

12. Analog Circuit Design for Fun and Profit

The first volume of this series of books dealt mainly with how to design analog circuits. It was an interesting collection of ideas, anecdotes, and actual descriptions of the processes used by various well-known analog circuit designers to accomplish their goals. You won't find much of that sort of thing in this chapter (although I hope it will be interesting nonetheless).

The inspiration for this chapter arose in part from a comment in the chapter of the first book submitted by Derek Bowers of Analog Devices. He admitted that some of his most elegant circuits turned out to be poor sellers, while other circuits (of which he was not particularly proud) became multi-million-dollar successes. In this chapter, I will offer a few words of advice to fledgling analog design engineers in an effort to help them distinguish between good circuits and good products. In addition, I'll alert fledgling circuit designers to a new person they will eventually encounter in their careers—the Marketeer.

Why I Wanted to Be an Engineer

As an engineering student, you probably think you have a good idea of what engineering is all about. I recall my goals when I entered engineering school in 1971. It was all so clear then. High school students with an aptitude for math and science were destined to become engineers, and I was one of them. Four years of college would be followed by a secure career in the Engineering Lab, designing circuits that would change the world. I worked a few summers as a Technician, and I knew what engineers did. They designed circuits, gave hand-drawn schematics to the drafting department to make them nice and neat, then had the Technician round up the parts and build a prototype. Then the Engineer would come back to the lab and test the prototype, and blame any shortcomings on the lousy job the Technician did building the prototype. After a few iterations, the prototype would be declared a success, the Engineer would disappear for a few days to do something in his office, then come back with a hand-sketched schematic of the next circuit. And life went on.

Then I graduated and became an Engineer. 1975 was not a good year to become an Engineer. Defense contractors had fallen on hard times, with the Vietnam War winding down. They weren't hiring Engineers. The economy was in tough shape, and the industrial companies were also hurting. Many of my fellow new Engineers were scrambling to get into a graduate school to hide until the job market got better. I was one of the

lucky few that actually found a job—mostly because I had worked part-time as a Technician to pay for school, and I therefore had “experience.” Just getting an interview in 1975 wasn’t easy. In fact, I had already been out of school for over a month when I got a call from the university’s placement office to tell me that a company had reviewed the graduating class’s resumé book, and had invited me for an interview. My resumé touted some knowledge of both analog and digital circuits, and I claimed I knew which end of the soldering iron to hold. I could cobble a collection of TTL gates together to do something, and could design a circuit with an op amp in it. I even had some experience in using and testing analog-to-digital converters. Fortunately, these were important things for this position, since my grade-point average was nothing special (too many extra-curricular activities . . .). I got the job.

Then I found out what Engineering was really like.

The first day on the job, my boss handed me the manual for the then-new Intel 8080 microprocessor, and told me to read it. Every day for the first week, he’d come into my office (actually, our office—four of us shared the same office) and ask me how I was doing. He was a pretty good engineer and teacher, and I got the chance to ask him some questions about things I hadn’t quite understood. It went well.

Then one day, he handed me a schematic of the 8080-based system he had just finished testing. This was my chance to see how he had designed the system’s bus structure, and implemented the various sub-systems and their interfaces to the processor. It was mostly pretty straightforward stuff—all digital at this point. Then a few weeks later he came into my office and asked me to design an analog I/O interface for the system, including the signal conditioning, A/D and D/A conversion, logic interface, and various other pieces. This was the moment of truth—I was on my own for my first design.

I had a handful of specs for the instrument we were supposed to interface with—voltage levels, source impedances, bandwidths, etc. I had the specs of accuracy of the original system. I had the manufacturers’ data sheets for every component imaginable. And a week or so later, I had a design done—one of those hand-drawn schematics I had worked from as a Technician, but now I was calling the shots! Then we reviewed the schematic—the boss told me he had forgotten to mention that we needed to be galvanically isolated from the instrument we were hooking into. No problem; I had used V/F conversion for the A/D, and a few opto-isolators later I had completed the revised design, including isolation, and he signed it off. I proudly marched into the lab, handed it to the Technician, and he saluted smartly on his way to build a prototype.

Then a funny thing happened. The design part stopped for a long time. There was some haggling about certain parts being no longer available. The purchasing guy complained that some of them were sole-source, and he wanted everything to be multi-sourced. So I spent some time redesigning; the basic idea stayed the same, but the schematic was revised time

and time again to comply with everyone's needs. Then the software guy came over from the next office. He wanted a complete map of each I/O address, a description of each function, and the timing pauses required between operations. No problem—I wrote it all up for him over the next week or so, in between interruptions from the Tech and the purchasing guy. We met again to review what I had done, and the software guy reminded me that the last project had included some provisions for calibration and self-test. Back to the schematic—I added the required additional channels and test modes, and was finally done. The prototype had grown somewhat, and I was amazed that the Tech was still speaking to me (he'd seen all this before).

Then the boss came in and asked me to document the operation of the circuit, including a description of every component's function. The purchasing guy came in with the manufacturing guy and they asked me for a complete parts list and bill of materials, and to sign off the final schematic. After a few iterations, everything was signed off, and the product went into production. I was eager to get to the next project.

Then it got interesting. The main processor board that my boss had designed developed reliability problems—it was an obvious bug in the clock circuit, which I found by putting my finger on the pull-up resistor for the +12V clock. Half-watt resistors get hot when dissipating a whole watt. I got to fix that one. The analog input section worked fine when we used one manufacturer's V/F converter, but was noisy when we substituted an "equivalent" from another manufacturer. I tracked the problem down to a difference in the power-supply decoupling needs of the two, and conjured up a scheme that was suitable for both versions.

As production started, I was often called to determine if a component substitution was possible because one or more parts was temporarily out of stock. In some cases, the substitution had already been done, and I had to figure out why it didn't work.

A full six months later, my boss asked me to design another circuit. Think about it—almost a half year between designs. Life as an Engineer was turning out to be very different from what I had expected. At least I was getting paid.

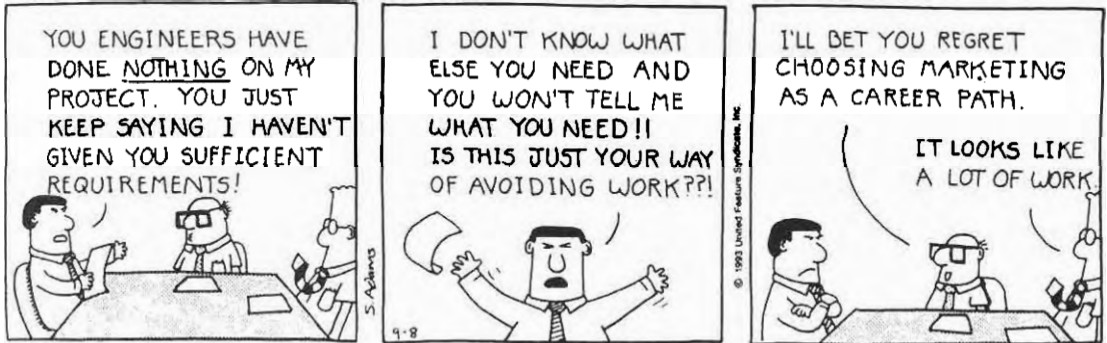
When I was actually designing circuits, I discovered an assortment of interesting processes at work. There is recall—remembering previous circuits that may help solve the problem at hand. There is invention—defining the problem, and creating a new solution for it. There is experimentation—often, a difficult problem will require numerous tries to get to the right solution. In some cases, these processes are aided by various embodiments of design tools, from decade boxes to advanced state-of-the-art expert-system-based software. Lots of tools are available to help the designer create a solution to a problem. And each idea is weighed carefully, using all necessary processes and tools, against an endless parade of design trade-offs, to improve reliability, increase production yield, and lower costs while maintaining or improving performance.

But it never ends at the design phase. After a circuit is done, and the first units are reduced to physical hardware, it remains to determine if the thing actually solves the problem it was intended to solve. Testing, debugging, characterizing, and (often) doing it all again are part and parcel of the product development process. And lots of other authors have described their personal versions of this process in their chapters.

I occasionally design circuits at home for recreation. Most are not the same as the kind of circuits produced by my employer, but my engineering training and avocational interest in electronics motivate me to keep designing circuits from time to time. Nobody will ever buy them. Total production volume is usually one. And I get a real thrill when I see one of them work for the first time. And any engineer who has never felt the thrill of seeing his first units work perfectly first time out will probably not stay an engineer very long. In fact, the experienced engineer should feel the same sense of excitement when “it works.” Often, circuits don’t work the first time. After an appropriate period (hopefully a short one!) of self-flagellation, the analysis of the circuit and troubleshooting begins, usually revealing an oversight or similarly simple error. The joy of finding the error usually makes the eventual event of a working circuit anticlimactic. And building circuits at home—with no formal documentation or parts lists required—the experience is as near to pure engineering as it ever gets. When I design circuits for myself, I define, design, build, test, redesign, rebuild, and use them. Unfortunately, it doesn’t work that way in the real world. Most of the time, someone else is telling you what to design. And someone else is building and testing “your” circuits. Yet someone else may redesign them. And most importantly, someone else is using your circuit, and has probably paid money to do so.

A design engineer should never lose sight of the fact that his continued gainful employment is dependent on producing circuit designs that solve a problem for which his employer will collect revenue. Circuit design for fun is best left to the home laboratory, for those engineers who still have one. Circuit design for profit is serious stuff. If you can combine the two, consider yourself lucky. Then find a second spare-time leisure pursuit having nothing to do with engineering.

I don’t design circuits for a living any more. I moved from Engineering into Marketing (by way of a few years in Applications Engineering) some years back, but stayed in the electronics industry. While some marketing skills are easily transportable across industries (especially in promotion and merchandising), the product-definition part of marketing generally is most successful if the practitioner is close to the technology. I have had occasion to recruit marketing engineers from the technical ranks of our customers as well as the design and product engineering areas of our own company. Most have done well, but all have expressed great surprise at the amount of work involved in the job, compared to their previous lives in engineering (and most of them thought marketing was going to be easier!).



DILBERT reprinted by permission of UFS, Inc.

Steps in the Product Development Process

The following steps broadly outline the product development process. In all cases, the “you” refers to yourself and your colleagues in whatever enterprise employs you. Product development is seldom a single-person endeavor.

1. **Concept**—Find a problem that you think you can solve.
2. **Feasibility**—Can you really solve the problem?
3. **Realization**—Design and build the product.
4. **Introduction**—Getting the product to the customers you don't know.
5. **Closure**—Move on to the next problem.

Step 1. Concept—Find a Problem That You Think You Can Solve

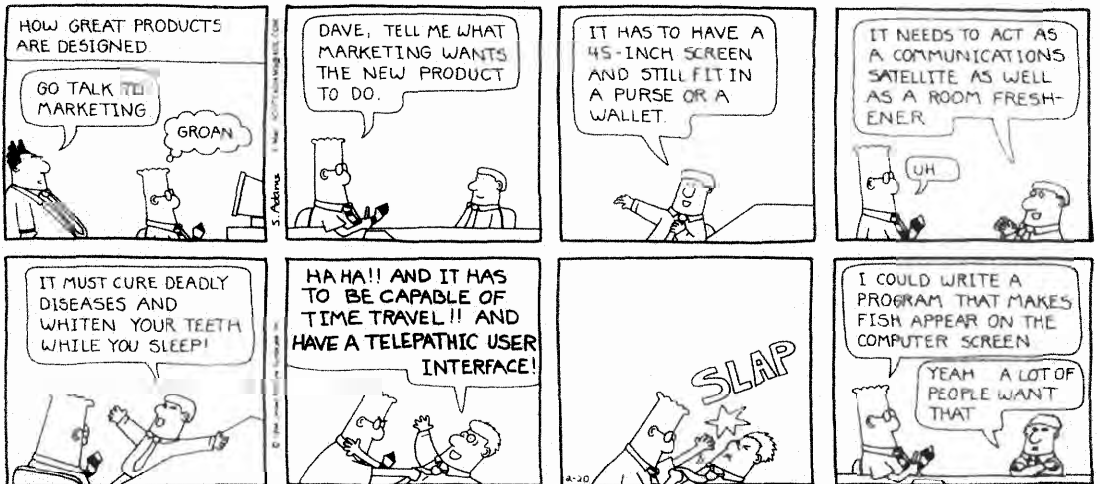
A product is (obviously) something that is meant to be produced (manufactured, delivered to someone for use, sold, consumed; take your pick). The point is that in the present era, very few circuits are designed for recreation only. Hardware circuit hackers are still out there, including the radio amateurs, but the fact is that most circuits are designed by engineers toiling for an employer. And that employer has an obligation to its customers and shareholders to create things that solve its customers' problems, and in so doing, generate a profit. Oftentimes, these solutions take the form of innovative circuits, processes, or architectures. However, there is a weak correlation between commercial success and technical elegance or sophistication.

A product must deliver benefit to the customer; it must solve his problem. A circuit can be a part of a product, but it is never the product. A user needs to see some benefit to using your circuit over another. I recall reviewing one particular product proposal from a design engineer that detailed a novel approach to performing analog-to-digital conversion. It seemed clever enough, but as I read it, the performance claims were no better than what existed on the market already. A cost analysis indicated

no improvement in cost over what existed already. Power wasn't better. No particular features seemed to be obvious—it was just another A/D converter. I just didn't see any great benefit to a customer. So I called the designer and asked him what I had missed. He replied that the architecture was novel and innovative, and there was nothing like it. We reviewed the performance he thought he could get, and a chip size estimate. After about fifteen minutes, I asked him to compare the proposed chip to another we already had in development. There wasn't any advantage obvious. Then I asked him to compare it to various academic papers. He replied that his architecture was more "creative" than various proposed schemes. But when I asked him to show me where this idea would lead (higher speed, more resolution, lower cost, added features, scalability, user features, etc.), he drew a complete blank. Even assuming device scaling or process additions, he (and I) couldn't think of where this would lead. I asked if the inspiration had come from a particular application or customer problem. The closest he could come was a personal-computer add-on card that he had seen once. He had no idea if the board was a big seller or not.

The project was shelved. But I suspect that one day his novel architecture (or more likely, some part of it) will be useful in solving a very different problem.

I have also had the opportunity to deal with newly hired marketing engineers. Their zeal for the perfect product often blinds them to reality, as noted in the comic strip. In defining specifications and features for a new product, there is the temptation to add every conceivable feature that any customer has ever asked for during the process of fielding requests from salespeople and customers. This leads to the frustration that engineers often have when dealing with marketeers. On the other hand, I have observed situations where the engineer has been unable to promise that a certain specification can be met, and a less-than-satisfying compromise was offered. Both parties need to spend some time analyzing which combination of features and specifications meets the requirements of the ma-



DILBERT reprinted by permission of UFS, Inc.

majority of the customers, and settle on these. “Creeping featurism” must be avoided, even if a customer calls just a week before design completion asking for one more feature, or if the designer discovers “a neat trick that could double the speed” at the last minute. Stick to the script! Last-minute changes usually result in future problems.

As difficult as it may be designing high-performance analog circuits, it’s equally challenging to figure out what to design in the first place. A wonderful circuit that nobody buys is not a good product. A rather pedestrian circuit that a lot of people buy is a much better product. This is a tough concept for most of us to swallow, but it’s the truth.

Making sure you understand the problem you are solving is probably harder than designing the circuit. You have to learn someone else’s job and understand his problems if you are going to have any chance of solving them. Numerous techniques have evolved over the years. One very effective methodology currently in vogue is called “Voice of the Customer,” or “VOC” for short. The entire VOC process is lengthy and involved, and will not be described fully here, except for the first steps, which involve customer interviewing.

I recall taking an IC designer to visit a customer in the video business. The designer had some ideas for a new A/D converter fast enough to digitize a video signal. A/D converters are generally described by their resolution (measured in bits) and speed (measured in conversions per second). We talked to the customer about his whole signal chain, from input connector to digital bus, to get a feel for the components he had used in his previous design. The A/D converter was an 8-bit part, with a certain conversion speed. As we talked, the customer began to complain that he couldn’t get the resolution he wanted from the digitized image. Aha! We had discovered the problem to be solved. He needed more resolution!

I glanced at the IC designer’s notes and he had definitely gotten the point—he had written “RESOLUTION” in big letters, underlined, and circled it. Then he scribbled next to it: “Only has 8 bits now—10 should be enough.” Unfortunately, there is another kind of “resolution” in video; it refers to the number of pixels on the screen, and when a video engineer talks about resolution, he means the speed of the converter, not the number of bits! Having done a fair amount of reading in preparation for the visit, I picked up on the error and asked the customer for a clarification. It went something like, “How much resolution do you need, and what does that mean in terms of the A/D converter?” His response was ultimately in speed terms and we got the discussion back on track (I knew it was back on track when the IC guy wrote “RESOLUTION = SPEED!!!?” in his notebook). It is important to understand your customer’s business and language before you go on the interview.

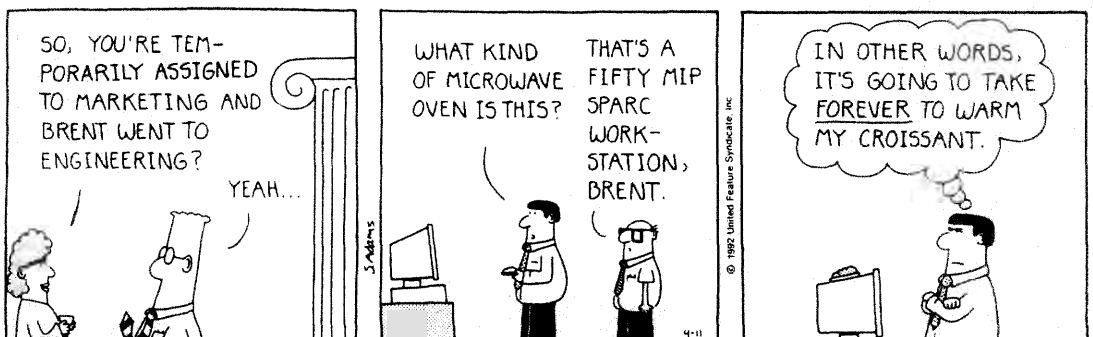
Another time, I listened to a customer complain bitterly about an A/D converter that he claimed was outside its accuracy specs. I offered to test the device for him to verify its performance. When I tested the part, it was fine, meeting all specs on the data sheet. When I returned the unit to the customer, he insisted on demonstrating to me exactly how bad the

accuracy of the converter was. I went with him into his lab, where he put the converter in the socket, and turned the system on. He then tested the system by applying a dc signal from a bench power supply to the input and displayed the digital output on a monitor. He didn't even measure the DC input—just made sure it was somewhere within the A/D converter's range, based on the reading on the front panel meter on the supply. Before I could ask how he intended to verify 12-bit accuracy with a known reference source, he showed me that the output code was very unstable, with several codes of flicker evident. This was obviously the problem—noise, not accuracy! We tried all the usual cures (changing the supply to the converter from a switcher to a linear, rerouting the grounds a bit, and adding decoupling capacitors where there hadn't been any), and each change helped. Finally, we had the output stable. A fixed input gave a steady output value, even though we hadn't checked the actual accuracy of the system (he actually had no suitable equipment for such a test anyway). But he was happy—his problem was solved. We were happy—we got the order.

The data sheet for our next A/D converter included detailed instructions on how many capacitors to use and where to locate them in the layout. It wasn't any more accurate a converter, but a lot fewer people complained about its "accuracy." And we added some tutorial information defining the various performance parameters of A/D converters, so the next customer who called complaining about accuracy would actually mean accuracy, and we would be able to diagnose and cure the problem faster.

Speaking the customer's language is critical to communicating with him. And by "language" I mean his own company jargon or slang. If you expect him to learn your terms, you'll find it a lot harder to get him to feel comfortable describing the problem he wants you to solve. And this advice applies to both engineers and marketers attempting to interview customers.

The VOC process suggests working with a number of customers to collect images that allow you to understand their problem as they see it. This is important—satisfying the customers' needs in a way that they can



DILBERT reprinted by permission of UFS, Inc.

understand is the secret to success. The next step after collecting images of the customer's-eye view of the problem is to re-state the problem in your own language so you can figure out a solution. All engineers should spend time with customers when they are in the process of discovering a problem to solve. Too often, a visit to a customer takes the form of trying to find a problem that fits your own creative solution. This violates all known problem-solving principles, but we all do it anyway. The obvious thing to keep in mind is that solutions only exist to solve problems; without a defined problem, it is only by sheer luck that a proposed solution does the job.

Some solutions are obvious—make it faster, more accurate, cheaper, lower power—but other problems exist that can be solved without the breakthrough innovations often needed to improve one of the conventional dimensions. These can only be discovered by talking to customers and analyzing the data in a meaningful way to reveal what features or qualities of a product the customer will value. But remember—customers are in a different business than you are. It is up to you to make the effort to learn the customer's business and language in order to actually understand the problem and offer a solution!

Interviewing a prospective customer involves some preparation. You should have a reasonable list of questions you want to ask, and you should be prepared to skip around the list as the conversation wanders. I have found it extremely useful to conduct interviews in teams of two. One person asks the questions, while the other scribbles the answers as fast as possible, trying to get it all down as nearly verbatim as possible. It's important to avoid adding too much commentary or analysis here—there's plenty of time for that later. Just get the facts down. If a series of questions has been missed, the note-taker can steer the conversation back to the areas missed. When I have tried to do customer interviews solo, I have often reviewed my notes only to find phrases like "He says the biggest problem is" or "The preferred package is," where I've been unable to get it on paper fast enough, and the conversation has taken a turn to another topic too quickly. A second pair of ears and hands can help immensely.

After the interview (which should end when the customer signals to you that he's done, not when you think time is up or have another appointment), the interview team should compare notes and make sure that both have heard the same things. It is useful to re-construct the entire interview as it occurred to help the recall process. Clean up the notes as soon as possible so they can be shared and reviewed later in the process.

After you've collected several interviews, the process of analyzing the data can begin. There is a strong temptation to give more weight to the last inputs you've received, unless you've taken the time to get them all in readable form. There is also a temptation to downplay inputs that contradict your own basic assumptions. Don't do it! Always remember that your product will be more successful if it solves the customer's problem than if it fits your personal model of the way things should be.

The process used for analysis of the raw inputs can be complicated or simple. The underlying principle is always to get a customer's-eye view of what is important, and respond to it in a product definition. Commentary like, "They all say they want power consumption less than 50 milliwatts. That's ridiculous—there's a 10-horsepower motor in the system! Besides, my circuit topology takes that much just to power up the output driver," is to be avoided. Things that appear from your own perspective to be obvious contradictions like this need to be reviewed and understood, not dismissed. In the case just mentioned, you may discover that the circuit you are designing is used by the thousands in the system, and that big motor is only used to move the system into position, and powered externally. The constraint on power is probably quite real. And you should figure out how important your output driver really is in his system.

Step 2. Feasibility—Can You Really Solve the Problem (and Is It Worth Solving)?

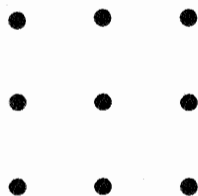
This step follows whatever analysis tools you use to reveal the features and performance requirements of the solution you are planning. VOC, QFD, and other methods can be used, but none is a substitute for experience, judgment, and general knowledge. At this point in the process, you should feel that you understand the requirements of the customers, and the first-cut solution is probably getting clear in your mind. In fact, you may think you have enough information to actually design a circuit at this point. Resist the temptation! You are in for some surprises. At this point, don't even try to complete the design—you'll find some feature you left out, or more likely, you'll have included a feature that only one customer (probably the last one you talked to!) wanted and which sounded like an interesting design challenge. Keep it simple at this point. Don't worry too much about the cost, or even the detailed architecture inside. Take a stab at the specs and features that seem important to the customer and difficult to meet, but don't waste too much time at this point.

There are usually several alternatives to solving the customer's problem. Usually the customer won't care much about the internal architecture, so you have a lot of freedom. You should get one pretty conservative solution defined quickly, then take some time to find alternatives that are better from the standpoint of cost, power, or ease in meeting some important specification for the customer. And feel free to think "outside the box."

This last expression comes from a course I once took on innovative problem-solving. A very simple puzzle is presented—draw three parallel rows of three dots each on a piece of paper; connect all nine dots by drawing four straight lines, never lifting the writing implement from the paper. The solution to the problem was an example of "going outside the box," as shown following.

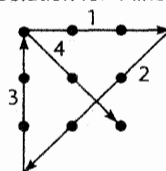
I had seen this puzzle before, and knew the trick, while others in the class were claiming it couldn't be done. I smugly told the instructor I'd be glad to show the others, since I knew the answer. Unfazed, he gave me

Going outside the box.



A. With four lines . . .

Solution for 4 lines:



the assignment of completing the puzzle by drawing only *three* lines. This put me in the same bewildered predicament as the rest of the class. After several minutes of torture, the instructor revealed the solutions—using four lines, then three, two, and even one line! (Solutions appear at the end of this chapter. . .)

You should also talk to people that can provide assistance—other designers, applications or marketing engineers, anyone with some experience. The chances are that a circuit to do something like this has been tried before. Remember the example of the A/D architecture without a home? Perhaps this is the right problem for that solution. Don't forget to check the literature. There's no sense in re-inventing the wheel—in fact, if someone else has a patent on that particular wheel, it could get expensive. And if you come up with an idea that looks original and has benefits over previous work, consider patenting it.

If it turns out that the customer's problem does not have a solution that you can find that satisfies all the needs, there are a couple of options. One is to give up and move to the next problem. This is sometimes the best course of action. Some problems just don't have satisfactory solutions yet. File it away, keep in the back of your mind exactly what makes a solution impossible at this time, and keep your eyes open for the enabling technology. At that point, go back and see if the problem still needs a solution.

If you can't find a way to meet all the required specs, try to meet as many as you can, and try the solution out on a few willing customers. It may turn out that solving three out of four is good enough. It may be three more than anyone else has proposed!

Whether you think you're meeting some or all of the requirements, when you are closing in on the implementation, you must check to make sure you're still on course. Try describing your solution in terms of the customer's problem as you understand it. Survey methods can be used to rate individual features for importance, and "kill specs," or a series of loosely structured second-round interviews with willing customers, will work. When proposing a solution, be open to suggestions for improvement. This is not the time for defending "your" solution; after all, it isn't a solution yet—only an idea. If you are willing to make changes, customers will be willing to suggest them. And you'll find out quickly what is important that you missed, and what is superfluous. Pay attention, and bring someone with you again to take detailed notes for review soon after the visit.

At some time, you will have to decide if this problem offers enough financial incentive for you and your colleagues to spend your time (and your employer's money) solving. This is the best time, before you invest a lot of time in the detailed design. I don't advocate a detailed market analysis that attempts to prove beyond a shadow of a doubt that this is the right thing to do. Instead, ask the customers if this is the right problem to solve. If they say no, figure out the right problem to solve, and solve that one instead. If you have your heart set on solving a particular problem, make sure somebody in your company solves this customer's most important problem before someone in another company does it.

You should go through the exercise of making sure the numbers add up. If you talk to ten customers in a certain end market, and they all claim 30% market share, you have a problem. You may be able to get some data from an independent source to determine the actual shares (and thus the volume estimates for your solution), but often you will have to rely on your own estimate. And determining which of these ten customers is likely to win in his market will be based on your own feelings about their relative competence as much as any market research you will be able to do.

The failing of many product-definition processes, including VOC, is the myth that all customers are created equal, and that all customer inputs have equal weight. In many companies, a marketing department or sales department determines which customers are the ones deserving of your attention. And despite the frequent culture clashes that occur among engineering, marketing, and sales, the truth is that all three organizations need each other.

Some companies downplay the role of marketing in the product-definition process, while others recognize it for the valuable function that it can be. Even those companies that downplay its importance practice it religiously. One analog IC manufacturer has carefully chosen a group of customers it believes that it can profitably supply with circuits. It has then matched up a senior design engineer with each of these customers to learn what problems they are facing and try to figure out solutions together. Such client-based or partnership arrangements are becoming common in the industry, and represent one approach to the product definition process. If you listen carefully to what your customer is saying, you should be able to figure out what you can do for him. But the practitioners of this marketing approach will often downplay the importance of the marketing role—after all, you just need engineers talking to engineers to figure out what to do next, and it will all work out, right?

What these engineer-marketeers fail to realize is that someone picked out which customers they should get close to, and the marketing process began there, not at the product-definition step.

Whoever chooses the target customers has to think long and hard about several things. First, which companies buy enough of the sort of products we make to justify a lot of attention? Second, which of these companies are in solid financial shape? After all, a customer without money to pay the bills is probably not a customer to pursue too aggres-

sively. Third, which customers set the standard, or are considered by their peers to be market leaders? And finally, which ones do you want to bet on? And you need to keep reviewing these questions every couple of years, because the answers to all the foregoing questions change over time.

One of the classic mistakes in the customer-selection process involved a particular component manufacturer (we'll call it "Company X," since this particular story has been handed down in the industry folklore for so long that the original company name has probably been long since lost) that chose not to extend credit to a start-up computer company started by two young engineers in a garage in Silicon Valley. That company grew to become giant Apple Computer, and certain key components in their products are never supplied by Company X.

Step 3. Realization—Design and Build the Product

Assuming you have decided to move ahead and have the commitment of all the resources you need to get the project done, this is the part where you design the circuit and develop a product. Try several approaches. Don't force a known solution into this design if it doesn't fit. Also don't try to force an innovation where none is needed. Are you aiming for the Nobel Prize or a circuit that solves a customer's problem?

The process of fine-tuning a design includes learning to tell the difference between a good circuit and a bad circuit. In most instances, the difference is obvious. One works and meets specifications (including costs!), and the other does not. Case closed. But what about the case where both circuits meet spec?

At this point, lots of questions need to be objectively answered (and some may not yet have objective answers!). Does one circuit have advantages in your manufacturing process? How about your customer's manufacturing process? Does one circuit lend itself to further improvements as technology progresses? Does one circuit have a clear path that parallels the electronics industry's unrelenting goals of faster, cheaper, lower power, smaller, more efficient? Can someone copy it easily and rob you of the profits that are rightfully yours? And most important, will one enable more profit over the long term than the other? This last one is that hardest to answer, and is left as an exercise for the reader.

And it gets messier out there in the real world. Sometimes both designs "almost" meet spec. One meets everything except the speed, while the other meets every spec except the accuracy. Now what do you do?

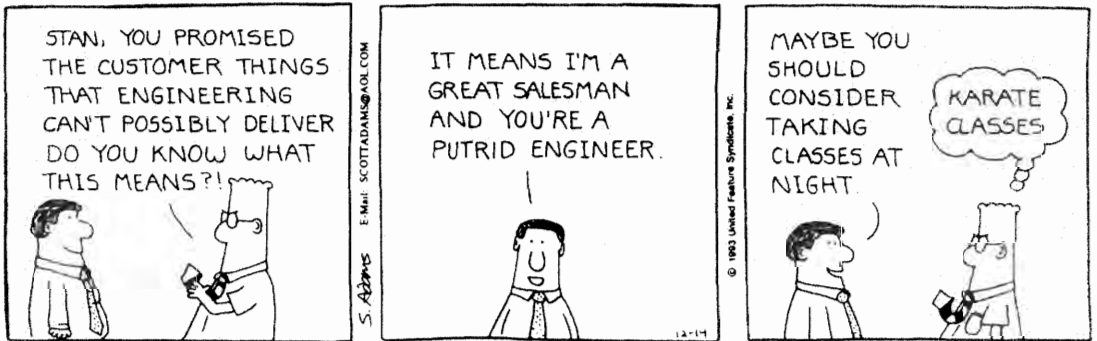
At this time, judgment separates the winners from the also-rans. This judgment must include common sense, experience of what has worked before and what has not, a real internal understanding of what the customer feels but is unable to express, and how the options compare with respect to all of this. Get opinions, facts, and make the call.

I won't comment too much on the actual circuit design process here. Skip to another chapter for details on how to design analog circuits. However, there is one design-related topic overlooked by many of the

chapter authors in this series. It has little to do with circuit success, but everything to do with product success. It is the schedule, every engineer's enemy.

When you know what the customer needs, you will also probably need to know when he needs it. This should have a major impact on the design approach you use. The first volume of this series suggested that in the case of designing a new IC, there are risks involved in new designs, new processes, and new packages. If you are designing to a tight schedule, you should probably not try to invent anything new. The more risks you take, the more likely it becomes that you will miss your customer's schedule. And if you miss his schedule, he will miss the schedule that his customer has given him. This means that everyone loses money.

Occasionally people in sales or marketing will make a promise to a customer relating to a schedule without consulting engineering. They are trying to keep the customer interested, and figure that if they get the order, they can apply enough management pressure to make the product-development process move faster than usual. I have also observed engineers making schedule commitments to customers that can't possibly be met ("Oh, yeah . . . I can get the new design done in a week or two . . . no problem"), ignoring the fact that the design phase of a product is usually not the most time-consuming part of the development process. One-sided commitments to customers will be a problem as long as enthusiasm and emotion get in the way of rational decision-making. Aside from increasing enrollment in karate classes, nobody wins.



DILBERT reprinted by permission of UFS, Inc.

It is also a good idea to qualify your customer by asking who his customers are. If there isn't a good answer, perhaps this isn't the right customer. Remember, a customer is someone who buys products from you and sends you money, not just someone who likes your ideas and thinks he might buy something someday. The latter is more of a prospect than a customer. I've been told that if you can't write down the phone numbers of three people who will buy your product, then you don't have a product. You should try this exercise on your customer, too. If your customer

has customers, try to talk to them. If you find out that all your customer's customers are planning to evaluate the new products at a particular industry event or trade show, you had better make sure that you have samples to your customer well in advance of the trade show so that he can assemble some prototypes to demonstrate. If he can't show units to his customers, both you and your customer may have to wait a year for the next show to launch a product.

While it sounds cold-hearted to focus only on customers, prospects are important, too. Never forget that. You should be responsive, courteous, and provide the support they need, and even a bit more. However, most prospects understand their place in the Grand Scheme of Things. Most of them will realize that their potential business may not represent your highest priority, and some will also become suspicious if you spend a lot of effort on their limited potential.

During the definition phase of a multi-channel D/A converter some years ago, I had determined that one potential market was numerically controlled machine tools and robots, since D/A converters are often used in position-control servomechanisms. Multi-axis motion controllers clearly needed multiple-channel D/A converters. I hit the books to find out the biggest manufacturers of machine tools and robots, and arranged a tour of them, focusing mostly on companies that were already customers of ours. The first few visits uncovered the sort of potential I had expected—on the order of a few hundred to a few thousand units each, with solid growth predicted for the next few years. Then I visited one company which was similar in size to the others (measured in annual sales), but which hadn't bought many chips from us in the previous year. In fact, they had only purchased small quantities of quite a few device types from us. This was puzzling, since they were housed in a very large building and had revenues comparable to the other companies. I presented the idea for the new product we were defining to the engineering staff. They listened attentively, made a few suggestions on certain specs that were important to them, and requested a few features. As I noted these inputs, I asked what their production volume was for the next year. They looked at each other, and after some discussion, determined that their production for the next year would be between 10 and 15 machines. I asked if they meant per week or per month, and they explained that the machines they made were very specialized, and sold into a price-insensitive market. Their production volume of 10 to 15 machines per year represented the same dollar volume as some of the other companies we had interviewed, since these machines were very big (which explained the huge building) and very expensive. They were grateful for the attention we had given them, and were happy to help us. They were also a bit surprised that we had chosen them as a target customer. However, their suggestions turned out to be useful in the product definition, and became strong selling points when we went back to the larger-volume customers.

By the time you have the first units, there are probably people waiting to try them. Some are inside your company (especially the people who

have to manufacture the product in volume); some are outside your company (customers). Presumably, there has been some effort expended to develop a way to evaluate the first units to see if they meet spec. Do it quickly, and as soon as you are satisfied that the units behave as you expect, get some in the hands of someone outside the company. Try to use someone who will tell other people if he likes it, and tell only you if he doesn't like it.

Very often, your interpretation of the customer's problem and his interpretation will still be different. The customer doesn't like your product. The reason is that you didn't meet the spec that was most important to him. Perhaps he didn't tell you clearly enough (or at all). Or else you didn't understand that it was so important. It doesn't matter where the fault lies—the customer is not happy because of a failure to communicate. This is inevitable. If people always communicated clearly, there would have been far fewer wars in human history. Misunderstandings have created much more important problems than anything that may occur between you and your customer (although it doesn't feel that way when it happens). Take a deep breath and try to work it out.

Situations like this call for diplomacy and tact far beyond anything taught in engineering school. I have observed the tendency for engineers to get defensive when a customer finds a flaw in their circuit, especially if it has met the internally defined specifications. "I did my part. If it isn't good enough for the customer, that's his problem" is a fairly ridiculous statement if you think about it in the context of a supplier-customer relationship. Similarly, the marketeer who says, "We did what they told us, now they should buy it," is also ignoring the obvious fact that he didn't really understand what the customer wanted. Remember—the customer has the final say. He has the money, and if you don't keep him happy, he'll send that money to someone else. If the product doesn't meet the critical spec, get back to work and fix it!

Another problem I have observed is the case where a design works "sometimes." This is worse than a circuit that doesn't work at all. Intermittent ability to deliver a product to a customer due to use of unqualified production processes, circuit blocks, packages, or whatever, will damage a customer relationship in more ways than you can imagine. In the old days, designers got one unit working, threw the finished documentation package over the wall to manufacturing, and went on to the next thing. That's not good enough. Manufacturing and Product Assurance must be routinely involved in the product-development process. They can offer valuable insight into mistakes that others have made, and help you avoid them. And while they may often ask lots of seemingly unrelated questions about a circuit during a design review, they are trying to help.

But having discussed what can go wrong, it is equally important to mention that usually it all comes together right. You give samples to the customer on the date you promised, he tries them, and calls back to say how much he likes them. His system works exactly as he had hoped, and he looks like a hero to his management. Then he shows his system to his

customers and they like it. His customers place orders, then he places orders for your product. Everyone smiles a lot.

Step 4. Introduction—Getting the Product to the Customers You Don't Know

If it's a product, there must be a customer. And at this point in the process you probably know some customers already—some are probably calling you for updates on the progress of your product, because they have decided to use it even before they have seen any units. In fact, if you don't have a first-name relationship with at least three potential customers, you ought to reconsider the whole thing. Occasionally, you'll be so far ahead of their thinking that your product will be exactly what they want even though they don't know it yet. There are some cases where this has happened—the personal computer, for example. But it happens so rarely that one of these per career is probably the limit. Without customers, you haven't designed a product—merely a circuit.

Giving a few samples to a potential customer is one way to introduce a product to the market. It works when there are only a few customers for a very specialized product. It's possible to know most of them. It gets more difficult when there are more potential customers than you can handle personally.

All customers will need help using your product. Some will need a little help, while others will need a lot of help. Still others will call you every day during the month they are designing a circuit around your product. Unless you have a lot of spare time available and need some new friends in your life, you have to create documentation adequate to allow them to use your marvelous widget without excessive hand-holding. Someone needs to write data sheets, instruction manuals, application notes, and troubleshooting guides. And that's not all. Unless you are personally going to be trained as a sales engineer, you will need to assume that other people with training in sales (yes, there is such a thing) will do the selling for you. If your product is going to be sold through a chain of distributors, you will need to provide sufficient training for them to understand your product's advantage over the competition (and how to handle situations where the competition is actually better in one respect or another). Unless you want every potential customer (or salesperson) to call you personally every time he has a question, you'll have to train other people to handle some of the questions in your place. This means time spent preparing and delivering training materials. It's difficult to fit this in while you're designing circuits.

Then there is the whole issue of external promotion to consider. There is a commonly held myth among both engineers and marketeers that derives from the "Build a better mousetrap and the world will beat a path to your door" axiom. It goes something like, "This product is so great it will sell itself." Too bad it isn't true. Here's what's wrong with that idea. The term "better" is completely subjective. If your customer hasn't been told why your product is better, he probably won't figure it out on his own.

He's probably too busy. You have to get the information to him somehow. Articles, seminars, trade shows, technical papers, newsletters—all of these are vehicles to get the information in front of the potential buyers' eyes. And all of these need careful planning and execution to optimize the return for each dollar spent. And of course, someone has to do the work.

Advertising is not as simple as it looks. A successful advertisement appears in the media that are read by the target customers, as determined both by examining the publishers' audit statements and observing what is on the desk of the customers you interview. Perhaps direct mail is a better choice. Perhaps your company has a complete suite of components for this problem—a “family” promotion of some kind may be in order. There are numerous vehicles available for product promotion—knowing them and choosing the most effective ones is the realm of the marketer.

The goal of product promotion is to generate leads, or names of people who are interested in possibly buying your product. There are other forms of promotion, of course, aimed at establishing or enhancing a company's image so that the product promotions will remain effective. But promotion does not automatically result in revenue. Poorly planned and executed promotion plans only waste money. But an effective promotion plan can work wonders.

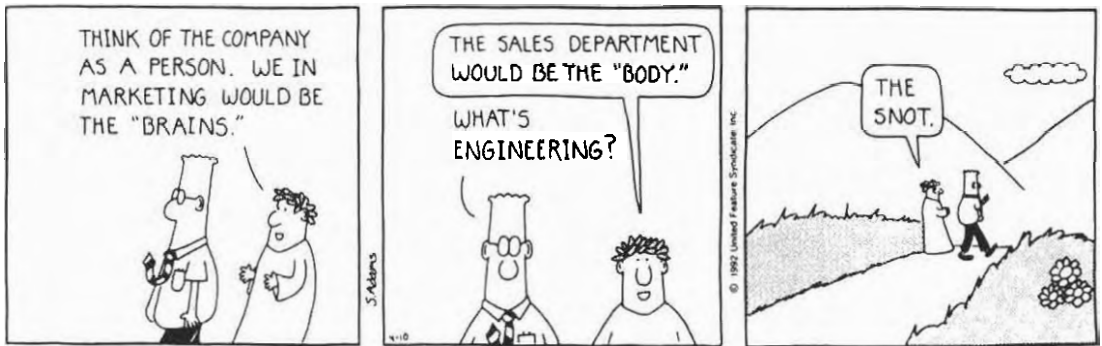
Even if your product is demonstrably better, the customer needs to know where the “door” to your company is located. Who does he call if he wants to buy the product? Does he know who your company is? Do the other people in his company know how to do business with your company? And lastly, if the manufacturer of the second- or even third-best mousetrap has a sales force that beats a path to your potential customers' doors, the world will have no reason to beat a path to your door, and you will not succeed. Having the world's best product simply isn't enough.

Yes, you need salespeople. Most engineers do not like salespeople. Many engineers consider circuit design a Higher Calling of some sort, and have little interest in the human interactions that enable the exchange of goods and services in a market economy. However, without these interactions, little commerce could take place.

Being on the losing end of a potential order due to a lack of relationship is frustrating. I recall one incident where after investing many months of effort, including several long-haul airplane flights, we lost a very big order at Customer A to Competitor X. I knew our product was better. I knew our price was better. I knew the overall solution cost was better. The overall system performance with our product was better. Yet we lost the order. We were all at a loss to understand why we lost the order. We had done everything right, by any measure. It took a while, but finally one of the Customer A designers told me what had happened. The order we were seeking was even more important to Competitor X than it was to us. The sales manager of X raised visibility of the impending lost business to the president of the company. The Competitor X president then phoned his old friend who was president of Customer A, and made an appointment to

play golf the next weekend. Somewhere on the back nine, the issue of the new project was raised, and Mr. X asked his old friend if there was any way to use his company's product in the new project. He had heard some disturbing things about possibly losing the order. The next day, Mr. A called his engineering and purchasing managers and instructed them to use the Competitor X product. They saluted smartly, and followed orders. In this case (and there have been numerous others over the years) the human relationship outweighed the objective and fact-based decision-making processes. Losing business this way is frustrating. Getting to the point where you can win this way takes a long time, a solid track record of success, and a good sales force.

It is worth noting that most salespeople have a pretty low opinion of engineers as well. They see most engineers as unable to see the obvious importance of their customer, and can't understand why it's hard to improve the performance of a circuit by a mere factor of two by just making a minor adjustment that should take no time and entail no risk.



DILBERT reprinted by permission of UFS, Inc

After introduction, someone must consider the management of the life-cycle of the product. Periodically reviewing the performance of a product against measurable data (sales, profits, units sold, etc.) is a necessary evil, and generally unrelated to circuit design. Long after a product has been introduced, someone (variously called a product manager, marketing engineer, or merchandising specialist, depending on the company's culture) reviews all these (and other) metrics, analyzes the cause, and undertakes corrective actions as necessary. If sales have declined, it may be that price has eroded due to new competition, a major program has ended, or some other phenomenon. It is unlikely that the manufacturing or accounting operations of your company will have visibility into the end customers, and they can only build product and report data. Someone who can examine the data and determine which course of action leads to the maximum revenue and profits must make the decisions regarding the product.

Of course, you may want to do much of this yourself. And that's fine, as long as you recognize that you will have less and less time available to design circuits. Or to learn about other kinds of circuits and systems. Consider such a decision very carefully.

Step 5. Closure—Move On to the Next Problem

While it is important to deliver circuit designs that meet certain specifications, it is not advisable to succeed once, and then rely on incremental improvements on the same idea from time to time for the rest of a career. Once you have completed the process of solving a customer's problem, it's time to declare victory and move on. Document what you did, make sure that the solution is on "autopilot," train others to understand the issues and trade-offs made, and then walk away.

You need to do new things from time to time to avoid getting stale. In the area of circuit design, doing the same things the same old way and just waiting for incremental improvements (new processes or components) can type-cast an engineer and limit his professional growth. If that's your choice, make sure you understand its implications. Most engineers I have known have looked for new ways to do things, and often find old tricks useful in solving new problems.

But where do you find new problems to solve? There are several sources of inspiration for what to do next. The best (and sadly, the most often overlooked) source of ideas for new products is your current customers. Remember the customer you designed a low-noise amplifier for last year? Perhaps he also needs a high-resolution A/D converter. Or the guy who needed a battery monitor—he might need something else vaguely analog in nature. Talk to him. But do a bit of homework yourself—find out what projects your company already offers so you don't spend a lot of time identifying a problem that others in your company are already solving. As companies grow, it becomes difficult to know what is going on in other parts of the organization; this is another place where a salesperson can be useful. He is expected to know what products his company has available now and in development to solve some customer's problem. If he hears his own customer express a similar need, he can then bring in the resources he needs to find the best available solution for his customer's problem.

I recall one visit to a customer where I had one of our design engineers with me. The customer was having a minor problem with one of our D/A converters, but had solved it by the time we got there. However, since he had the "factory guys" there, he wanted to tell us about another problem that he couldn't solve. Ever the eager marketing type, I asked him to tell me more—if my own group didn't have the solution, I could carry the message to the relevant group and get him the help he needed. The customer then launched into a lengthy dissertation on what was wrong with a particular class of IC that didn't work quite right—it was something that connected to a D/A converter, so I was curious. About five minutes into the interview, my colleague interrupted the customer to

inform him that we didn't make that kind of device, and we couldn't help him. He didn't want the customer to waste his time explaining a problem we couldn't solve.

As it happened, however, another part of our company was in fact in the final design stages of a chip that was very well suited to solving the customer's problem. I had to play the diplomat and remind my colleague about the device under "secret development," and encourage the customer to keep talking. I took lots of notes, forwarded them to the appropriate group in my company, and we eventually did some very nice business with that customer.

Engineers working in high technology need to keep abreast of the latest research in their field, including new technologies. Many analog circuit designers look with disdain upon digital design; however, there are powerful techniques available in the digital domain that have performance and cost advantages over any attempt to duplicate them in the analog domain. Knowing something about them can help broaden your range of available trade-offs.

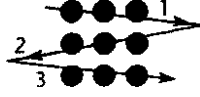
Read the journals; attend a conference or two each year, including one intended for your customers. Talk to people, especially others in your company who deal with a lot of customers. Buy things and take them apart to see how they work. Find out who is trying to solve similar problems to yours, perhaps in a different end application. The ideas you encounter may someday be useful. Learning is almost never in vain—an idea whose present worth is questionable sometimes becomes a solution to a problem in the future. And solving problems profitably is quite satisfying indeed.

And here are the solutions to the "connect-the-dots" problem . . .
"draw three parallel rows of three dots each on a piece of paper; connect all nine dots by drawing four straight lines, never lifting the writing implement from the paper."

Analog Circuit Design for Fun and Profit

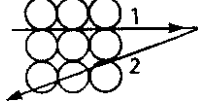
B. With three lines

Solution for 3 lines:



C. With two . . .

Solution for 2 lines:



D. And finally, with one line . . .

Solution for 1 line:

