1 and R2, depending on the logic state. This current changes at the same rate (di/dt) as the logic state changes (the rise/fall time of the signal). In this respect, the two resistors look exactly like a switching logic gate. Therefore, they might need to be decoupled (with bypass capacitors) just as a logic gate might need to be decoupled (with bypass capacitors). The issue really is related to loop areas and currents, which are one of the primary causes of EMI. Thus, you may occasionally hear people say that Thevenin terminations may also require decoupling capacitors to quiet down EMI emissions.

INTEL-1010
U.S. Patent 7,151,430

*AC Termination* Yet another variation is the addition of a capacitor in series with the parallel terminating resistor (Figure 10-13). The primary advantage of this is that the capacitor blocks DC current, so there is no steady-state current flowing through the termination. At first glance it would appear that this strategy otherwise has all the advantages of the parallel termination strategy. However, the "cost" of this, of course, is the added component.

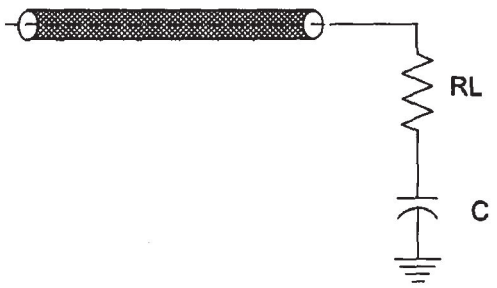



**Figure 10-13** AC termination.

There are some other subtle and some not-so-subtle problems with AC termination. If a large capacitor value is used, there can be considerable power dissipation, and the strategy is little different from normal parallel termination. If a very small capacitor is used, it will cause overshoot and may interfere with the rise and fall times of the signal.

There are also some subtle interactions that take place with AC termination using a small capacitor. As the voltage across the terminating resistor changes, the current through the capacitor changes. This causes the capacitor to charge or discharge with an RC time constant related to the component values. If the time constant is too short, the capacitor charges during the half-cycle and the voltage at the receiver changes (and distorts) accordingly. Very long lines can also lead to signal distortion.[4]

*Series* Series termination (Figure 10-14) is becoming more and more common in today's high-speed designs. It has the two desirable attributes of a single component and no DC current draw at all. The series termination resistor is placed at the front of the trace, however, not the end; the end is left open-circuited. There-

[4]Brooks, Douglas, "Transmission Line Terminations, It's the End That Counts!" available from Mentor's Web site at www.mentor.com/pcb/tech_papers.cfm.

INTEL-1010
U.S. Patent 7,151,430

fore, there is a 100% positive reflection from the far end of the trace that reflects back toward the front.




**Figure 10-14** Series termination.

The fact that there is a single reflection traveling back up the line is of little consequence if there are no devices along the line; that is, if the loads are lumped at the far end of the line. But if the loads are distributed along the line, the loads will see different signal (voltage) levels depending on which way the signal is traveling. Some designs actually take advantage of this and are designed to switch only at the higher signal level. The computer PCI bus is one example of this.

The value for the series termination resistor is set so that the sum of it and the output impedance of the driver totals to the impedance of the trace. That way, there is no secondary reflection from the driver. The voltage pattern we expect to see is a single, positive reflection from the far end of the trace that travels to the front of the trace and is completely absorbed in the source resistor and driver.

That means that we have to know the output impedance of the driver. That isn't too serious a problem, since the output impedance should be specified by the manufacturer. But an additional complication is that the output impedance is often *different* depending on the output state. That is, the output impedance of the logical high state may be different from the output impedance of the logical low state, making it impossible to properly terminate both states with a single series resistor.

*Diode* The diode termination technique is not designed to minimize reflections, only to limit them. Two diodes are place on the line as shown in Figure 10-15. Reflections can exist at both ends of the line and will travel back and forth along the line. This technique simply limits the magnitude of the reflection to between Vcc and ground (or more properly, one diode drop above or below these levels, respectively). There is no power impact of this technique, and the diodes can be placed anywhere along the line. In high-speed designs, the diodes must be very fast switching devices.

INTEL-1010
U.S. Patent 7,151,430

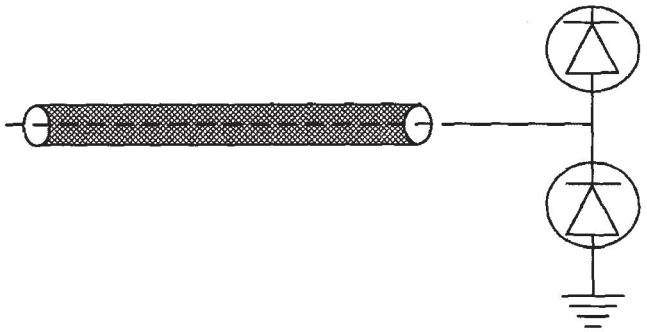


**Figure 10-15** Diode termination.

## SOME DESIGN ISSUES

In this section we discuss a few design issues related to controlled impedance traces that can be very important.

*Changing Trace Layers 1* We have indicated that for there to be impedance control, and therefore control over reflections, the impedance of the trace must be constant over its entire length. Since impedance is a function of geometry, this implies the geometry must be constant, also. Consider, however, what happens when a trace changes layer (Figure 10-16). It is important that the *impedance* of the trace stays constant. But the geometry has changed, the relationship to the planes has changed, and the (relative dielectric constant of the) material may have changed. Therefore, it is highly likely that if we keep the same trace width, the impedance will change.

Since the environment often changes when we change layers, we usually have to also adjust the trace width when we change layers. This is necessary because what is important is maintaining a constant impedance (not just a constant geometry) on each layer. If we are designing with a certain impedance in mind (say 50 Ω) we have to determine what a 50-Ω trace looks like on *each layer* of our board. Then, if we move a 50-Ω trace to that layer, we may have to change the trace's properties to maintain the desired impedance on that particular layer.

INTEL-1010
U.S. Patent 7,151,430

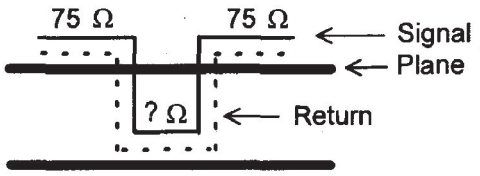



**Figure 10-16** When changing layers, care must be taken to ensure that the impedance of the trace is constant on all layers, even if that means the geometry needs to change.

*Power Planes* The location of the return signal is an integral part of the geometry of impedance determination and control. This point simply cannot be overemphasized. For coaxial and twin lead cables, the location of the return current is fixed by the design of the cable. On PCBs, the return signal wants to return directly underneath the trace (refer to Appendix B). This simple concept has some significant implications.

*All* signals have a return signal associated with them. Every single one. Y*ou* (as the PCB designer) need to know where they are. It is ironic that some designers spend an enormous amount of time worrying about, strategizing about, and routing 50% of the signals. They then ignore the other 50% of the signals (the return signals) assuming they will take care of themselves.

It is (almost) impossible to achieve impedance control on a PCB without a reference plane for the return signal to travel on. Thus, unless you are really good, and have some really well-thought-out strategies, all high-speed designs need power and ground planes. It is (almost) impossible to achieve impedance control without them.

The return signals can, and do, travel on either the power or the ground plane. Power and ground planes are the same thing to AC signals. The distinction is only a DC distinction. (If you have trouble visualizing this, consider how many capacitors—which pass AC currents—there are between the power and ground planes.) Typically, the return signal will be on the closest plane to the signal trace, which we then call the *reference* plane.

*Changing Trace Layers 2* If we do anything to prevent the return signal from following directly underneath the trace, we impact the geometry. This, then, impacts the magnitude of the impedance, causes an impedance discontinuity, and

INTEL-1010
U.S. Patent 7,151,430

therefore causes a possible reflection. Therefore, we must be careful in our designs to never do anything that restricts the return signal.

Figure 10-17 illustrates three typical routing situations. (Note that this is the same figure as Figure 9-10 in Chapter 9. In Chapter 9 we talked about this issue from the standpoint of loop areas and EMI.) In Figure 10-17a, a trace is routed above a reference plane and the return signal (dotted line) returns on the plane directly under the trace. In Figure 10-17b, the trace routes through a via and proceeds on the trace layer on the other side of the reference plane. The return signal crosses over to the other side of the plane, and remains directly under the trace. In neither case are we interrupting the return signal path. There is no impedance discontinuity in either of these situations (assuming we have constructed the trace itself correctly on each layer).

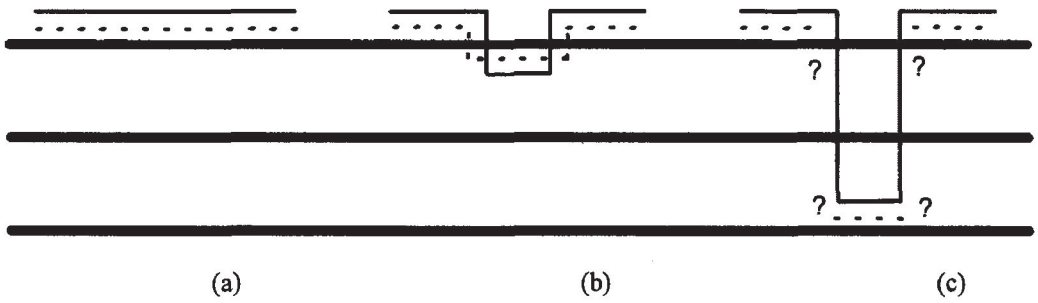



**Figure 10-17** It is important to know where the return currents are to maintain impedance control.

What about Figure 10-17c? Here the trace routes through a via to a completely different trace layer. To maintain impedance continuity, the return signal must find its way to a new reference plane several layers away. A good question is this: How does it do that? Unfortunately there is not a really satisfying answer.

We know this situation works because lots of people (including us) do this. The practical answer is that the return signal travels to the other plane through a nearby bypass capacitor (since there are lots of them distributed around our board). But it is possible that the path to the bypass capacitor and back under the trace creates an impedance discontinuity that may then create an unwanted reflection.

Different engineers have different attitudes toward this:

INTEL-1010
U.S. Patent 7,151,430

1. Some ignore the problem as being relatively minor.
2. Some (e.g., Henry Ott and Howard Johnson in some e-mail forums) have suggested we might consider placing bypass capacitors beside each via for the specific purpose of providing a path for the return signal (for both impedance control reasons and EMI loop area reasons).
3. A few take the position that no traces at all shall be routed in this manner to ensure that this type of reflection problem cannot happen.

Most people consider the third option to be extreme. But if your engineer takes this position, at least you now understand what his or her thinking is. After all, the engineer is the customer and we do what the engineer asks.

Note: If you want to know what UltraCAD does, we follow the first option, and so far have not experienced a problem. If you fell into the trap of considering a "hard" or solid via between the two planes, consider the consequences if one plane is ground and the other is Vcc. You can end up with a lot of angry electrons this way, at least for a moment.

*Slots in Planes 2* As we have emphasized, the geometry associated with the traces must be controlled. This means, in particular, that the return signal must be allowed to follow directly on the plane underneath the trace. Consider a situation illustrated in Figure 10-18. (Note that this is the same figure as Figure 9-7. In Chapter 9 we said this could cause an EMI problem. You will see this figure yet again when we talk about crosstalk. There are several reasons not to allow slots on planes.) A slot exists, for some reason, on the plane. A controlled-impedance trace routes across the slot. The return signal must loop up around the slot and back again to the trace. The geometry of the trace has changed at this point, causing an impedance discontinuity. This is analogous to cutting a coaxial cable and then resplicing it as suggested in the figure. If you have ever done this with your TV cable, you ended up with a severe ghosting problem on the TV caused by the reflections, which were in turn caused by the impedance discontinuity at the splice.

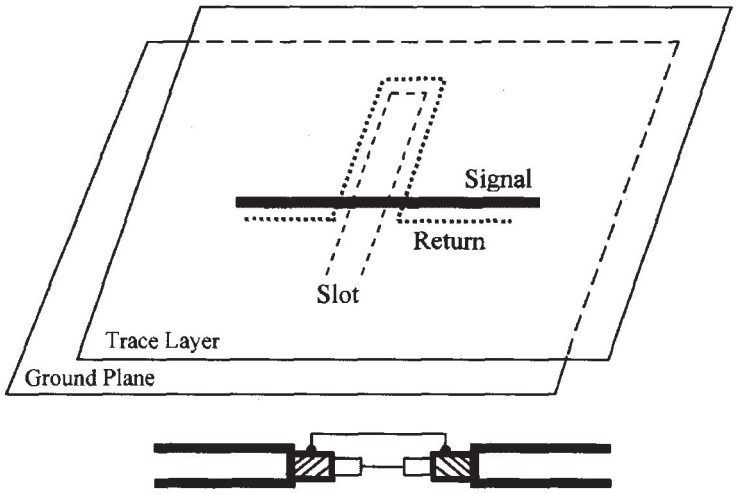



**Figure 10-18** A discontinuity on the plane can cause the return signal to loop, causing, in turn, an impedance discontinuity.

## STUBS

In the last chapter we talked about stubs from an EMI standpoint. There is also an impedance aspect to stubs that is important. Figure 10-19 (the same as Figure 9-12 from the previous chapter) illustrates how stubs can exist when we make a connection from a trace to a pad or device. The stubs are unterminated sections radiating away from a controlled impedance trace, almost like the "Y" problem we will talk about in the next chapter. Signals can reflect from the far (unterminated) end of the stub and cause signal integrity problems.

Stubs S1 and S3 are short enough (well within the critical length) so as not to cause much of a problem. Stub S2, however, will result in a significant impedance discontinuity and damaging reflections, and such long stubs are to be avoided. Obviously, Stub S4 will cause a reflection problem.

INTEL-1010
U.S. Patent 7,151,430

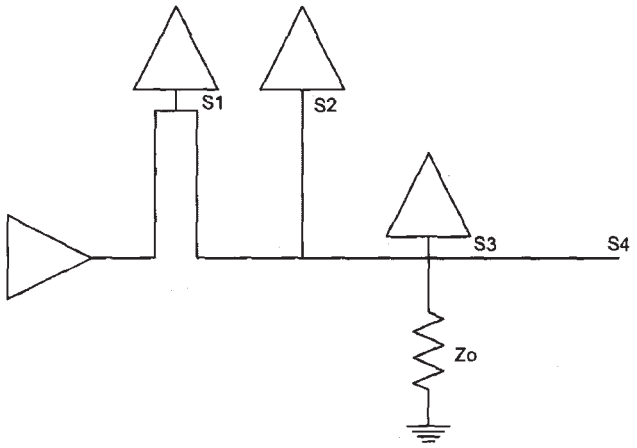



**Figure 10-19** Stubs are unterminated transmission line segments and must be kept short to avoid reflections.

The rule for stub length most people use is about 1/10 or 1/12 the rise time. For a 1 ns rise time (about 6 inches) the rule would be about 75 to 100 ps, or about .5 inche. I have run simulations with signal integrity models and found that reflections are just becoming detectable in this range. Nevertheless, good design practice would be to keep stub lengths shorter than this. Lengths less than .25 inches are usually easily achievable.

## ABSOLUTE VS. RELATIVE VALUE OF ZO

A common question board designers ask is "What should the value of Zo be? What value should I use?" With a few exceptions, the answer is that it doesn't matter much. The value of Zo is not nearly as important as is the fact that (whatever it is) (a) it is constant everywhere along the trace, and (b) that it is terminated properly with that value (whatever it is).

The absolute value of Zo is important under these conditions (there may be other special cases that should be obvious when they come up):

1. If the trace is mating to the outside world, usually through an impedance-controlled connector, then the trace must match the connector.

2. If the trace is mating with a trace on another board, again probably through a connector, a specified impedance is probably necessary because all boards must "play" together without special matching.
3. A very few special components have strict impedance-matching requirements.
4. The circuit designer may specify an impedance level because it is easier to select or specify termination resistors or circuits for the board. (Note that this is not a technical requirement, but is a policy requirement.)

Otherwise, devices are reasonably tolerant of the impedance of the trace, and a specific value of trace impedance is not particularly important.

We do a lot of boards in the video processing industry. The signals frequently come onto and leave the board through 75-Ω connectors. Obviously the traces leading to and from these connectors must also be 75-Ω traces. But the traces on the inner circuits don't need to be 75 Ω. Nevertheless, some of our customers require 75-Ω traces everywhere on the board, because "these are video signals." It's as if there is something special about a video waveform that only works on a 75-Ω trace.

There is not uniform agreement among engineers regarding whether we should use relatively higher trace impedances (say 80 Ω or more) or relatively lower trace impedances (say 30 Ω or so). If you subscribe to various e-mail forums you will hear different theories about either of these extremes, as well as why we should all stick with 50 Ω.

One argument says that higher impedance results in lower currents and therefore lower EMI problems. Others argue that there are offsetting voltage issues with higher impedances that negate any EMI gains. Another argument is that higher impedance creates more voltage-related noise problems on the board. Still others argue that lower impedances cause increased current and therefore power supply requirements. Still others try to offer proof that 50 Ω offers the best overall compromise. I have not heard any convincing argument that there is any impedance level that is inherently better than any other within the reasonable range of about 30 Ω to about 100 Ω.

INTEL-1010
U.S. Patent 7,151,430

# Chapter Endnote

## On Formulas

These are the generally accepted formulas for impedance-related calculations and can be found, except where noted, in several publications.[5]

In the following formulas:

Dimensions are in mils

$\varepsilon_r$ = Relative dielectric coefficient

$\varepsilon$ = base of the natural logarithm ($\cong 2.71828$)

Ln = natural logarithm.

**Microstrip:**

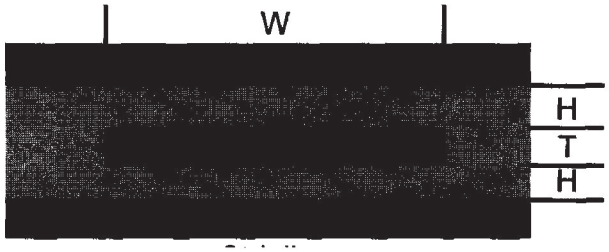



$$Zo = \frac{87}{\sqrt{\varepsilon_r + 1.41}} \text{Ln}\left(\frac{5.98H}{.8W + T}\right)$$

[5]IPC-D-317A, "Design Guidelines for Electronic Packaging Utilizing High-Speed Techniques," January 1995, IPC p. 22ff.

$$Co = \frac{.67(\varepsilon_r + 1.41)}{Ln\left[5.98H/(.8W + T)\right]}$$

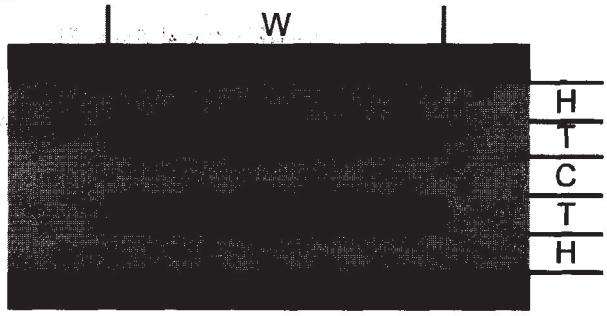
**Stripline:**




$$Zo = \frac{60}{\sqrt{\varepsilon_r}} Ln\left(\frac{1.9(2H + T)}{.8W + T}\right)$$

$$Co = \frac{1.41\varepsilon_r}{Ln\left[3.81H/(.8W + T)\right]}$$

**Dual Stripline:**




$$Zo = \frac{80}{\sqrt{\varepsilon_r}} Ln\left[\frac{1.9(2H + T)}{.8W + T}\right]\left[1 - \frac{H}{4(H + C + T)}\right]$$

$$Co = \frac{2.82\varepsilon_r}{Ln\left[2(H - T)/(.268W + .335T)\right]}$$

**Embedded Microstrip:**[6]




$$Zo = \frac{60}{\sqrt{\varepsilon_r\left[1 - \varepsilon^{\left(-1.55H_1/H\right)}\right]}} Ln\left(\frac{5.98H}{.8W + T}\right)$$

**Propagation Time (nanoseconds/inch):**

Stripline:

$$PropagationTime(T_{pd}) = \sqrt{\varepsilon_r}/11.8$$

Microstrip:[7]

$$PropagationTime\left(T_{pd)}\right) = Br \times \sqrt{\varepsilon_r}/11.8$$

Where:

$Br = .8566 + .0294Ln(W) - .00239H - .0101\varepsilon_r$
W = trace width (mils)
H = distance between the trace and the plane (mils)
$\varepsilon_r$ = relative dielectric material between the trace and the underlying plane
Ln represents the natural logarithm, base e

[6]Brooks, Douglas, "Embedded Microstrip Impedance Formula," *Printed Circuit Design* magazine, February 2000.
[7]See Chapter 2.

**Corrections for capacitive loading:**

The commonly used correction factor for capacitive loading of a trace is

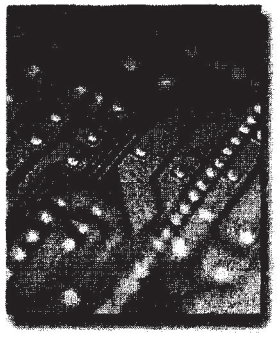
$$\sqrt{1 + \frac{C_d}{Co \times l}}$$

where Cd = capacitance added by the device loads
Co = intrinsic capacitance of the line per unit length
l = length of the line

Its proper usage for this correction factor for impedance (Zo) and propagation times (Tpd) is:

$$Zo' = \frac{Zo}{\sqrt{1 + \frac{C_d}{Co \times l}}}$$

$$T_{pd}' = T_{pd}\sqrt{1 + \frac{C_d}{Co \times l}}$$



# 11 SOME TRANSMISSION LINE SIMULATIONS

*M*entor Graphic's Signal Vision simulator is a useful tool for illustrating transmission line issues. In this chapter we look at several simulations of both good and bad designs to see what the differences are between them.

## BASIC SIMULATION

Figure 11-1 illustrates a basic simulation setup. The individual elements of the simulation are selected and placed on the work surface and connected together. The elements are then given values for analysis.

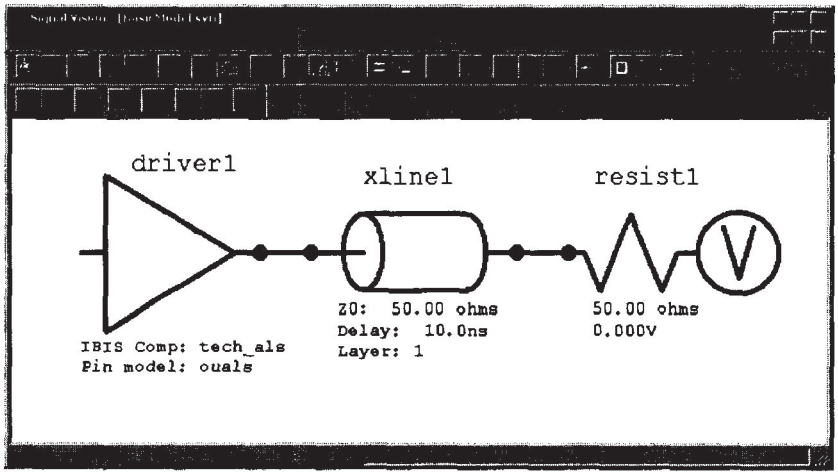



**Figure 11-1** Basic simulator setup.



INTEL-1010
U.S. Patent 7,151,430

Figure 11-2 shows the various screens for defining the values. It is almost always necessary with any manufacturer's simulation tool to define a software model for any active device used, in this case the driver. These software models are usually (but not always) IBIS models.

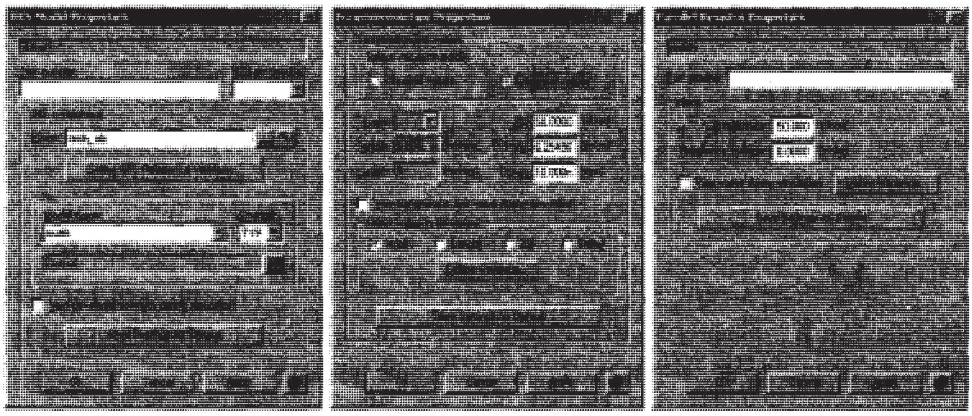


**Figure 11-2** Submenus for setting the element parameters for the simulation.

Double-clicking the driver element produces the first submenu shown in Figure 11-2. We have selected a generic IBIS model provided by Mentor Graphics, designated as "tech_als." We select this model because it is convenient and because it has a fast rise time (approximately 1.0 ns).

Double-clicking the transmission line element brings up the second submenu. We can define the transmission line in terms of its board stackup parameters or by its electrical parameters. The board stackup parameters have obvious value if we are investigating real board parameters or if we are integrating the Signal Vision tool with the Expedition board design tool and analyzing an actual design in progress. In this instance we select electrical parameters to define a 50-Ω line 10 ns long.

Finally, we double-click the load resistor and select the loading for the line, shown in the third submenu. Here we have selected 50 Ω for our initial simulation, tied to zero volts (ground).

We could add other elements to the simulation and change any of the parameters through these various screens.

INTEL-1010
U.S. Patent 7,151,430

Finally, we set up the simulator control, Figure 11-3. Here we are telling the simulator to cover a time period of 20 ns with a pulse that is starting low and rises, and stays "high" for 19 ns. That is, we a simulating a step function rise in signal and looking at it for 19 ns.

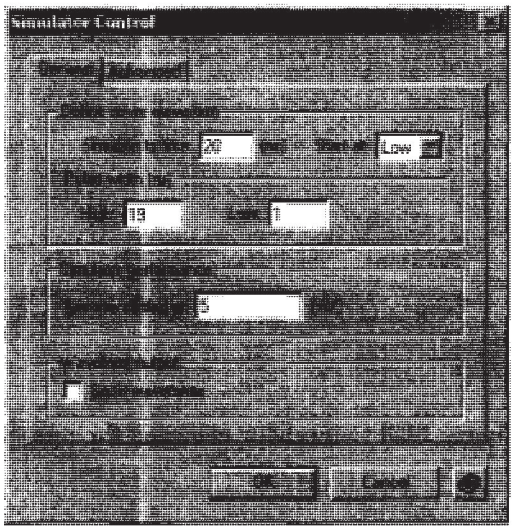


**Figure 11-3** Simulator control submenu.

The result is shown in Figure 11-4 and looks just like we would expect for a properly terminated transmission line. The driver signal immediately rises to 2.5 volts and 10 ns later the voltage at the load resistor rises to 2.5 volts. This time delay is the propagation time down the 10 ns transmission line. Since the transmission line is terminated correctly, there are no reflections.

Now let's do something more interesting. We'll terminate the line with a 150-Ω resistor and see what the results are. To see the results clearly, we'll need to extend the simulation time much further out, in this case to 100 ns. The result of the simulation is shown in Figure 11-5.

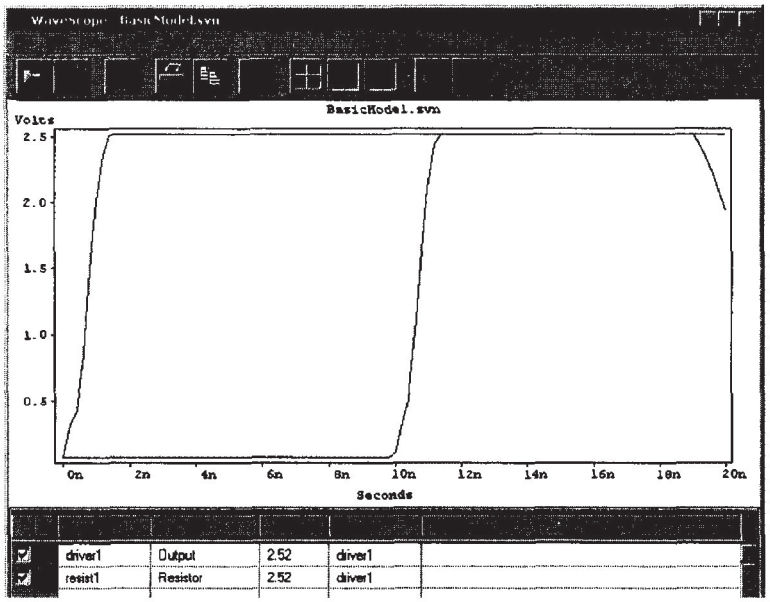



**Figure 11-4** Simulator response for a properly terminated line.

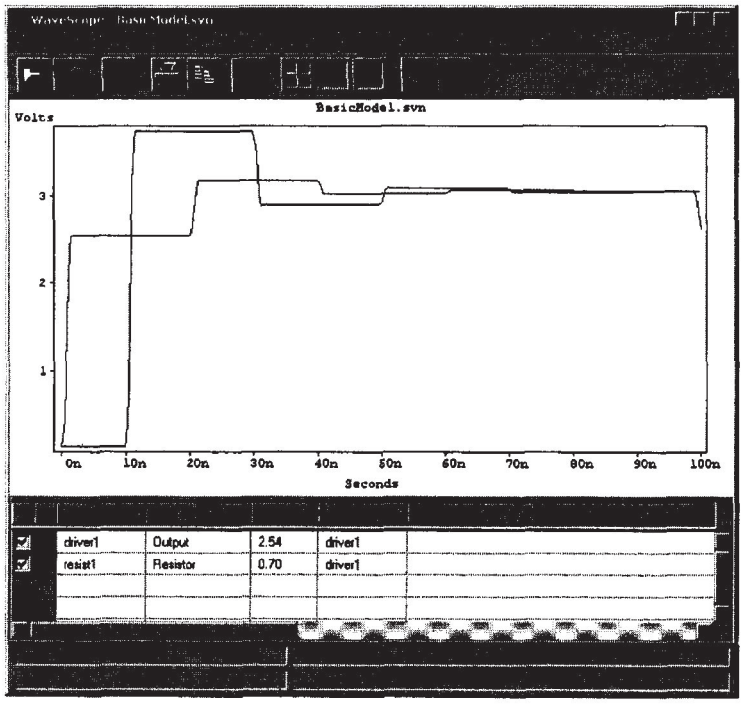
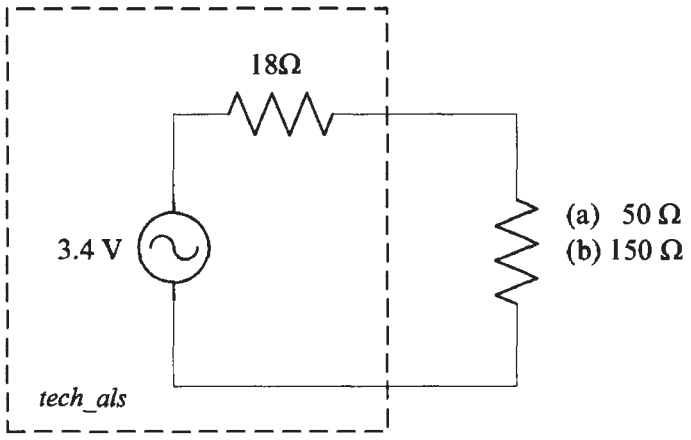



**Figure 11-5** Simulator response for a 150-Ω termination.

To fully appreciate what is happening here, we need to look at the equivalent circuit of the simulation, including the Thevenin equivalent circuit of the driver in its "high" state (see Chapter 5). The tech_als software driver model used here has an output impedance of about 18 Ω in the "high" state (Figure 11-6). The trick is, the equivalent circuit for the simulation is different for the very first instance than it is in the steady state. In the very first instance (fast rise time), the transmission line looks (to the driver) like a 50-Ω load (a). The driver has not yet seen the load (whatever value it is) at the far end of the trace. In the steady (low-frequency) state the driver sees *only* the load at the end of the line (b) and no longer the transmission line. Figure 11-5 is a representation of what happens as we transition between these two times (initial response and long-term stabilization).

Figure 11-6 shows the equivalent circuit for this simulation. At the very first moment, the driver sees the 50-Ω load of the transmission line. The 50-Ω transmission line and the 18-Ω source impedance of the driver combine to form a voltage divider (see Chapter 5). The voltage at the driver output immediately rises to (50/68)3.4 = 2.5 volts. This 2.5-volt signal propagates down the transmission line to the load and arrives at the load 10 ns later.




**Figurer 11-6** Thevenin equivalent circuit for the driver. The transient load (a) the driver sees is different from the steady state load (b).

Because the line is not terminated correctly, the signal reflects off the load with a positive reflection coefficient (see previous chapter) of

$$\rho = (RL - Zo)/(RL + Zo) = (150 - 50)/(150 + 50) = .50$$

A reflection of .5 x 2.5 = 1.25 volts occurs, so the reflected signal traveling back up the line toward the driver is the initial 2.5 volts plus the 1.25-volt reflection, or 3.75 volts. Note that this is higher than the maximum output available from the driver itself (3.4 volts).

Ten ns later (at 20 ns) this reflected signal arrives back at the driver. But now there is a negative reflection because the 50-Ω transmission line sees a source impedance (at the driver) of only 18 Ω. At the source end the reflection coefficient is

$$\rho = (RL - Zo)/(RL + Zo) = (18 - 50)/(18 + 50) = -.47$$

The 1.25-volt reflected signal is reflected again, this time with a magnitude of –.47 x 1.25 = –.59 volts. The voltage at the driver is now 3.75 minus this reflected .59 volts or 3.20 volts. The reflected signal continues back and forth in this manner until it dies out and stabilizes at its long-term (DC) value.

The long-term DC value is found from Figure 11-6 using the long-term load value of 150 Ω. The long-term stable value is (150/168)3.4 = 3.0 volts, which can also be seen from simulation, Figure 11-5.[1]

Now let's look at the same model with only a 20-Ω load resistor. The model result is shown in Figure 11-7. As before, the voltage at the driver output immediately rises to (50/68)3.4 = 2.5 volts. When that 2.5-volt signal arrives at the far end of the transmission line, it reflects with a negative reflection coefficient equal to

$$\rho = (RL - Zo)/(RL + Zo) = (20 - 50)/(20 + 50) = -.43$$

The magnitude of the reflected signal is –.43 x 2.5 = –1.07 volts. So the immediate signal at the load is 2.5 – 1.07 = 1.43 volts. This signal travels back toward the driver, where the reflection coefficient is –.47, as before. So the magnitude of the next reflection is .47 x 1.07 = .50 volts. Therefore, the signal level at the driver output just after 20 ns is 2.5 – 1.07 – (–.5) = 1.93 volts.

[1]This kind of problem is solved manually through a lattice diagram. There are numerous sources, including the Web, for information on using lattice diagrams to solve reflection problems.

INTEL-1010
U.S. Patent 7,151,430

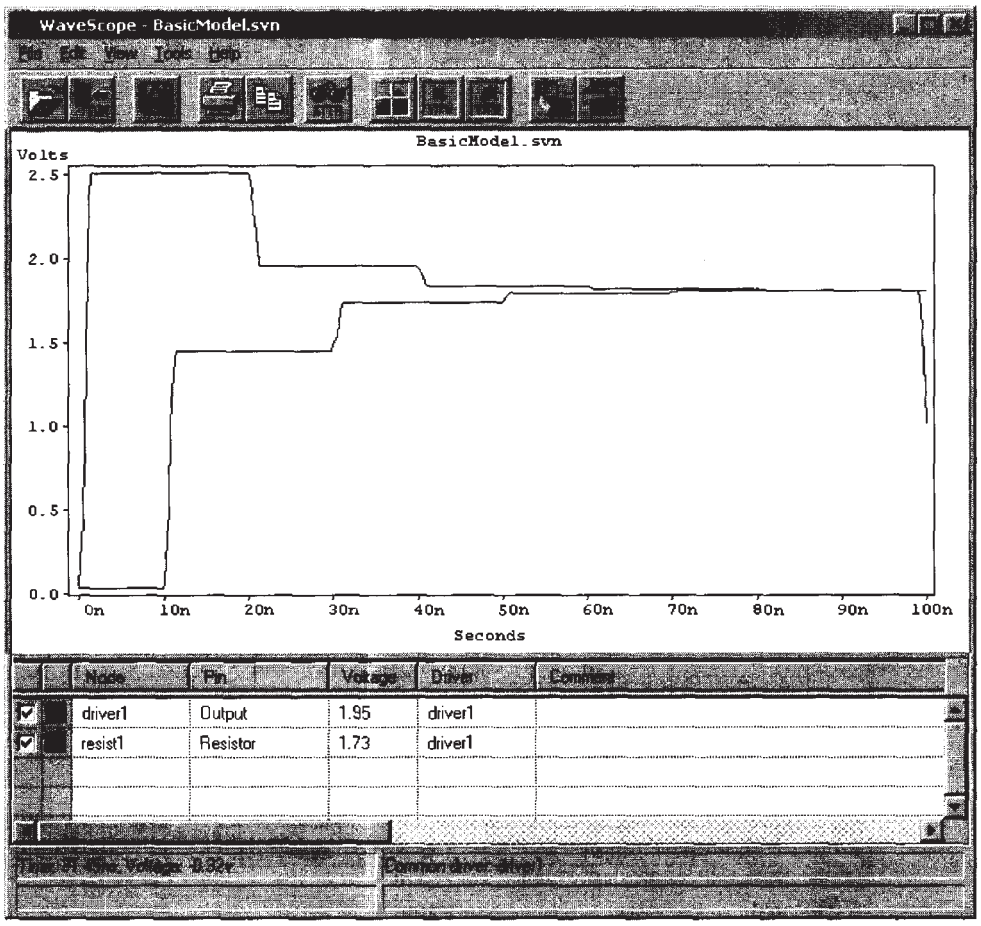



**Figure 11-7** Simulator response for a 20-Ω load.

As before, the reflections continue until they die out. The system will stabilize at a DC value. The 20-Ω load combined with the 18-Ω source impedance yields a (20/38)3.4 = 1.79 final voltage.

As one final basic illustration, let's select a perhaps more realistic simulation. We'll set the trace length at 2 ns (equal to about 6 in) and simulate a square pulse width of 12 ns high and then 12 ns low. If we perfectly terminated the line, we would expect a perfect square wave at the load end on the line. Each waveform would be stable at either 0 volts or 2.5 volts, and the load waveform would follow the driver waveform by 2 ns. Figure 11-8 illustrates the case of a poorly terminated

line (150 Ω). As can be seen from the figure, the waveforms are not very clean and could very well cause system logic errors because of the incorrect terminations.

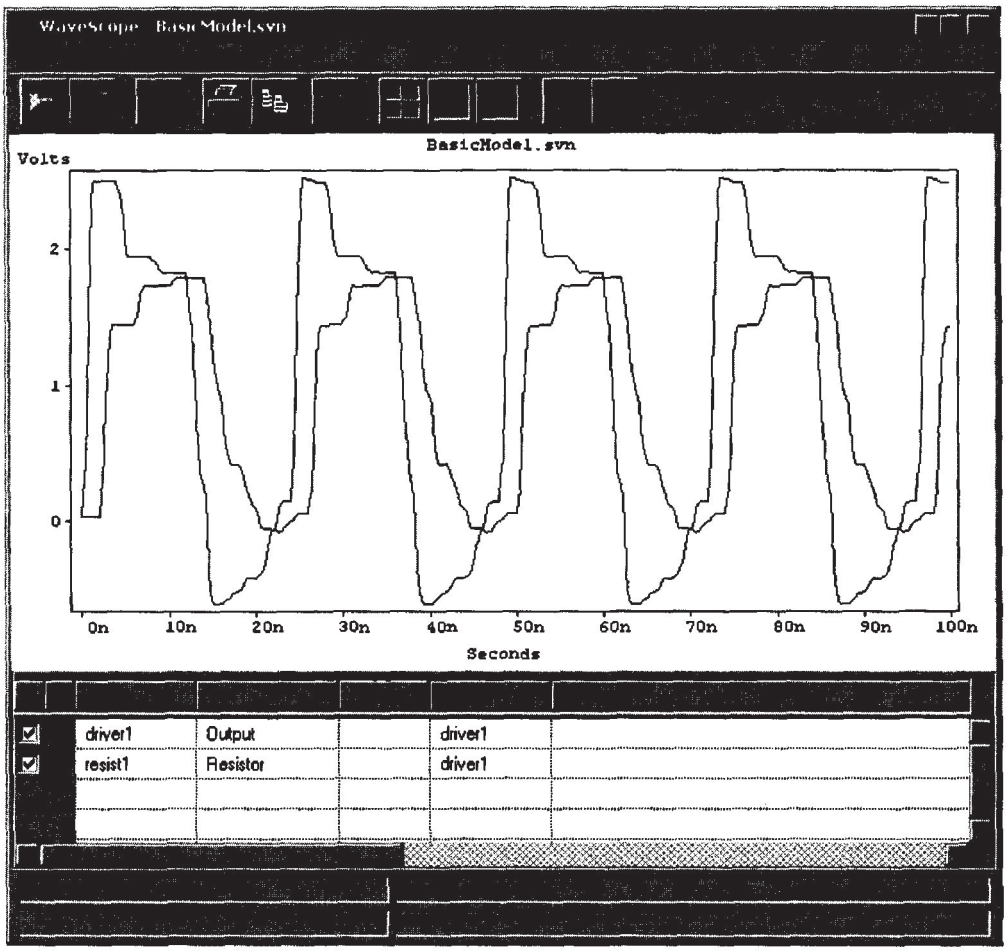



**Figure 11-8** Simulator response of a poorly terminated (150-Ω) line to a repetitive square wave input signal.

## SERIES TERMINATION

As described in the previous chapter, series termination allows for a full reflection at the far end of the line. The reflection is absorbed at the source end, however, and no further reflections exist. Figure 11-9 is an equivalent circuit for our simulation of series termination.

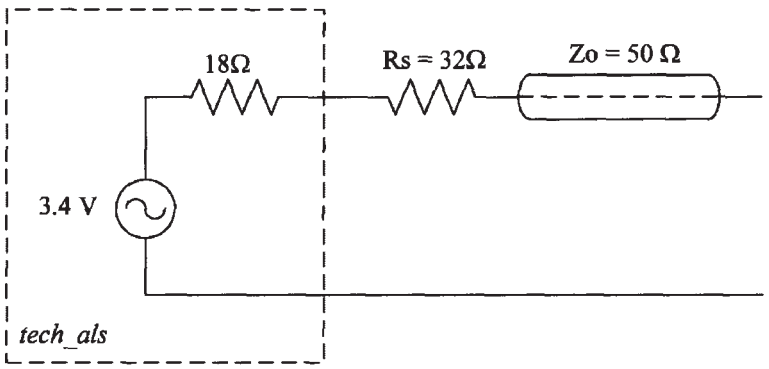



**Figure 11-9** Equivalent circuit for a series termination.

The series terminating resistor, Rs, is chosen so that the sum of it and the source impedance of the (tech_als) driver equal the Zo of the line (50 Ω). Figure 11-10 is our Signal Vision model for simulating this circuit. The far-end 10k resistor is used to simulate an unterminated line yet make provision for a measuring point to display on the model output. It has negligible impact on the actual simulation.

The result of the simulation is shown in Figure 11-11. The two curves shown in Figure 11-11 are for the signals at the driver output and at the far end of the 50-Ω trace. The source resistance of the driver and the series terminating resistor together form a voltage divider with the 50-Ω trace. So the voltage at the near end of the trace rises to one-half the 3.4-volt source voltage. Two ns later this 1.7-volt signal arrives at the far (unterminated) end of the line where the reflection coefficient is 100%. The reflected signal is then 3.4 volts. When this reflected signal arrives back at the source resistor, the

INTEL-1010
U.S. Patent 7,151,430

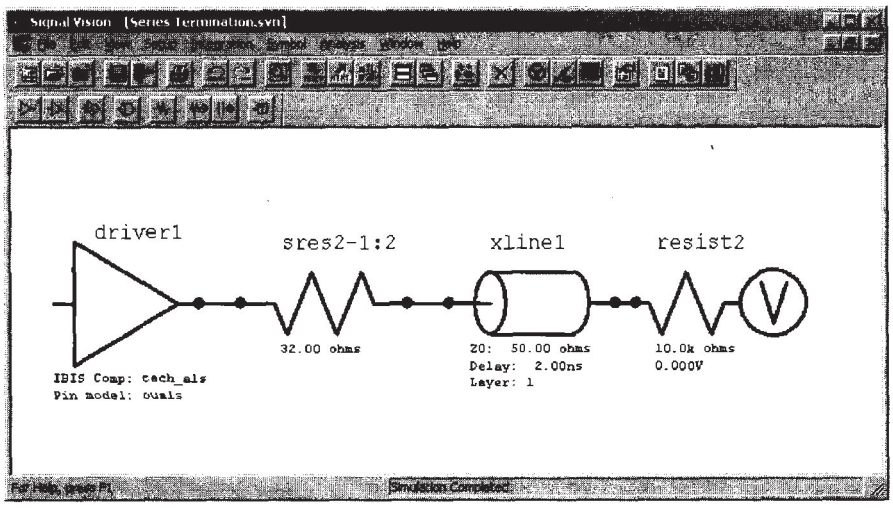



**Figure 11-10** Signal Vision model for series termination.

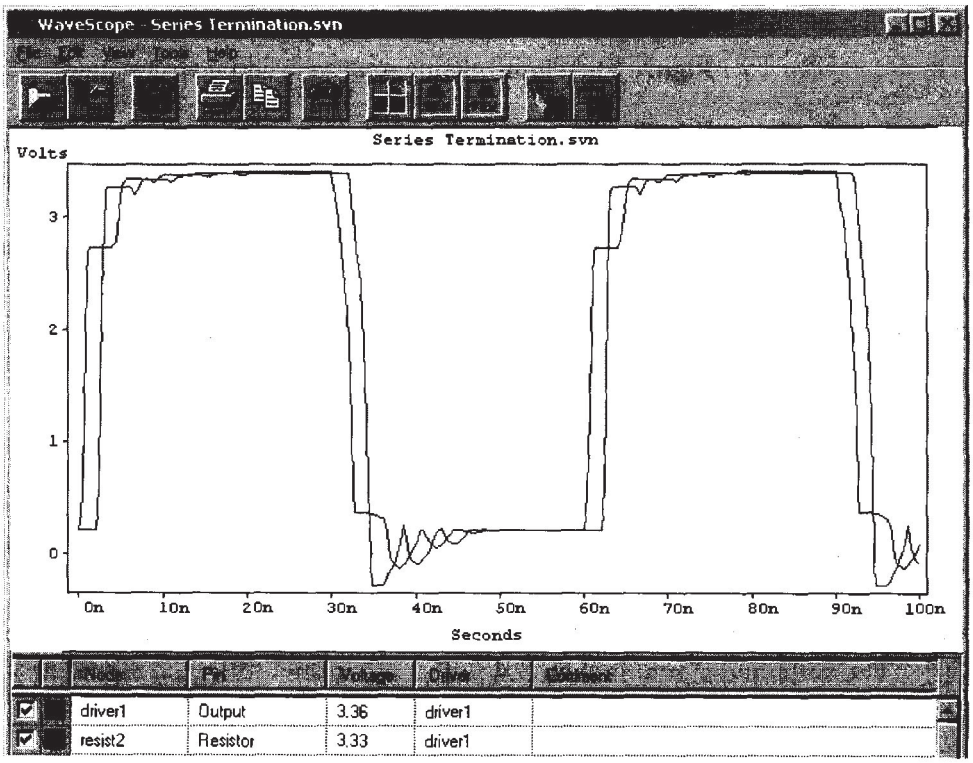
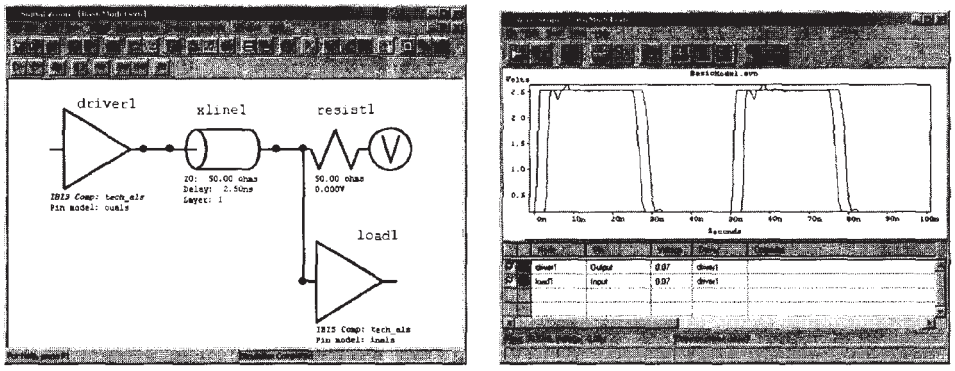



**Figure 11-11** Simulation of a series terminated trace.

energy is absorbed and the signal stabilizes at the 3.4-volt source level without further reflections.[2]

## PLACEMENT ISSUES

Figure 11-12a shows the ideal termination placement for a parallel termination. Figure 11-12b shows a resulting simulation. The results seem pretty clean. But what if we have to place the terminating resistor perhaps an inch in front of the receiver? (Note: Admittedly an inch might be a worst-case scenario, and there might be some other solutions available, such as internal routing channels and populating the back side of the board. Nevertheless, we sometimes are faced with dimensions of this magnitude.)




(a) (b)

**Figure 11-12** Ideal placement of a parallel termination.

If we place the terminating resistor an inch in front of the receiver, that means there is an unterminated transmission line segment between the terminating resistor and the receiver. The simulation for this, and its result, are shown in Figure 11-13a and 11-13b. There are some pretty healthy spikes that result from this placement.

[2]The signals are not perfectly clean in the simulation because the source impedance in the model is not perfectly linear, nor is the source resistance the same for both the high and low signal states.

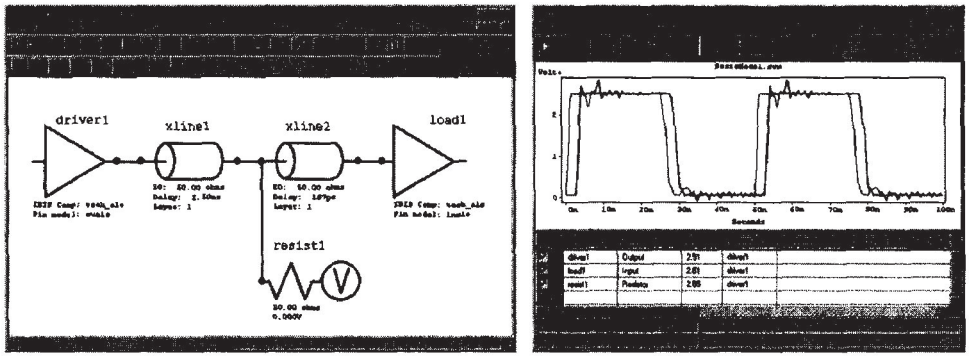



**Figure 11-13** Placing the terminating resistor before the final device can cause reflections.

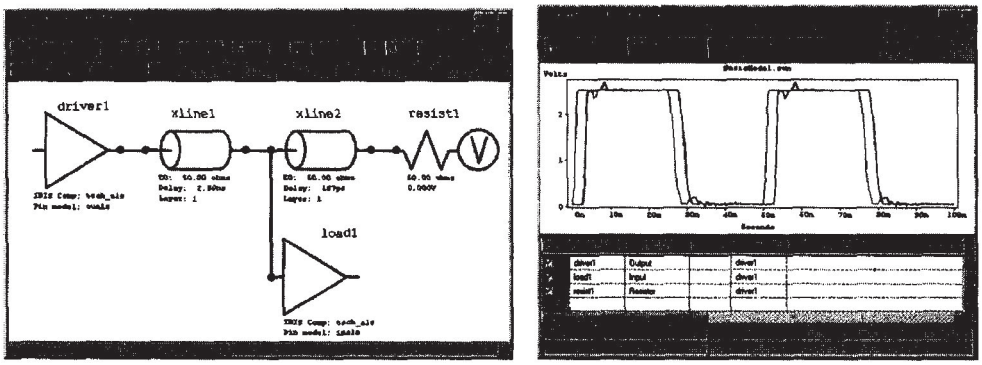



**Figure 11-14** Placing the termination resistor after the final device is nearly identical to the ideal placement.

A better solution is simple but the implications are not always recognized by designers. Instead of placing the termination resistor *before* the receiver, place it *after* the receiver. Seems simple, doesn't it? The difference is that now there is a terminated transmission line segment between the device and the termination. Before, there was an unterminated segment; now there is no unterminated segment. Figure 11-14a and 11-14b show this simulation and its result. The results are virtually identical to Figure 11-12a and Figure 1-12b.

The simple act of placing the termination after the receiver instead of before the receiver makes all the difference in the world.

## BRANCHES, OR YS

It is not uncommon for a trace to branch into two or more segments somewhere along its length. We often call this a Y. If we are dealing with controlled impedance traces we must be very careful how we handle these branches.

Figure 11-15 illustrates three different branch situations. The first (a) illustrates a 50-Ω trace that branches into two individual 50-Ω traces. A designer might incorrectly interpret the design rule that trace impedances must be constant everywhere along the trace as applying here, believing that this approach is correct. In fact, the impedance is not constant everywhere along this trace. Two parallel 50-Ω traces (like two parallel 50-Ω resistors) look like a single 25-Ω trace. Therefore, the 50-Ω trace from the driver "sees" a shift to 25 Ω at the point where the traces Y and there will be a reflection from that point.

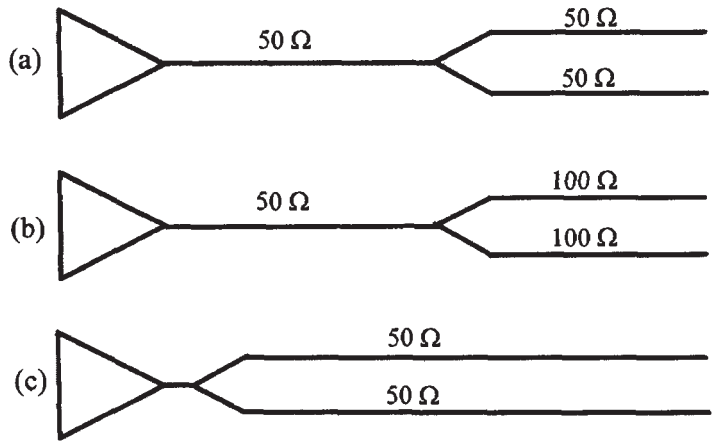



**Figure 11-15** Different techniques for handling branches, or Ys, in a trace.

There are two possible ways to handle such a branch correctly. Either one will produce a clean waveform with no reflections. One of these is shown in Figure 11-15b. Design the traces after the Y as 100-Ω traces. Two parallel 100-Ω traces will look like a single 50-Ω trace and there will be no impedance discontinuity or reflection at the Y. The 100-Ω traces, of course, must then be terminated correctly with 100-Ω resistors.

INTEL-1010
U.S. Patent 7,151,430

The other way to handle a Y is shown in Figure 11-15c. Bring the branch in very close to the driver so that the impedance discontinuity happens well within the critical length. Then the discontinuity will have little effect.

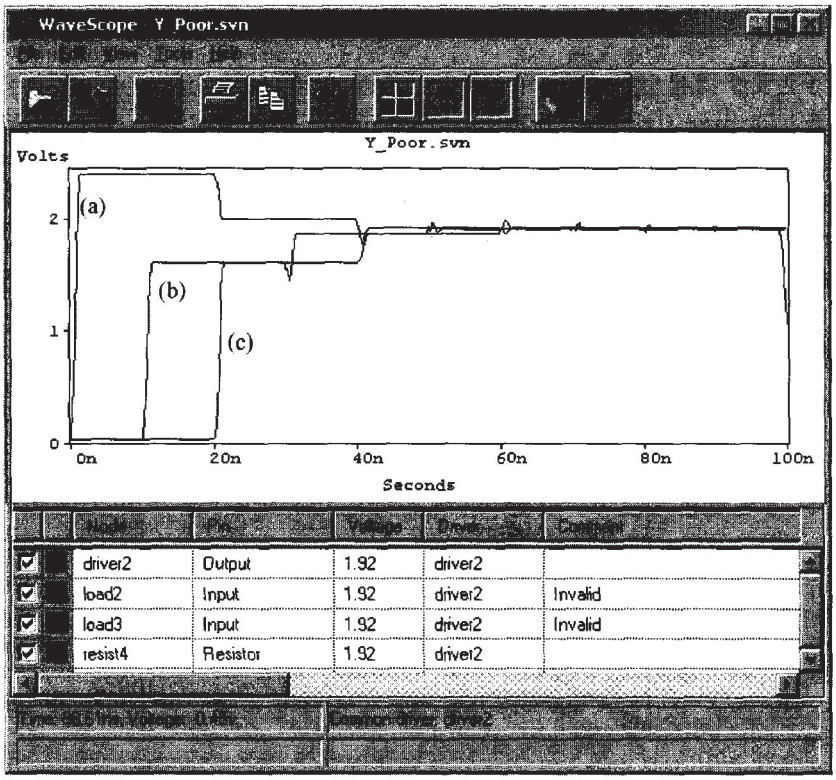
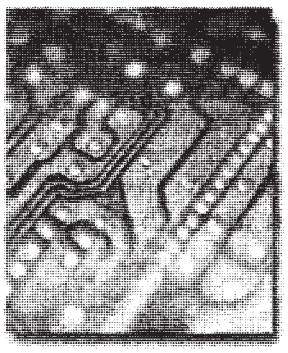



**Figure 11-16** Simulation of an improperly constructed branch, or Y. The driver signal is denoted by (a). The reflection from the improperly constructed branch occurs 10 ns later (b). The final signal at the termination is denoted as (c).

Figure 11-16 illustrates the results from a simple simulation of the improper way (Figure 11-15a) to handle a Y. A 50-Ω trace extends 10 ns and then branches into two 50-Ω traces for another 10 ns. The end of each branch is terminated in a 50-Ω resistor. There is an obvious reflection caused by the Y. The pattern is typical of a load (in this case 25 Ω) that is lower that the characteristic impedance of the trace (starting out at 50-Ω, see Figure 11-7). The reflection at the Y (at 10 ns) is evident.



# 12 CROSSTALK

In Chapter 9 we talked about how current flowing along traces can radiate EMI. In Chapter 10 we talked about how traces couple to adjacent planes, how that relates to return currents, and then to impedance control. In Chapter 16 we will talk about differential traces coupling to each other and the implications that has for signal integrity and impedance control. All of those coupling mechanisms are essentially the same; they rely on the facts that (a) like charges (electrons) repel each other, and (b) flowing electrons can create magnetic fields that induce currents in other conductors. In this chapter we look more closely at the specific coupling mechanism that can occur between adjacent traces on the board itself.

## FORWARD VERSUS BACKWARD CROSSTALK

Consider the case where we have two traces adjacent to each other (Figure 12-1). Current (the flow of electrons) flows down one of the traces (we call this trace the *aggressor* or *driven* line). That current will couple into the adjacent trace (called the *victim* line) and create two different noise signals. One of those noise signals will flow in the victim trace in the same (forward) direction as the aggressor current. The other noise signal will flow in the victim trace in the opposite (backward) direction. These two currents have different characteristics and degrees of importance in our circuits.

Consider a current flowing down the driven line (in Figure 12-1) and consider an electron as it passes by the specific point X. Electrons are negatively charged particles. Since like charges repel each other, electrons at that point along the victim line will be "repelled" or will move away from point X. They will move in either direction (arrows labeled (3)), so there will be both a backward and a forward component to this reaction. This kind of coupling is almost exactly what we



INTEL-1010
U.S. Patent 7,151,430

see in capacitors (electrons flowing onto one plate "repel" electrons from the other plate) so this is often referred to as *capacitively coupled crosstalk.*

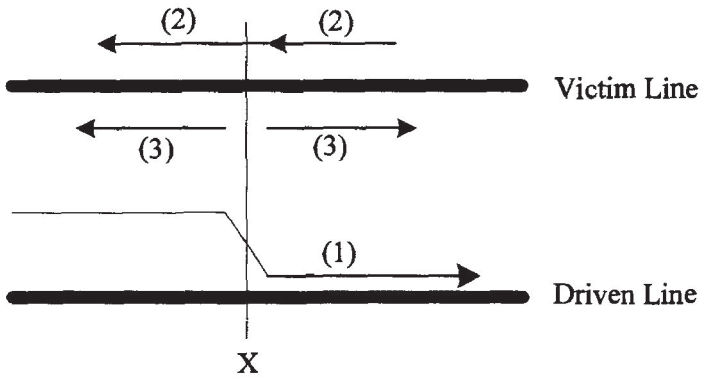



**Figure 12-1** There are two causes of crosstalk, inductive and capacitive coupling, which lead to two types of crosstalk, forward and backward.

The current flowing in the direction of the arrow on the driven line will generate a magnetic field around the line. That field will intersect the victim line and induce (or generate) a current going the opposite direction in the victim line. This mechanism is exactly the same mechanism we find in transformers and in motors and generators. In fact, we can think of the driven and victim lines as being the primary and secondary windings of a poorly designed transformer. The induced current (2) always goes in a direction opposite that of the driven current. We call this induced current *inductively coupled crosstalk.*

Note that both these mechanisms (capacitively and inductively coupled crosstalk) depend on the fact that the driven current is changing. No coupling occurs if no (or a constant DC) current is flowing. The faster the current is changing (i.e., the higher the frequency or the faster the rise time) the stronger the coupling between the traces. It also seems intuitive that the closer the traces are, the stronger the coupling will be. Already we have two secrets for reducing crosstalk (coupling): *Slow down the signals and spread the traces further apart.* (Wasn't that easy?)

There is one other point we talk more fully about later. Note that capacitively and inductively coupled crosstalk tend to reinforce themselves in the

backward direction. Both create components flowing in the reverse direction. But they tend to cancel each other in the forward direction. Capacitively coupled crosstalk creates a forward component that flows in the same direction as the driven current, but inductively coupled crosstalk flows in the opposite direction. These components almost exactly cancel, particularly in stripline environments. This intuitively leads to another design rule: *If crosstalk is an issue, try to keep all the sensitive traces in a stripline environment.*

*Backward Crosstalk* There are significant differences in the nature of forward and backward crosstalk. Consider Figure 12-2. This figure is in two rows, Row (1) and Row (2), with four frames each. Look first at Figure 12-2a, Row 1. The two rows of dots at the top represent elements of a victim trace. There are two rows because we are going to let the top one represent backward crosstalk and the other one represent forward crosstalk. The single row underneath them represents elements of the aggressor (driven) trace. There will be a signal, transversing from a low signal level to a high signal level, flowing from left to right along the aggressor trace.

The two rows, 1 and 2, differ in that the victim trace in Row 2 is terminated in a high impedance at the near end. The victim trace in Row 1 extends indefinitely in the backward direction.

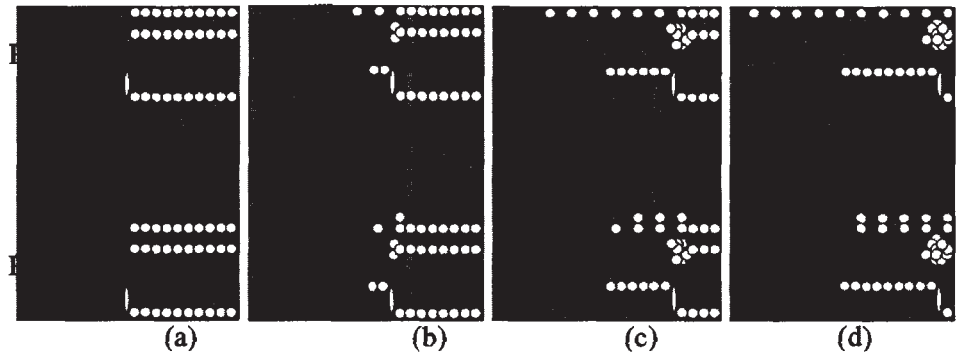



**Figure 12-2** Forward crosstalk has a pulse width approximately equal to the rise time of the aggressor pulse with an amplitude that increases over the coupled length. Backward crosstalk has an approximately constant amplitude but increases in pulse width as the coupled length increases.

In the first frame (Figure 12-2a), the aggressor signal is just starting down the trace. In the second frame (Figure 12-2b), the signal has started part way down

the trace. In the third frame (Figure 12-2c), the signal has moved well along the trace and by the last frame (Figure 12-2d) the signal has reached the far end of the trace.

Let's look first at backward crosstalk (top pair of dots) for Row 1. Note how, as the driven signal moves down the aggressor trace, each time it passes by a victim element it *kicks* that element to the left. Each element, in turn, starts moving backward before the next element gets kicked. What results is a string of elements moving to the left (backward) stretching out in length. Especially note that by the time the aggressor pulse gets to the far end of the aggressor trace, the backward string of elements is twice as long as the coupled length between the victim and aggressor traces is. This happens because the first element starts moving at the same moment the aggressor trace starts moving. When the aggressor signal reaches the far end, the first element has moved equally far in the opposite direction. But the last element along the victim line doesn't even start to move backward until the aggressor signal reaches the far end. Thus the string of backward-flowing elements stretches twice as far as the length of the coupled region between the traces.

There are two extremely important things to notice here. First, we are talking about *time*, not units of distance. It takes a certain amount of time for the aggressor signal to travel down the trace. We call this the *propagation time* of the signal traveling along the trace. Second, if we assume the signal flows at 6 inches per ns, and each circular element represents .25 ns, then the coupled region between the two traces is 2.5 ns, or 15 inches. The backward crosstalk signal stretches twice as far, or 5 ns.

We can generalize this as follows: *The backward crosstalk pulse is (almost) constant amplitude in magnitude but twice as wide as the propagation time represented by the coupled region.* This backward crosstalk pulse width is what so many people have trouble understanding. The width is not a function of coupling strength. Width is purely a function of the *length* of the coupled region. The statement is worth repeating: *The backward crosstalk pulse width is purely a function of the length of the coupled region.*

Now, Row 2 shows exactly the same thing with one modification. The vertical line at the left of the victim trace represents the beginning of the trace. It might represent a driver or a load. As the first element of the backward crosstalk row gets kicked backward by the aggressor signal, it reflects off the end of the trace and begins flowing in the forward direction. This is not to be confused with forward crosstalk. It is simply a reflected backward crosstalk signal. (One obvious question

INTEL-1010
U.S. Patent 7,151,430

is this: Could we eliminate this reflection if we correctly terminated this end of the victim trace using the transmission line termination principles we learned about in Chapter 10? The answer is "Yes." More on that later.) In this case, the first element of the backward crosstalk signal reaches the forward end of the trace at the same time the aggressor signal does. But the last element on the victim trace has just started its travel in the reverse direction. Before the show is over, it must travel all the way back to the beginning of the trace, reflect off the beginning of the trace, and travel all the way back to the forward end again—a distance that takes twice as long as it took the aggressor pulse to travel the length of the coupled region. Again, we see that the backward crosstalk pulse width is twice as long as the propagation time down the coupled region.

A word about the magnitude of the backward pulse: If the coupled region is short, the magnitude of the backward signal will be small. It will grow as the coupled length increases. But there is a limit to how high the magnitude will grow until it stops and just levels off. That limit in magnitude is reached when the coupled length is about equal to the so-called *critical length* we have already discussed (Chapter 10). That is, the magnitude of the backward coupled pulse reaches its maximum about when the length of the coupled region equals one-half the rise time of the aggressor signal. This should be considered more as a rule of thumb than as a factual statement, but it is a rule that seems to hold approximately true most of the time.

What that maximum level actually is is a different story altogether. We talk more about that later on.

In summary, backward crosstalk grows fairly quickly to a *constant magnitude* pulse whose pulse width is *twice as wide* as the propagation time down the coupled region.

*Forward Crosstalk* The second row of dots for each victim trace represents what happens with forward crosstalk. In Figure 12-2b Row 1 or Figure 12-2b Row 2, as the aggressor signal just starts to propagate along the trace, the first element in the forward crosstalk row is kicked forward. It immediately *bumps* into the next element. As the aggressor signal moves forward one more increment, the first two victim elements get kicked further along and bump into the third element.

This continues as the aggressor keeps kicking the forward crosstalk elements along. By the time the aggressor signal reaches the end of its trace, the forward crosstalk elements are all bunched together at the far end of the victim trace.

INTEL-1010
U.S. Patent 7,151,430

Note that the forward crosstalk illustration is exactly the same for Row 1 and Row 2. It does not matter whether the victim trace begins at the beginning of the coupled region or extends further back. Since the forward crosstalk signal never travels in the reverse direction, it doesn't care whether there is a reflecting barrier or not.

The quantity of the forward elements represents the magnitude of the forward crosstalk signal. As suggested in Figure 12-2, the forward crosstalk signal will continue to increase in magnitude the longer is the coupled region. Although there is a theoretical limit to how high the forward crosstalk signal can grow, we are never likely to reach that limit on circuit boards (the coupled region can't be long enough). For our purposes, the forward crosstalk signal simply continues to grow larger and larger as the coupled region increases.

There is one other distinctive thing about the forward crosstalk signal. All the elements representing the signal are clustered on top of each other. This represents the width of the forward crosstalk pulse. In theory, the width of the forward crosstalk pulse is no wider than the rise time of the aggressor signal that creates it. The pulse starts to build as the aggressor signal starts to rise, and it has finished building when the aggressor signal has reached its maximum value.

In summary While the backward crosstalk signal is relatively constant in magnitude and has a pulse width twice as wide as the propagation time down the coupled region, *the forward crosstalk signal has an amplitude that continually increases as the coupled region increases and has a fixed pulse width equal to the rise time (or the fall time) of the aggressor signal.* Simulations in Chapter 13 will illustrate these differences quite clearly.

We also show in Chapter 13 that for all practical purposes, forward crosstalk is very small in microstrip environments and virtually nonexistent in stripline environments. Therefore, for the rest of this chapter, we focus only on backward crosstalk.

## ESTIMATING CROSSTALK

*Calculations* We can develop theoretical formulas for calculating crosstalk based on some simplifying assumptions, but the practical reality is that crosstalk is very difficult to calculate, at least without special tools. The HyperLynx tool does a credible job simulating crosstalk, as will be shown in the next chapter. Otherwise,

INTEL-1010
U.S. Patent 7,151,430

there aren't very many good ways for quantifying what the level of crosstalk might be in your particular design.

There are three things that affect the magnitude of the (backward) crosstalk noise signal:

1. The degree of coupling that exists between the aggressor and victim traces.
2. The distance over which that coupling occurs.
3. The effectiveness of any trace terminations that might exist.

*Coupling* Referring to Figure 12-3, the degree of coupling is proportional to, and never exceeds the value determined by Equation 12-1:[1]

$$\frac{1}{1+\left(\frac{D}{H}\right)^2} \quad or \quad \frac{H^2}{H^2+D^2} \qquad [12\text{-}1]$$

where D and H are as defined as in the figure. From this, it is apparent that (as we already knew) crosstalk goes down as H goes down and as D increases. So, the first rule in minimizing crosstalk is to create your stackup so that the traces are close to their reference planes, and spread the traces out.

It is instructive to note something else. Trace width does not appear in this relationship. Trace width has only a minor effect on crosstalk. The other two dimensions are far more important than width (at least for reasonable dimensions).

*Distance* If the coupled region is very short, crosstalk does not have time to develop. Therefore, it stands to reason that the crosstalk amplitude will be small for short coupling regions. We have already discussed the fact that the amplitude is not related to distance for long distances. (Remember, we are talking about backward crosstalk here.) The magnitude builds up as the length of the coupling region in-

[1] Johnson, Howard, *High Speed Digital Design, A Handbook of Black Magic*, Prentice Hall, 1993.

INTEL-1010
U.S. Patent 7,151,430