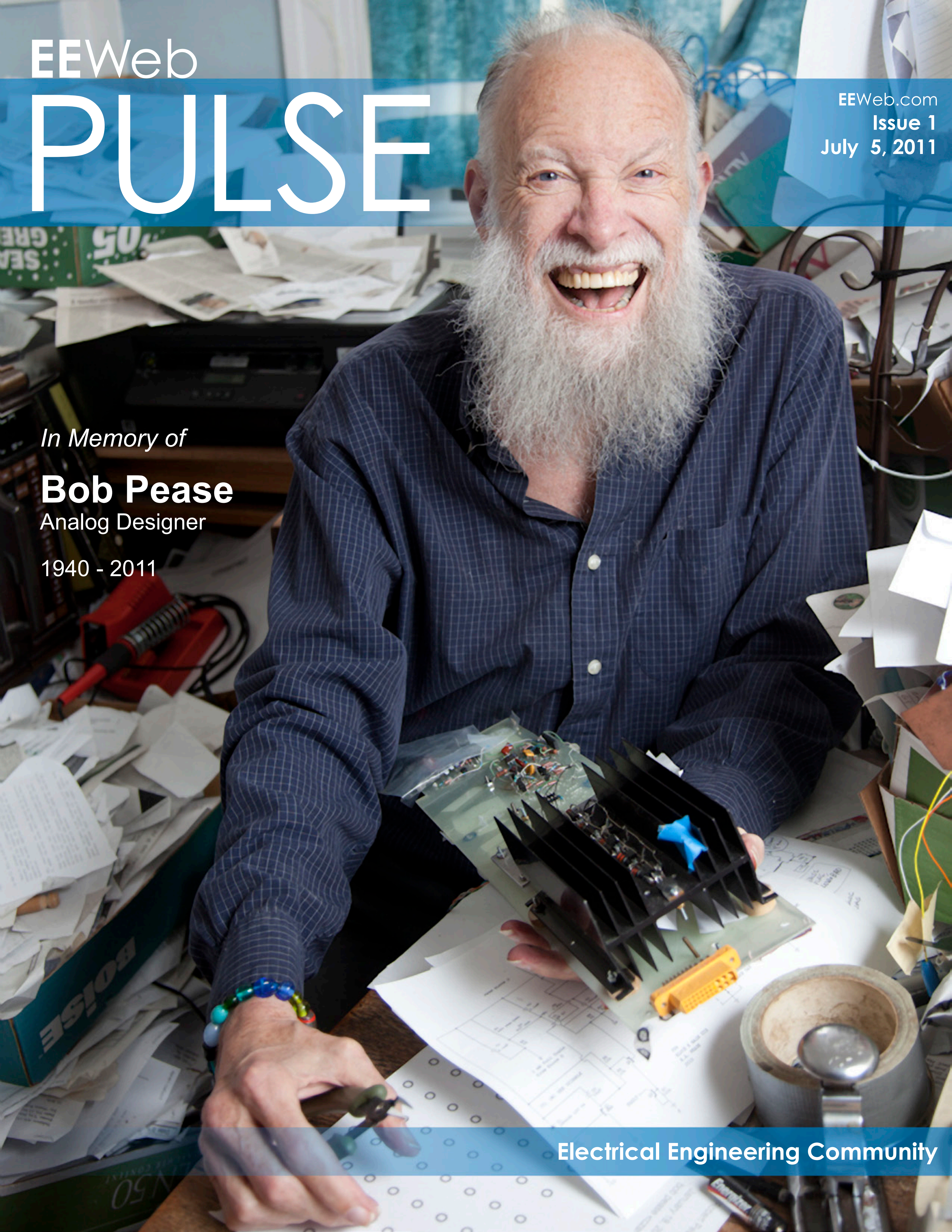
EEWeb
PULSE
EEWeb.com
Issue 1
July 5, 2011



In Memory of

Bob Pease
Analog Designer

1940 - 2011

Electrical Engineering Community

TABLE OF CONTENTS

Remembering Bob Pease

Analog Designer, 1940-2011

Bob Pease was killed in a tragic car accident on June 18 after leaving a memorial for Jim Williams, a colleague and fellow analog circuit designer. Bob devoted so much of his life to helping his fellow engineer, and his untimely death deeply affected the electrical engineering community.

We had the privilege to speak with some of Bob's friends, colleagues, and associates who kindly took a moment to share their memories of Bob.

Wanda Garrett, National Semiconductor

"I first met Bob when I was interviewing for an application engineer position about 28 years ago. I did not interview with him, but when lunch time came around, I went to eat with him and my hiring manager, Al Kelsch. Although I had just met them both, I could tell right away that these two individuals had very different styles. I could tell that Bob was a rather rambunctious, outgoing guy, and Al was a very straight-laced, conservative guy. For a company to have these two completely different people working together that seemed to get along so well, I figured this would be a good place to come. So when I was given the offer to join, I jumped at it because I thought it was going to be good, and I was going to get the chance to work with both Al and Bob.

When we went out, Bob was going to drive in his Volkswagen. The first thing he did was open up the door and clear out a spot in the back seat for Al to sit. I piled into the front seat, and as we were getting ready to leave, Bob said, "Hey, do you like to bake bread?" And I said, "Sure." So out from under his driver's seat he pulled the recipe for bread. I thought that was kind of cool."

"There were sort of two aspects of Bob's style as a designer. One aspect was that he emphasized first thinking about the problem and figuring out what the answer should kind of look like, so he could recognize it when he saw it. This also allowed him the ability to detect when he was wrong. As an example, he and I did a joint paper a number of years ago for a power supply conference. The topic was related to the tools that we offered to go along with our SIMPLE SWITCHER regulators. I was talking about the tools, and he was talking about the use of the tools and how to get good results. He was emphasizing thinking, understanding the design, and doing bench validation so that users don't get fooled by possible mistakes. More simply, he would kind of make an assumption, and test it to see if he was correct. But it was necessary for him to kind of know the magnitude of the answer he was aiming for so he could interpret whether he was looking at the right answer or not.

He had another approach also, which was to be very collaborative. If he was working on a problem, or he had a design that he wanted to make sure was sound, he would call together a bunch of people. This included reviewers, friends, and anyone else that he figured would have a helpful opinion. Sometimes he would call a "beer check." In other words, if someone found an error, he or she earned a beer. If he was writing an article, he would distribute it to a panel of reviewers, and we would all get a chance to chime in and say what we thought was right, wrong, good, or bad. His panel probably had 30 or 40 reviewers for most of his columns and books."

Continued: Wanda Garrett

"If Bob felt someone deserved his respect, he made his respect evident. He didn't just respect anyone because of his or her job title, which probably annoyed some of the supervisors. If he saw that a person was sensible, that he or she could think and contribute, then that person got his respect. What mattered to him was the ability to think and analyze."

"One thing that most people might not know about Bob is his interest in music. He really enjoyed a lot of obscure music from the 40s, 50s, and 60s. He had a very nice, high tenor singing voice; he really enjoyed singing. One of the reasons he was so excited to move to San Francisco was that he could sing in a large cathedral there.

He was also part of the singing club at National, and one of the songs that he had a lot of fun with was kind of a novelty song called "The Green-Eyed Dragon with the Thirteen Tails." He and I worked that one up and that was one of his signature songs. He sang it while I played the piano."

"He used to stash a variety of signs under his driver's seat. This was so when he would see another driver with, for example, a broken tail light, he would be able to communicate the issue to that person using a designated sign."

Ken Baker, Analog Devices

"I had been corresponding with Bob in emails regarding various applications before I met him. When I was finally introduced to him, I shook his hand and said, "Bob, you know I've been corresponding with you over the past couple of years, and now that I've met you, I like you in spite of yourself." He threw back his head and started to laugh. And he said, "You know Ken, I like you too." And we were friends from there on in whenever I would visit the west coast to meet with him. "

"Bob and I used to just go to lunch, chew the fat about things in general like Analog Devices, National, his trips, and of course his clunky old Volkswagen."

"Bob had one of the desks like mine—piles of junk all over the place! But he still knew where everything was. That's how Bob was; what looked like chaos to an outsider was Bob's own special form of organization."

"We used to talk about SPICE and how we hated it, because some engineers lived with SPICE. And he told them, "Solder up the circuit and run it that way. That's the real way because SPICE is just an emulation. And I used to tell Bob, "Tell them to solder their fingers together."

Paul Grohe, National Semiconductor

"Back in 1990 I was going to the College of San Mateo, and I was recruited to come and interview with Bob for a job at National. I came into the interview kind of reluctantly because I was actually sick that day. I came in, went through the interview process, and talked to Bob, who was very reassuring, telling me, "Don't worry, don't worry." Days later I came home and my mom said, "This really strange guy stopped by to drop off this letter. He was driving a Volkswagen that looked like a dinosaur." In the letter was an offer to work as Bob's technician. The fact that he was willing to personally deliver the job offer meant a lot to me, and is a testament to the type of person he was."

"He had an office, a couple lab benches, and at one point, before he retired, he had basically an entire room because the mess was getting too large, and encroaching on others' benches. On his bench were a Kepco power supply, a Tektronix 475 oscilloscope, usually a Wavetek 176 function generator, and piles of Fluke DMMs."

"Bob would have many ideas for me. He would come up with a concept, scribble out a circuit, and give it to me to build. I would either do an air-wire circuit like he would do, or I would actually do a neat one. I would then tweak it, and he would come by while I was doing the measurements. Bob couldn't do this because, while he was talking on the phone he couldn't build circuits. So I was doing his leg-work at the time."

"He was a master with a soldering iron. On the cover of his troubleshooting book, there is a board with all of the wires floating up in the air. It's for the LM331 voltage-to-

Continued: Paul Grohe

frequency converter. That is how he did it. Every trace was a resistor, and between traces was a capacitor, built up in the air like that. Everything was built on copper-clad and wire-wrap sockets."

"Bob loved LF411s. They were his favorite part. He also loved 2N3904s and 2N3906s. He would go through buckets of those parts. He always kept it very simple as far as his parts selection; he always knew how to push the parts."

"There is no way to dissect the way he would think. He was just so intimate with the knowledge of the circuits, and had seen and done it all. You could draw a circuit in any direction and Bob could figure it out."

"My favorite expression of his was, *Show me where it says I can't do it.*"

Bob loved LF411s. They were his favorite part. He also loved 2N3904s and 2N3906s. He would go through buckets of those parts.

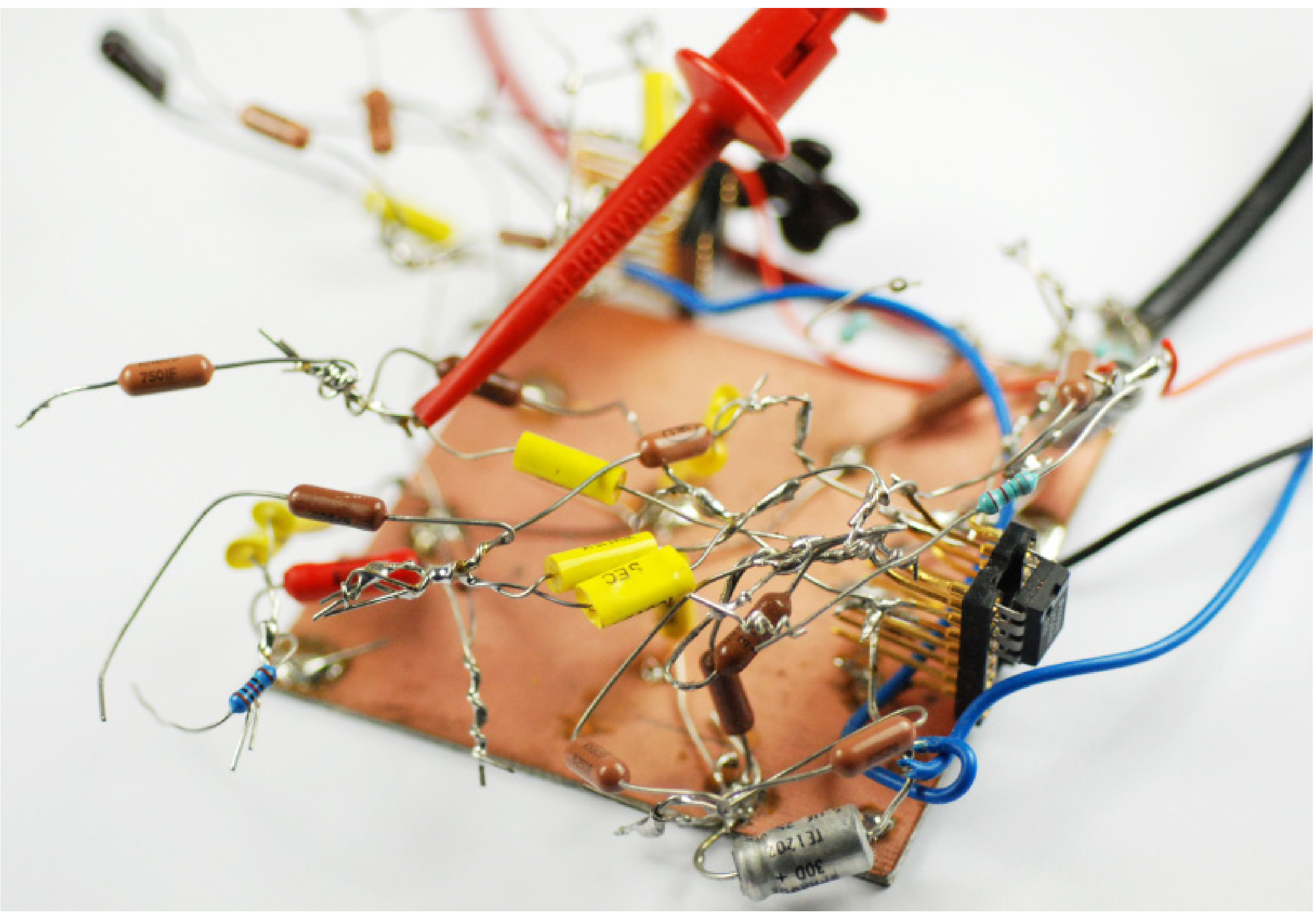


Image: One of Bob Pease's circuits, courtesy National Semiconductor.

Robert Pease

Analog Designer

This interview was taken on April 26th of this year.

How did you get into electronics/engineering and when did you start?

I was 17 years old when I started. I built a Knight-kit 10-watt audio amplifier. When I was a kid I was not into car radios or things like that. I was not one of those kids who took apart my father's car radio and put it back together and did it ten more times. I never did that. It is very simple. I came in late.

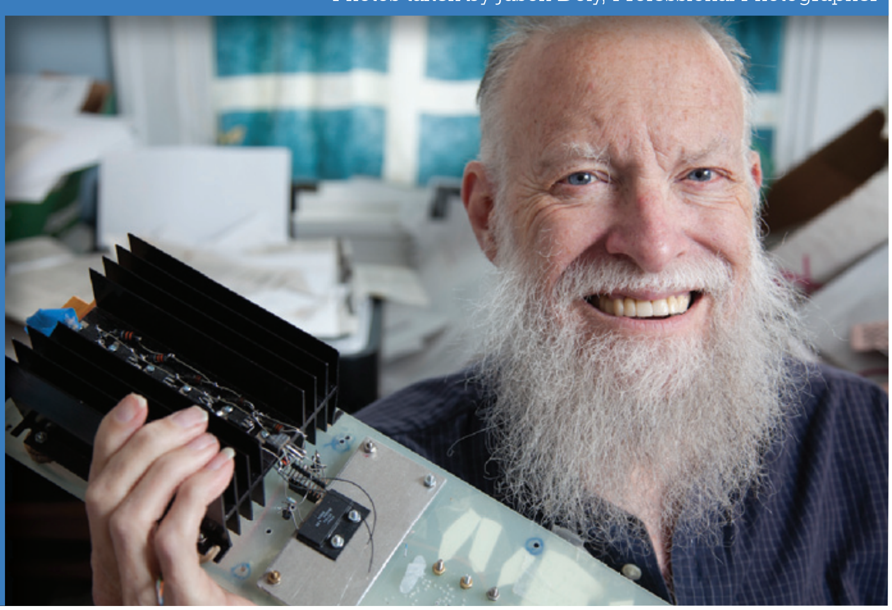
I was good at physics and math. In my senior year at MIT, I was taking some physics courses, I think it was course #8.721 and I said, "This stuff isn't physical anymore." You know, quantum theory and eigen vectors. And then I thought maybe I should get into electronics, and I did, and it was fun! I had a good piece of luck. I think it was my sophomore year at MIT, with Len Kleinrock, who has recently been properly hailed, 50 years late, as one of the inventors of the internet. He was one of the better teachers I've had and I really enjoyed learning about electronics design.

Well, I am still doing design with Piecewise Linear Electronic Circuits and transistors. So that's how I snuck up on electronics.

After MIT, where did you first go to work?

Even before I left MIT, I went over to Philbrick Researches and I helped do some technical writing. I may not have been the best engineer at that time because I would have been as green as you can get, but I helped them produce some data sheets, which then got into more electronics and transistor design.

Photos taken by Jason Doiy, Professional Photographer



Before you got into designing ICs, what were you doing?

I was doing board-level circuit design, and that was a lot of fun because we could make money out of it. One of my buddies invented the P2, which was about 3" x 1.5" x 1". We were selling that thing that used about enough parts to make a seven-transistor AM radio, for $220. That was a lot of money back in 1961. So we figured out how to do things that other people were not able to do. We packaged them nicely, wrote good application notes and there you are. That was the Philbrick legacy.

You can still find the Philbrick Applications Manual. Analog Devices was nice enough to publish it as a service to people who are trying to remember the "good old days" of engineering design.

Generally, I do not do much digital. I know how to use a D-flip flop and gates, but I just do not do that much digital design. That is its own specialty. I have never made a mistake writing software, because I do not write software.

Do you have any favorite circuits?

For many years one of my favorite circuits was the Philbrick 4701 voltage-to-frequency converter and it was 0.1 percent linear with a nice safety margin. It was revolutionary for its day. We would get out there and make several million dollars of revenue per year based on the voltage-to-frequency converter. The patent number is 3,746,968.

This circuit got patented, though the patent has long since expired, but it used a cheap LM301 and a few diodes to make a 0.1 percent linear converter. And if I went back in a time machine to 1940 I could build a three-digit DVM to help the war effort, except for two things: I could not solder, and I was one year old. I would have had to show my mother how to solder and I did not know how to do that either.

Anyhow, some of these circuits are, or have been revolutionary, and some have been evolutionary. We keep having fun inventing circuits.

After working with George A. Philbrick Researches what did you do?

I worked there for 15 years, then I moved to Silicon Valley and have had a lot of fun ever since! I worked for National Semiconductor for 34 years, starting in 1976.

Can you tell us a little more about your IC Design work?

I have designed about 27 linear ICs. I started at Philbrick and designed a few ICs there. When I came to National they said, "Here is a linear data book and you have to be responsible for all the circuits for applications engineering."And I said, "Good, wonderful." I learned everything I could about linear ICs and I started to work on the corners of projects to do linear IC design. I helped out and got better at learning and found a couple of disaster areas, too, which I was able to fix. After that I made some more designs and got more into writing columns for Electronic Design Magazine.

What has been your favorite project?

The 4701 was my favorite because it led to not just a 10 kilohertz voltage-frequency converter, but 100 kHz and then a megahertz, and it was ultra linear. That was one of my better projects and it was over 40 years ago. But I worked on a lot of other projects that I truly enjoyed.

> I thought maybe I should get into electronics, and I did, and it was fun!

What are you currently working on?

I am doing some work, some consulting, and some contracting. I work on high precision current sources—high power. I have been having fun recently with audio amplifiers that are more linear than 0.1 parts per million. That is a little challenging. I can see below 0.1 parts per million. In fact, I can see below 0.02. I did quite well with that. I use a scope for the set up when testing it. I can use either Agilent or Tektronix but I need to use an old-fashioned analog scope because I do not think most digital scopes will let you do what I am doing. With a digital scope, there are certain things you cannot do. You cannot be certain you saw something. AN-1485 is a 20-page application note on how to measure below one part per million. Lots of op amps are below a part per million, but when you get below that, you cannot just use an Audio Precision because it stops at about three parts per million. I figured out how to get it below .01 parts per million but people cannot normally hear that. I did a good three-hour lecture at the Audio Engineering Society (AES) in San Francisco four years ago about audio. There are things you cannot hear that I can measure and there are things that you can hear that I cannot measure. We had a lot of fun with that.

What challenges do you foresee in our industry?

Well I see a lot of digital stuff that is insane, and I have no idea. There is still some analog stuff that can be pretty good. You can have 100,000 engineers graduating in China that have no idea of a creative process or critical thinking, so I would like to think that we are pretty good in this area, but we cannot lead forever.

Storm Peterson said,

"Predicting is very hard – especially about the future."

I am not good at foreseeing the future.

Putting the R in RTL:

Ray Salemi is a veteran of the EDA industry and has been working with Hardware Description Languages since he joined Gateway Design Automation—the company that invented Verilog. Over the course of his career he has worked at Cadence, Sun Microsystems, and Mentor Graphics. Ray is currently an Applications Engineer Consultant with Mentor Graphics.

Ray Salemi
Application Engineer

Many years ago, when I was a young man and the Boston Red Sox had just lost the 1986 world series, controversy stalked the land of hardward development. A new technology called Register Transfer Language or RTL threatened to replace the time honored tradition of placing gates on schematics and connecting them by hand. (This technology had replaced drawing gates on paper and connecting them by hand.)

Those who advocated RTL predicted a new day, in which it would be easy to generate designs of 50,000 gates or more, while those who wanted to remain with schematics claimed that they could easily design more efficiently and keep up with the RTL designers. But soon, as ASICs topped hundreds of thousands of gates and then a million gates, the schematic designers were left in the dust.

Fifteen years later the schematic vs RTL controversy flared again in the world of FPGA. As I listened to engineers argue the merits of RTL vs schematic capture, I could swear that I had been sucked into a wormhole and dropped back into the 1980s. Of course, I knew how the RTL vs. schematic argument would end having seen it all before. RTL won before and has won again.

This column is devoted to RTL design. Each month, it will discuss another wrinkle in the seemingly simple design cycle of writing RTL, simulating it, and running it through a synthesis tool. We'll look at how synthesis tools react to RTL and how to write test benches to make sure the RTL is doing what you want it to do.

This month we are going to kick off our set of columns by discussiong the most basic of Register Transfer Language actions: transfering data into a register. We're going to look at the coding styles you use in order to get your EDA tools such as synthesis and simulation to recognize what you are trying to do.

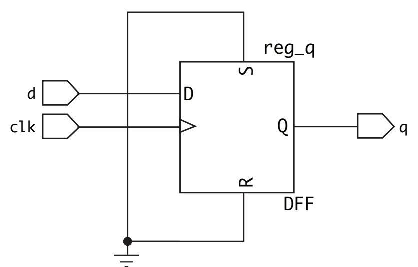
One might think that EDA tools should be able to take any form of RTL code and figure out how to make logic that will implement it. For the most part, one would be right, because as long as you write something that can, in some way, be construed as describing hardware, you will get results out of an EDA tool. However, if your code doesn't follow the industry's standard RTL coding styles, you may get different kind of hardware depending upon the synthesis tool you use. What's worse, your simulation tool may interpret the hardware differently than the synthesis tool, and the you'll have simulation/synthesis mismatches.

We're going to look at how you code four basic styles of flip-flops, and we'll use this coding style as the basis for future articles on synthesis and simulation.

The Basic Flip Flip

The basic flip flop is a flip flop with clock, an input, and an output. When we see a positive edge on the clock, the input signal gets flopped into the output. We can see how to code this flip flip in Figure 1.

```
15 ff_proc: process (clk)
16 begin
17   if (clk = '1' and clk'event) then
18     q <= d;
19   end if;
20 end process ff_proc;
```




```
module ff (output reg q, input d,
               input clk);

  always @(posedge clk)
   q <=d;

end module
```

Figure 1

The code at the top of this example is a VHDL process that describes this flip flop, and the code at the bottom is a complete SystemVerilog module that does the same thing (SystemVerilog has replaced Verilog in the industry).

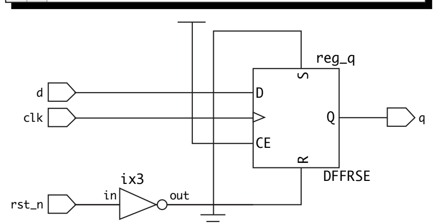
Both code snippets contain a process which is sensitive to the clock. The VHDL process is sensitive to all changes on the clock, and we check for the rising edge on line 17. The Verilog always block is only sensitive to the rising edge of the clock.

In both cases, when the code sees a rising edge it places the value of d into q and we have a flop. We can see what this flop looks like in the picture in the middle. This picture was generated using Mentor Graphic's synthesis tool, Precision. It shows clk, d, and q. The set and reset signals are grounded.

The Basic Flop with a Synchronous Reset

Of course, you would never want to use a flop such as the one in Figure 1 because the lack of a reset will create problems when you try to simulate your design. Without the reset, you'll have X's in your simulation. These may go away, but they will cause mismatches if you try to make your RTL simulation match your gate level simulation.

```
ff_proc: process (clk)
begin
  if (clk = '1' and clk'event) then
    if (rst_n = '0') then
      q <= '0';
    else
      q <= d;
    end if;
  end if;
end process ff_proc;
```




```
  always @(posedge clk)
   if (!rst_n)
     q <= 0;
   else
     q <= d;
```

Figure 2

We can write code so that the synthesis tool will take advantage of the reset pin that exists on all flip flop models (see Figure 2). In Figure 2, we've added a reset signal to the inputs of the flop called rst_n.

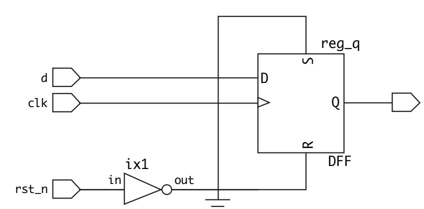
Once our process is activated by the positive edge of the clock, we check rst_n and if it is 0, we store a zero in the q output. Notice that we do not attach the rst_n signal to q. While this may work properly with some synthesis tools, others will get confused to see a reset signal being used for data.

Notice that type of flop that was instantiated by the synthsis tool: DFFRSE. This is a flop with a synchronous reset signal.

The Basic Flop with an Asynchronous Reset

If we want our flop to reset immediately when rst_n goes to 0, we need to implement an asychronous reset. We do this by adding the rst_n signal to the sensitivity list of our process. Now the process will execute when the reset signal changes value and we can check for the reset. We can see the asynchronous reset implemented in figure 3.

```
ff_proc: process (clk, rst_n)
begin
  if (rst_n = '0') then
    q <= '0';
  elsif (clk = '1' and clk'event) then
    q <= d;
  end if;
end process ff_proc;
```




```
module ff (output reg q, input d,
               input rst_n, input clk);

  always @(posedge clk or negedge rst_n)
   if (!rst_n)
     q <= 0;
   else
     q <=d;

end module
```

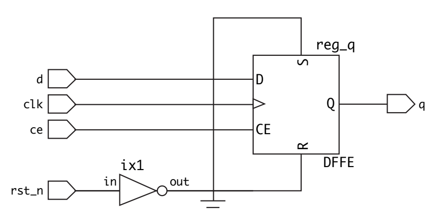
Figure 3

Notice that the VHDL is sensitive to any change to rst_n and checks to see if the rst_n signal is 0 before it transfer the data. The Verilog code is only sensitive to the negative edge of reset. If we allowed it to be sensitive to both edges of rst_n, then the reset could act like a clock on its positive edge. We don't want that.

The Basic Flop with Asynchronous Reset and a Clock Enable

The final pattern we'll examine is a flop with a clock enable. We can see the implementation in Figure 4. We've added another signal to the input ports, and we check the clock enable just before we transfer the data from d to q.

```
ff_proc: process (clk, rst_n)
begin
  if (rst_n = '0') then
    q <= '0';
  elsif (clk = '1' and clk'event) then
    if (ce = '1') then
      q <= d;
    end if;
  end if;
end process ff_proc;
```




```
module ff (output reg q, input d,
           input rst_n, input ce,
           input clk);

  always @(posedge clk or negedge rst_n)
  if (!rst_n)
    q <= 0;
  else
    if (ce) q <=d;

end module
```

Figure 4

Notice that Precision recognized the pattern for the clock enable and attached the signal to the previously unused CE port on the flop model. Now we'll use whatever clock enable scheme our FPGA technology implements to have a clock enable.

Summary

In this article, we discussed the fact that synthesis and simualtion tools are happiest when we use recognized coding patterns. Then we looked at the coding patterns that define four common flip flops. In next month's article we'll examine ways of creating combinatorial logic in RTL and some of the ways that our combinatorial code can create unexpected latches.

Selecting Passive Components with a Buck Converter

Tamara Schmitz
Intersil Corporation

Often, power design is the last consideration in a system. Then, most users want a box that just works—taking in one DC voltage and producing another. This box can take various forms. It can be a step-down to generate a lower voltage or step-up to generate a higher voltage. There are also plenty of special options, like step-up/down, flyback and single-ended primary inductor converter (sepic), which is a DC-DC converter that allows the output voltage to be greater than, less than, or equal to the input voltage. For a system that will run on AC power, the first AC-to-DC block will probably create the highest DC voltage level needed by the system. Therefore, the most widely used devices are step-down converters, also called buck converters. Here we will start with a basic step-down voltage converter selected to help light load efficiency and then discuss the consideration for selecting the surrounding components.

The two main types of step-down converters are Low Drop-Out (LDOs) and switching regulators. LDOs give clear and stable voltages while switching regulators are optimizing for more efficient operation. High efficiency means less energy is lost in the conversion, simplifying thermal management. The more current that needs to be produced means a bigger system which generates more heat. Since switching regulators have higher efficiency and are the most prevalent solution, we are focusing on them here. Furthermore, to simplify the discussion, we'll limit ourselves to buck (step-down) converters for simplicity. Figure 1 shows the basics of a type of step-down switching regulator, a synchronous buck converter. The term synchronous buck indicates that a MOSFET is used as the lower switch (labeled in Figure 1 with Ilower going through it.) Comparatively, a standard buck regulator has a Schottky diode as a lower switch. The main benefit of a synchronous buck regulator compared with a standard buck regulator is better efficiency due to a lower voltage drop of the MOSFET versus a diode.

The timing information for the lower and upper MOSFETs is provided by a pulse-width modulation (PWM) controller. There is only 1 input shown in Figure 1 to the PWM while in many schematics there are two inputs to the PWM. The second input voltage to the circuit is the supply voltage of the PWM. The drawn input to the controller is a voltage fed back from the output. This loop allows the buck converter to regulate its output in response to load changes. The output of the PWM block is a digital signal toggling up and down at the switching frequency. It turns on only one MOSFET at a time. Allowing both MOSFETs to be on simultaneously would cause a short from Vin to GND, which would destroy efficiency and is not advised. The duty cycle of this signal determines the percentage of the time that the input is directly connected to the output. Thus, the output voltage is the product of input voltage and this duty cycle.

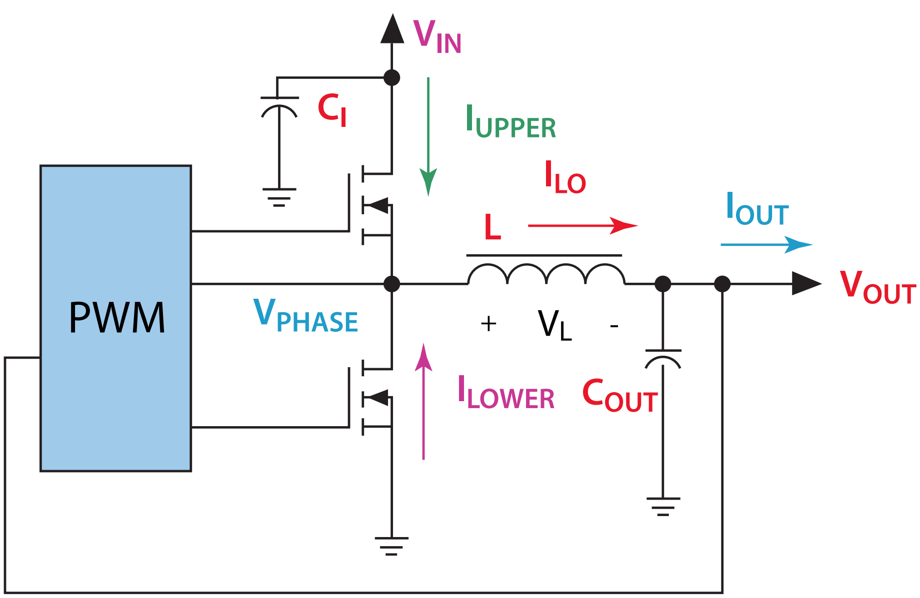


Figure 1: *Basics of a Synchronous Buck Converter*

Choosing the IC

The control loop noted above allows the buck converter to maintain a steady output voltage. That loop can be implemented in a number of ways. The simplest converters use either voltage or current feedback. These converters are rugged, straightforward and cost-effective. As buck converters began to be used in a variety of applications, a weakness was found. Consider the power circuitry for a graphics card. As the video content changes, so does the load on the buck converter. The system can handle a wide range of changes, but the efficiency rapidly degrades for light load conditions (when only a little current is needed.) If efficiency is a concern, it's time for a better buck converter solution.

One such improvement is called hysteretic control. An example is the Intersil ISL62871. The efficiency versus load is presented in Figure 2. These converters are designed for worst-case conditions, so light load is not a permanent situation. These DC:DC converters are better at coping with changes in load variations without drastically affecting system efficiency. Figure 2 shows the efficiency of the ISL62871 measured with different output currents. This variance in output current shows the performance for different loads.

Choosing the Switching Frequency

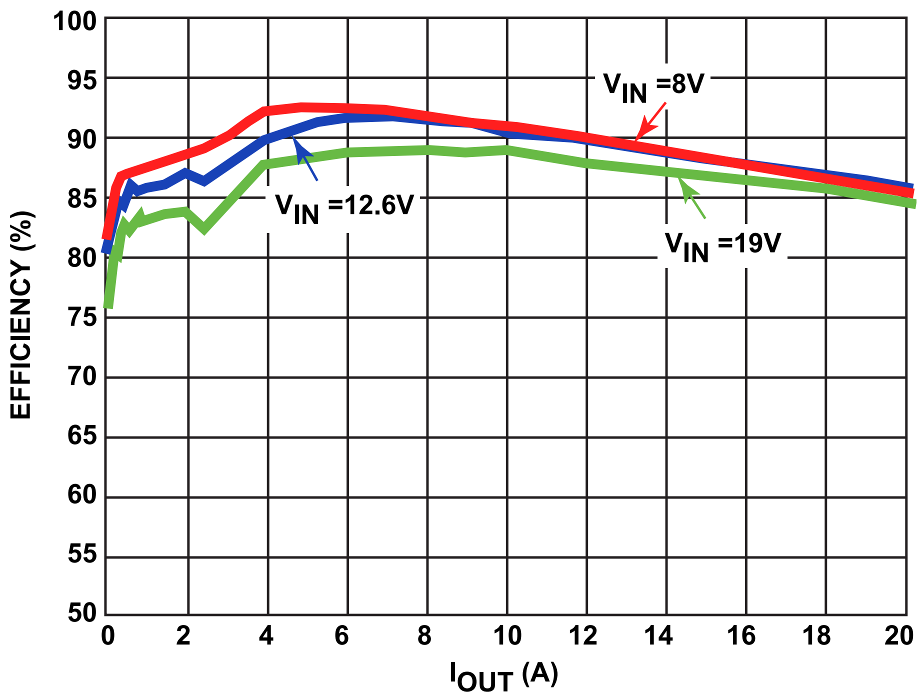
Although switching frequency is sometimes fixed for a device, it is still worth discussing it. The chief trade-off is efficiency. In the simplest terms, the MOSFETs have a certain turn-on and turn-off time. As the frequency increases, the transitional time increases as a percentage of the total period. The result—the efficiency is reduced. So if efficiency is the most important design goal, consider lowering the switching frequency. If the efficiency of the system is adequate, then you might be able to allow a higher switching frequency. The higher frequency will allow the use of smaller external passive components, namely the output inductor and capacitors, to reduce your system size.

External Components

While you could be ambitious and try to design a fully discrete solution of about 40 components – that requires significant additional effort. Alternatively, let's examine the external components in Figure 1 and the parasitics they play in the performance of the system.

The five additional components we must select are the input capacitance, the output capacitance, the output inductor, and the upper and lower MOSFETs. The output inductor is selected to meet the output ripple requirements and to minimize the PWM's response time to a changing load. The lower bound of possible inductor values is set by the ripple requirement. Before running out to find the smallest (and possibly cheapest) inductor that will work for you, remember that they are not ideal. Real inductors have a saturation level. That saturation level must be higher than the peak current in the system to create a successful design. Experienced designers also know that the inductance isn't constant versus current. In fact, the value of inductance drops as you pull more current through the component. Check the inductor datasheet to ensure that your chosen value can handle the peak current in your system. It seems that erring on the larger side might be the best inductance choice. There is a balance, though. Larger values of inductance do reduce output ripple, but they will also limit the slew rate. Too large an inductance value will limit response time to

Figure 2: *Efficiency versus Load for the Intersil ISL62817 with Vout=1.1V*



a load transient. So, in selecting the inductor, there is a clear trade-off between a quieter output due to lower peak-to-peak ripple or needing the system to respond quickly to a load change.

The input capacitance, C1, is responsible for sourcing the AC component of the input current flowing into the upper MOSFET. Therefore, the RMS current capacity must be sufficient to handle the AC component of the current drawn by that upper MOSFET. It is common to use a mix of input bypass capacitors in parallel point. For quality and low temperature coefficient, ceramic capacitors can decouple the high frequency components. Bulk capacitors supply the lower frequency RMS current, which is tied to the duty cycle. (There is more RMS current when the system is operating further from 50% duty cycle.) The bulk capacitance can be several multi-layer ceramic capacitors in parallel. In lower cost applications, however, several electrolytic capacitors are typically used. In surface mount designs, solid tantalum capacitors may be chosen for C1, but be careful to note the capacitor's surge current rating. (Surge currents are common at start-up.) When choosing any capacitor in the buck converter system, look for small equivalent series inductance (ESL) and small equivalent series resistance (ESR) in addition to the total capacitance required. One final hint in regard to capacitor voltage ratings: To minimize hard-to-find failures, choose capacitors with ratings 1.2 to 1.3 times greater than the input voltage, that is, the voltage across them.

The output capacitor COUT must

filter the output to the load during a transient event. Interestingly, the equivalent series resistance (ESR) and voltage rating have more effect on the choice of capacitor than the actual capacitance value. Notice that the peak-to-peak current ripple from our inductor is transformed into peak-to-peak voltage ripple by the ESR of the output capacitor. Since the system probably has a limit on output voltage ripple, it is important to choose a capacitance (or set of parallel capacitors) that will minimize the ESR. Again, capacitors must have sufficient voltage rating. With this combination of requirements, approach the capacitor tables from vendors to find a suitable solution. One final caution, pay extra attention to the ESR data; it might not be given in the table at the same frequency as your switching frequency. Check the component datasheet for adjusted values of ESR.

The MOSFETs are typically chosen for Rds(on), total gate charge and thermal management requirements. Review several manufacturers' datasheets. Choose something like the Infineon BSC050N03LS with 35nC of gate charge and Rds(on) of 5 milliohms for the upper MOSFET. Complement that with the Rds(on) of 1.6 milliohms for the lower MOSFET (BSC016).

Closing the Loop

As discussed earlier, the output is fed back to the input. This connection creates a compensation loop. There are various types of compensations, such as Type I, Type II, and Type III. The type refers to the number of poles in the solution. Type I compensation is a single-pole solution, Type II is a two-pole solution with one zero and Type III is a three-pole solution with two zeroes. Each type increases in component count from the previous one, yet also allows for greater flexibility in design. For performance, set the bandwidth of this loop to be approximately a quarter of the switching frequency. In addition, make sure the phase margin is greater than 30 degrees and less than 180 degrees, a typical stability criterion.

> To minimize hard-to-find failures, choose capacitors with ratings 1.2 to 1.3 times greater than the input voltage, that is, the voltage across them.

The design process is similar with a hysteretic buck converter compared with a voltage-mode converter. Luckily, the high quality hysteretic-mode control helps overshadow the parasitics of the external components to ease selection of the 5 components discussed above.

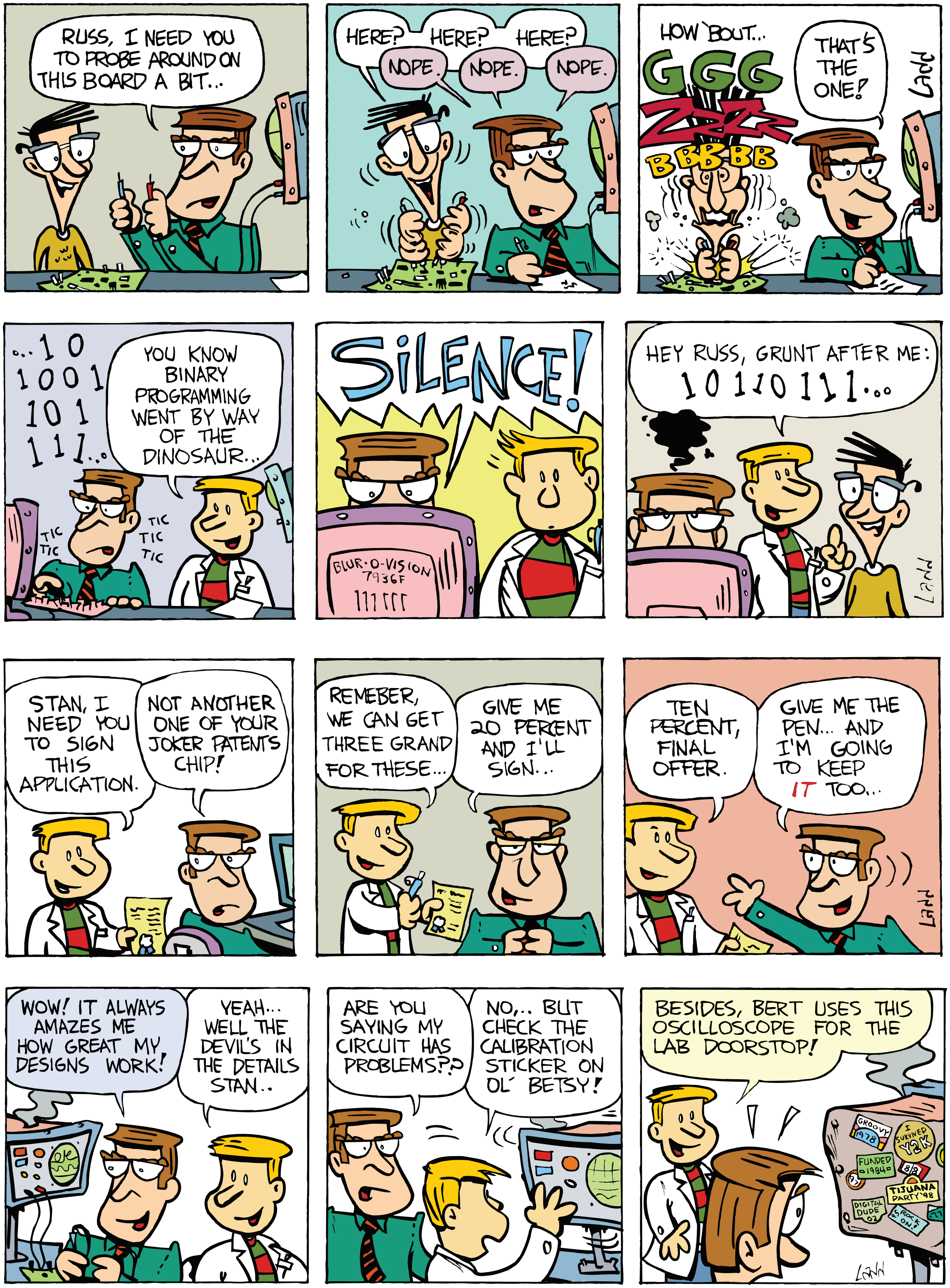
To summarize, the process of designing a buck converter, first choose a controller IC and then select accompanying external components. There are different parameters that are important for each selection. Once the MOSFETs, output inductor, input and output capacitors are chosen, finish with compensation.

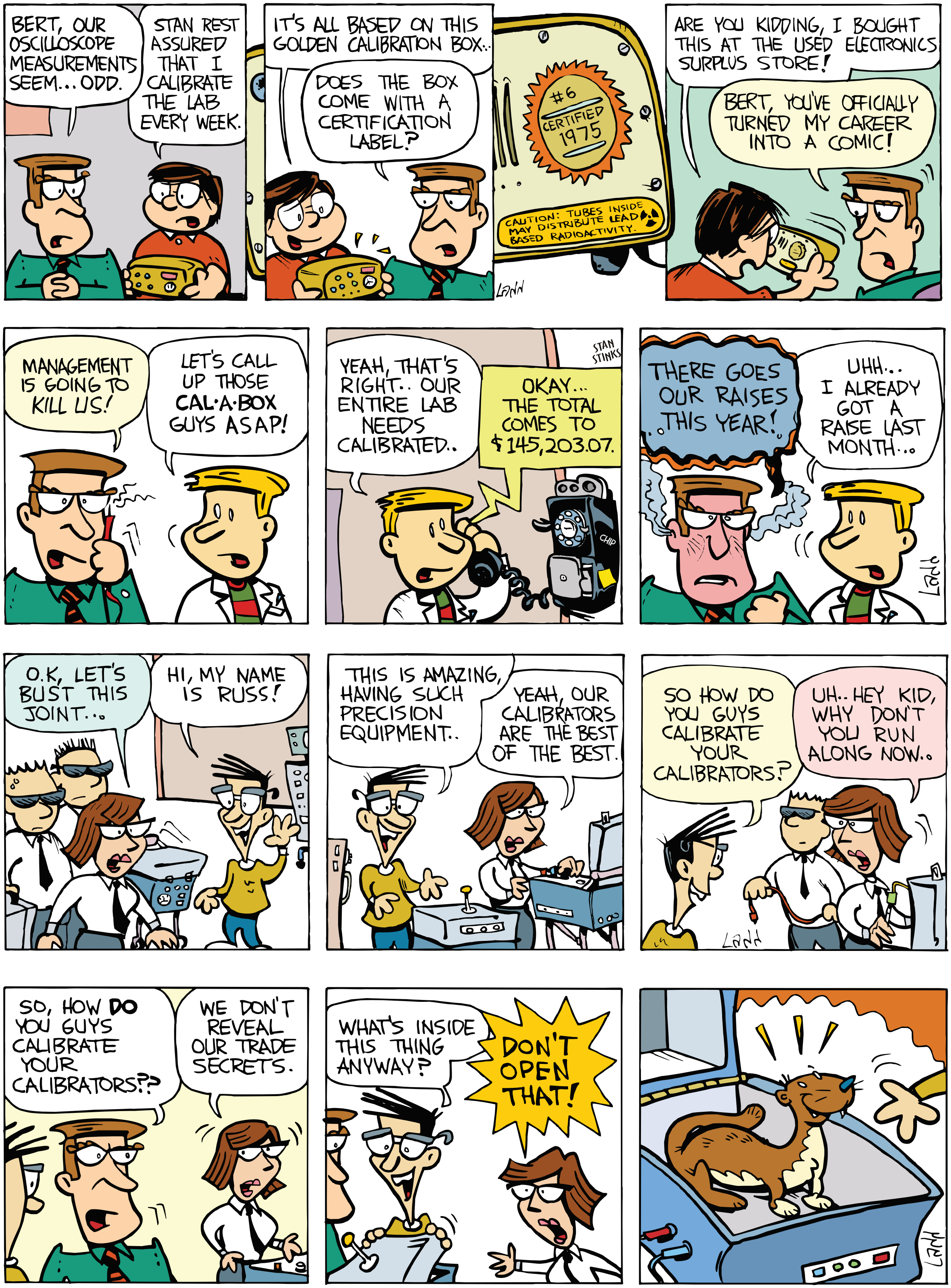
Plenty of work goes into designing a good buck converter — and more integrated versions are now available. Some designs have integrated MOSFETs. Some designs integrate the compensation. A select few have integrated the output inductor as well. One such offering is Intersil's ISL8201M. All that is needed is a resistor to set the output voltage, an input capacitor and an output capacitor. That is good news for busy system designers.

About Tamara Schmitz

Tamara Schmitz is a Senior Principal Applications Engineer and Global Technical Training Coordinator at Intersil Corporation, where she has been employed since 2007. Tamara holds a BSEE and MSEE in electrical engineering and a PhD in RF CMOS Circuit Design from Stanford University. From 1997 until 2002 she was a lecturer in electrical engineering at Stanford; from 2002 until 2007, she served as assistant professor of electrical engineering at San Jose State University.

RETURN TO ZERO




BERT, OUR OSCILLOSCOPE MEASUREMENTS SEEM... ODD.
STAN REST ASSURED THAT I CALIBRATE THE LAB EVERY WEEK.
IT'S ALL BASED ON THIS GOLDEN CALIBRATION BOX..
DOES THE BOX COME WITH A CERTIFICATION LABEL?
#6 CERTIFIED 1975
CAUTION: TUBES INSIDE MAY DISTRIBUTE LEAD BASED RADIOACTIVITY.
ARE YOU KIDDING, I BOUGHT THIS AT THE USED ELECTRONICS SURPLUS STORE!
BERT, YOU'VE OFFICIALLY TURNED MY CAREER INTO A COMIC!
MANAGEMENT IS GOING TO KILL US!
LET'S CALL UP THOSE CAL·A·BOX GUYS ASAP!
STAN STINKS
YEAH, THAT'S RIGHT.. OUR ENTIRE LAB NEEDS CALIBRATED..
OKAY... THE TOTAL COMES TO $145,203.07.
CHIP
THERE GOES OUR RAISES THIS YEAR!
UHH... I ALREADY GOT A RAISE LAST MONTH...
O.K, LET'S BUST THIS JOINT...
HI, MY NAME IS RUSS!
THIS IS AMAZING, HAVING SUCH PRECISION EQUIPMENT..
YEAH, OUR CALIBRATORS ARE THE BEST OF THE BEST.
SO HOW DO YOU GUYS CALIBRATE YOUR CALIBRATORS?
UH.. HEY KID, WHY DON'T YOU RUN ALONG NOW..
SO, HOW DO YOU GUYS CALIBRATE YOUR CALIBRATORS??
WE DON'T REVEAL OUR TRADE SECRETS.
WHAT'S INSIDE THIS THING ANYWAY?
DON'T OPEN THAT!