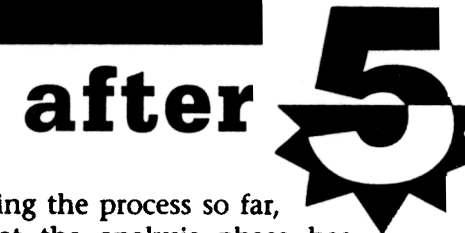


MAKING A TOASTER FIT FOR A KING



Welcome to the comics section — or the closest thing to comics in THE INSTITUTE. After 5 is the part of THE INSTITUTE where engineers can be themselves. You won't find any news or serious scientific writing here. Just sit back and enjoy this fairy tale.

If you have a humorous story with an engineering slant that you want to tell the world, send it to THE INSTITUTE via e-mail (g.gillespie@ieee.org), fax (732-562-1745) or by regular post to THE INSTITUTE, 445 Hoes Lane, Piscataway, NJ, USA 08855-1331.

BY SIVASIS PANIGRAHI
Spokane, Wash., USA

Once upon a time, in a kingdom not far from here, a king summoned two of his advisors for a test. He showed them both a shiny metal box with two slots in the top, a control knob, and a lever. "What do you think this is?" he asked.

One advisor, an electrical engineer, answered first. "It is a toaster," he said. The king asked, "How would you design an embedded computer for it?"

The engineer replied, "Using a four-bit micro controller, I would write a simple program that reads the darkness knob and quantifies its position to one of 16 shades of darkness, from snow white to coal black. The program would use that darkness level as the index to a 16-element table of initial timer values. Then it would turn on the heating elements and start the timer with the initial value selected from the table. At the end of the time delay, it would turn off the heat and pop up the toast. Come back next week, and I'll show you

a working prototype."

The second advisor, a computer scientist, immediately recognized the danger of such short-sighted thinking. He said, "Toasters don't just turn bread into toast, they are also used to warm frozen waffles. What you see before you is really a breakfast food cooker. As the subjects of your kingdom become more sophisticated, they will demand more capabilities. They will need a breakfast food cooker that can also cook sausage, fry bacon, and make scrambled eggs. A toaster that only makes toast will soon be obsolete.

"With this in mind, we can formulate a more intelligent solution to the problem. First, create a class of breakfast foods. Specialize this class into subclasses: grains, pork, and poultry. The specialization process should be repeated with grains divided into toast, muffins, pancakes, and waffles; pork divided into sausage links, and bacon; and poultry divided into scrambled eggs, hard-boiled eggs, poached eggs, fried eggs, and various omelet classes. At run time, the program must create the proper object and send a message to the object that says, 'Cook yourself.' The semantics of this message depend, of course, on the kind of object, so they have a different meaning to a piece of toast than to eggs.



"Reviewing the process so far, we see that the analysis phase has revealed that the primary requirement is to cook any kind of breakfast food. In the design phase, we have discovered some derived requirements.

"Specifically, we need an object-oriented language with multiple inheritance. Of course users don't want the eggs to get cold while the bacon is frying, so concurrent processing is required, too. We must not forget the user interface. The lever that lowers the food lacks versatility, and the darkness knob is confusing.

Users won't buy the product unless it has a user-friendly, graphical interface. When the breakfast cooker is plugged in users should see a cowboy boot on the screen. Users click on it, and the message 'Booting UNIX v. 8.3' appears on the screen. (UNIX 8.3 should be out by the time the product gets to the market.) Users can pull down a menu and click on the foods they want to cook. Having made the wise decision of specifying the software first in the design phase, all that remains is to pick an adequate hardware platform for the implementation phase. An Intel 80386 with 8MB of memory, a 30MB hard disk, and a VGA monitor should be sufficient. If you select a multitasking, object oriented language that supports multiple inheritance and has a built-in GUI, writing the program will be a snap."

The king wisely had the computer scientist beheaded, and they all lived happily ever after.