software people got involved with hardware. It seems to me in Eagan there's going to be plenty of room for a hardware division so why shouldn't some software people have a go at this? In a way the hardware is becoming more and more mechanical. The circuits aren't all that difficult. It's a matter of making choices--using judgement in choosing parameters--how much of this and how much of that? There's 20 or 30 dimensions in choosing a computer--what kind of registers, how many registers, how many semaphore flags has been a popular subject lately. I think it would be interesting and quite helpful to get more software involvement. I'm serious. I think it would be nice to do some in Eagan. I don't think you've got room in Mendota Heights, but in Eagan it would be interesting to see a software-dominated hardware design. I think it would be very helpful and it might be the answer. I struggle in choosing parameters with the inputs I get primarily from customers. I do listen to some people in Mendota Heights--not too many, but a few--but I listen primarily to customers. I don't try to follow them short term, but after they've had a machine for 2 or 3 years I ask them, "What's really wrong here?" They do tell me and I try to use judgement. I look for some common thread--some consensus, and so far in my career there hasn't been much consensus which makes this very hard. Each customer has a different view of what is important to him. Maybe it's not true when we have hundreds of customers, but I was working with dozens of customers, and I tend to work with the same ones because we've established this rapport. I still see this diversity of feeling. What has happened in recent years is people have begun to appreciate simplicity. This has happened on the part of our customers. It's obvious that you should have more of everything but some customers are beginning to say, "You know, I think I see some merit in not having a lot of complexity." In fact, it's sort of a fad. RISC computers are popular now. I'm encouraged that I was on the right track trying to keep it simple so I'm going to keep on that trail, but I can easily make mistakes. Sometimes we have to fix them up on a product. I think it's wrong to have all the software people in Minnesota and all the hardware people in Wisconsin. That doesn't seem right. How do we mix this up? We're building a nice big building in Colorado, so I've invited Bob Ewald to send people there. Why don't we have a significant contingent of software people to help with CRAY-4 design? For that matter, we could do a