Excerpt from Oral History with Gardner Hendrie

Interviewed by Len Shustek, June 4, 2007

CHM Reference number: X5690.2010

Oral History excerpt description:

DDP-116 General Purpose Computer, 1965

The 116 was the world's first 16-bit minicomputer and reasonably successful, with 172 shipped. But its successors, the IC-based 416 and 516, sold more than 750 machines. In this oral history excerpt, Gardner Hendrie, the designer of the DDP-116 describes the origin of the machine.

Shustek: So you were assigned to design this new computer.

Hendrie: Yes. So Fritz walks me up to Bill Wolfson's office up on the second floor. He's sort of head of marketing. He basically says to me, "Okay, we have a computer that sells for $89,000", and he shows a little graph of sales history. It's doing very nicely, growing nicely. He says, "But, you know, I think that the market is elastic, and if we could build a computer for $30,000, we'd sell a lot more of them. So your job is to figure out how to build a computer that we can sell for $30,000."

Shustek: Okay. No other constraints?

Hendrie: That's all he said. That's all there was. So I went back to my office, and I started doodling around with, well, what are the various kinds of machines that I might be able to build for $30,000? I looked at drum machines. I actually looked at quartz delay line machines, because a company in Michigan had a minicomputer that was built on quartz delay line. Sonic delay lines, but in quartz rather than magnetostrictive delay lines.

Shustek: Or mercury or anything else.

Hendrie: Or mercury, yes. Mercury is the worst of all worlds. Of course I looked at a core memory machine. I looked at whether it'd be serial or parallel. I just sort of said, well, nobody's told me anything except what the price has to be, so I'll just try to think about all the alternatives and then work my way down to what I think is the best choice.

Shustek: Sounds like a great job assignment.

Hendrie: Oh, yeah. It was really fun. Now you know why I took the job. I wasn't being told, "Build this, and here are the instructions, here's the specs." There were some role models out there to look at. DEC had the PDP-5. I don't believe they'd introduced the PDP-8 yet. There was a CDC 160 that'd been around forever. There were the Ramo-Wooldridge machines. We were in the real time business. I mean, the DDP-24s are almost all being used to control flight simulators, which is a real-time application. You're running a piece of equipment and trying to simulate the flight characteristics of an aircraft. I looked at some of Ramo-Wooldridge's machines they'd developed for process control. Of course they'd started out with a drum machine, but they had come up with some core machines later on. Finally, I said, "Okay, I think we need to build a core memory machine." One of the good things was that 3C, among their products, they had another product line besides their digital circuits. They built core memory systems, and they had a group that was competing in the core memory system business.

Shustek: Selling to whom?

Hendrie: To anybody who needed a core memory. This was black art, remember. This is still '64. And they built very high performance ones which, of course, is a lot blacker art than slower ones.

Shustek: Were they selling to computer manufacturers who were competing with the computer division of 3C?

Hendrie: No, they were primarily selling to people who were building special systems that needed a core memory.

Shustek: Okay.

Hendrie: Special electronic equipment. That's my understanding. So I had a built-in source of core memories. That wasn't bad. Of state-of-the-art core memories. I'm sure that's probably what wiped out all the other ideas pretty quickly. But, of course, I did have to find out how much they cost and what the cost implication [was]. We also had a series of digital circuits called S-packs that were one and five megahertz, so I knew I had a good line of circuits. I didn't have to develop any circuits. Fundamentally, what I had to do was design the system. I decided that we needed to build a binary, short word length core memory system. I'd had enough experience with the PDP-5 at Foxboro that I knew that 12 bits was not long enough for a reasonable instruction word plus addressing ability. So I said we got to make it at least 14 bits. I think the first spec I wrote out was a 14-bit machine that could indirectly address, because CDC taught me about indirect addressing. It was a good way to address larger memory than you can fit into the fields of the instruction word? They could indirectly address, well, whatever 14 bits can do. That seemed reasonable. That was going to give me, what, 16K, yeah. I would be happy with that. Well, this is early '64. I think I joined sometime in early April of 1964. I'm not there more than a couple of weeks and IBM announces the 360.

Shustek: In April of '64.

Hendrie: Yes. And so the marketing guy -- not Bill Wolfson, who was the V.P., I think he was V.P. of sales and marketing -- the marketing guy, a fellow by the name of Sid Halligan[ph?], I go up to see Sid Halligan and talk about my machine and he has only one pitch. The 8-bit byte is going to be it. Everything's going to be modulo-8 in the future. You really ought to make it a 16-bit machine. I said, "Yeah, that makes sense." So, thinking about my cost target a lot, I said, "Ohhhh, well, all right, we'll make it a 16-bit machine."

Shustek: That raises the cost 10% or something.

Hendrie: Exactly. That's why I wanted as short a word length as I could build a reasonable machine. So made it a 16-bit machine. The core memory technology we had, I think it was a one and a half microsecond memory. It was pretty darn fast in those days. I can't remember exactly, that's terrible. So I proposed it be fast, because I concluded that one of the great things about a short word length machine versus a longer word length machine -- yeah, you were going to have to execute more instructions, but it was easier to make a short word length machine fast than it was a long word length machine. You could always substitute executing more instructions, doing more instructions, but [if] you did them real fast, you'd still get the job done just as fast as you might in a long word length machine [where] each instruction did more.

Shustek: It's a lesson that, 20 years later, people would learn in the RISC field.

