Oral History of Charles (Chuck) Thacker

Interviewed by:
Al Kossow

Recorded: August 29, 2007
Mountain View, California

CHM Reference number: X4148.2008

Al Kossow: It's August 29th, 2007. This is tape #1. We're talking to Chuck Thacker today. Thanks for coming by.

Charles (Chuck) Thacker: My pleasure.

Kossow: And I had prepared a list of things I'd like to talk to you about kind of starting with kind of the bio things, just where you're from, when you were born, all the usual stuff like that.

Thacker: Sure, well I was born in Pasadena, California in February of 1943 and I grew up mostly in the Los Angeles area until I wound up going to Berkeley when I moved north, and I've never moved south again.

Kossow: You have any siblings?

Thacker: I have one brother.

Kossow: Okay, is he still down there?

Thacker: No, he lives in Reno. He's not really a technologist at all.

Kossow: Oh, okay. I read the talk that you had with Mihai Budiu in October of 2006. So there are a couple of things that came up from that, that you wanted to be a physicist early on. So, was your father in technology?

Thacker: My father was an engineer and I wanted to be a physicist not because of the physics but because of the engineering part of it. I wanted to be an experimentalist and in those days that meant particle accelerators.

Kossow: Where'd your dad work?

Thacker: He and my mother were divorced when I was quite young so I never really knew him very well, but he worked on power stations and things like that.

Kossow: He was an electrical engineer.

Thacker: Yeah, but I was attracted by, you know, the large number of technological problems in a particle accelerator, spanning the gamut from, you know, logic design of the data acquisition stuff to designing enormous power supplies for things like the magnets and the klystrons [ph?] that run them, and all that sort of thing. And so it seemed like a good way to really do some very interesting and cutting edge engineering.

Kossow: Where did you go to high school?

Thacker: I went to high school in Highland Park, California, a high school called Franklin, which is now not really a very academically oriented place. But, you know, the demographics of Los Angeles has changed a lot since I lived there.

Kossow: And then from there you went to Cal-Tech?

Thacker: I went to Cal-Tech for a while. Then I went to UCLA for a while and then finally I moved north and went to Berkeley and I finally graduated from Berkeley.

Kossow: So, when did you go to-- ?

Thacker: 1963.

Kossow: Okay, and then your Bachelor's was in?

Thacker: Physics.

Kossow: In?

Thacker: 1967.

Kossow: Okay, and then you started on Genie in '68?

Thacker: Yeah, I had planned to go to graduate school but I was actually married at that time. My wife and I were married in '64.

Kossow: Okay, you met her up here?

Thacker: Yeah, I met her in Los Angeles, actually, and I had planned to go to graduate school but since I was fully self supporting, I had to work for a while before doing it. So, I took a job, actually, working for the only inventor in the Berkeley phone book, a guy named Jack Hawley whom you may know peripherally. Yeah, we've actually put Jack in the mouse business.

Kossow: So, what was he doing back then?

Thacker: Well, he had run and I met him at a company that I had worked for earlier called Berkeley Instruments, which made digital automated weather stations. And they were very interesting to people who wanted to put weather stations on top of mountains and things like that, because they were very low power. They used transistor circuitry but with a lot of fairly clever mechanical stuff in the sensors. Jack was a very interesting guy because he knew a lot about mechanics but not much about electronics. So, when I graduated and was looking for a place to work, he had set up as a sole proprietor of an inventorship in Berkeley. Actually, it was in Albany and I went to work for him, I told him, for a fixed period of time and I made him a deal that he couldn't refuse. I said, "I'll design your electronics for you, if you teach me how to run all your machine tools." So, he did and he had, you know, he had a good lathe, and a good milling machine, and a good drill, and all the surrounding tools that you need to actually build mechanical stuff. And so I learned how to use all that, became a fairly good machinist.

Kossow: Have you done much of that?

Thacker: I have not and I would be absolutely doomed today because it's all digitally controlled now, and of course at that time you used micrometers and carefully turned the dials on the machines in order to fabricate things. But in any case, one day a friend of his dropped by the shop named Paul Persons, I believe, who was a rep for an electronics distributor. And he had been calling around in the Berkeley area and he called at the Genie project, and he mentioned that they were looking for an engineer and he thought I might like to interview there. And I said, "Well, you know, I'm pretty happy with this." Jack said, "Well, you know, the project is coming to an end," and so on and so forth. So, I went and talked to the Genie folks and I guess they liked what they saw because they hired me, and that was in some ways the best decision and in some ways the worst decision in my life, because I never finished graduate school. I got so interested. I had worked with computers before, but--

Kossow: What did you work on?

Thacker: Only peripherally. I mean, I started out by trying to program an LGP30 but discovering that, you know, the challenges of actually spacing out the instructions on the right place on the drum were just daunting. And so I didn't really enjoy it, so I stopped doing that, and then as a physics student, I actually had to write programs because FORTRAN programs are needed for analyzing experimental results. And so I learned how to program in FORTRAN and programmed a 1620, in fact.

Kossow: Where did you pick up your electronics background?

Thacker: At Berkeley. I took both physics courses and engineering courses, but a lot of it was just building things.

Kossow: Okay, so you weren't ever a Ham or anything?

Thacker: I was a Ham. I was a Ham radio operator. I was in a science club. I was in all the things one did in high school if one was a young nerd, and so where was I? Right, so Genie held together for another two years, roughly, and they had built one of the machines that actually captivated me, which was a 940, and wrote a timeshare operating system for it.

Kossow: So, I'm really curious about that because that's where everybody starts coming together. You meet Butler [Lampson].

Thacker: Right, I met the whole bunch.

Kossow: And Peter Deutsch.

Thacker: Right.

Kossow: So, what was it like? What was the environment like? Was Ken Thompson there then?

Thacker: No, he was not, or he may have been somewhere but he wasn't associated with our project.

Kossow: Because he kind of drifts through it.

Thacker: Yeah, he drifts in and out, but we were on the fourth floor of Corey Hall, which was at that time the top floor, and in a little thing on the northeast side, which was the suite of offices and a lab for the project. And the 940 itself was in a machine room down on the first floor.

Kossow: Was the 6400 in by that time?

Thacker: No, the 6400 was in the computer center, which was down the hill, more on the central campus. Because at that time, there were two computer science departments. There was a computer science department that was part of the college of letters and sciences, and there was the computer science department that was part of the EE department. And now they've merged, they no longer have that distinction, but the one down the hill was actually trying to build a timesharing system for the 6400, which never apparently worked out very well.

Kossow: That was Paul McJones and Butler.

Thacker: Paul McJones and Butler was there, and Charles Simonyi, and so the 940 was actually a marvelous machine and I enjoyed it a lot because for the very first time you could sit down and program interactively, which was a new thing. You know, I had programmed by punching punch cards and getting a printout, a print deck back.

Kossow: So, had Dave Evans left by that time?

Thacker: He had left, yeah. He had left. That's right.

Kossow: So, it was Wayne [Lichtenberger] , and Mel [Pirtle]?

Thacker: People often believe, and in fact, Dave Patterson believes that I actually took a PhD from Berkeley. In fact, they made me an honorary distinguished alumnus a few years ago, and I asked the chairman of the department at the time, I said, "Wait a minute, but I didn't graduate from the engineering department. I was physicist." And then he fessed up. He guiltily said, "Well, the problem is, Chuck, we've been going two of these year and we're now out of distinguished graduates. So, we're broadening out net." And I said, "Ah, in a few more years, it will be anybody in the U.C. system." But, in any case, yeah, Dave had left.

Kossow: So, I spent a few days with Wayne. Wayne is a really nice guy, but I never had the chance to meet Mel.

Thacker: Yeah, he's great.

Kossow: So, what was Mel like?

Thacker: Well, Mel was an interesting guy. He was the project leader, the principal investigator, and very smart, good teacher, not a great businessman. But I never really got to know Mel very well. I did get to know Butler and Peter, because Peter and I shared an office. But, Mel traveled a lot, for instance. So, in any case we had planned to build a follow onto the 940, a much larger machine and we were planning that and figuring out how to do it, and we finally realized that it would just be very difficult within the constraints of the university to actually do this. Because we were talking about a lot of money, and although ARPA was a fairly free spending agency at that time, particularly for innovative computer stuff, this was even kind of beyond their capabilities. We knew that it would take, you know, several million dollars and so we looked around for alternatives and we decided to do what was then a startup, although that term had not really been coined yet. And we looked for investors, and Mel and I actually went to New York one time and talked to Arthur Rock, and put together the necessary money, although we didn't really get as much as the business plan said we would need, and that's bad because that was what resulted in the demise of the company. We just couldn't get it funded and there was finally a recession in 1970, which made it even harder.

Kossow: Right, did the [Univ. of California] Regents end up investing?

Thacker: Yes, they did and I think they are a little bit irritated about that, but, you know, they're no longer regents anymore. So, it's water over the dam now, but in any case we built the machine and it worked more or less to spec, and it was quite an astonishing piece of engineering because it had two large drum memories, a large Bryant disc file. It had core memory, a central core memory.

Kossow: Sort of flaky Ampex memory.

Thacker: It was kind of flaky Ampex memory, that's right, and it was fronted by a thing, which today we would call a cache. We called it the fast memory because it was very small, of course. And then it had five independent other processors, there was one that handled scheduling for the whole system, one that handled I/O to the terminals, one that controlled the rotating memories, and two which were slightly different. They were specialized to be CPUs. So, this was a multi-processor in 1968.

Kossow: All microcoded.

Thacker: All microcoded. Microcoded, the machine was very interesting because we wrote, actually I wrote this, and micro assembler which complied for the microinstructions of the machine, or assembled for them, the output of which was paper tape. And that paper tape was fed to a numerically controlled drill, which drilled holes in a fixed pattern circuit board, into which you inserted diodes. So, this was a diode matrix memory. We actually experimented with several different kinds of memory because IBM at that time had been building all kinds of strange microcoded things, transformer, read only memory, capacitive. We actually had one of their capacitive memory machines upstairs.

Kossow: Controlling the tape drives.

Thacker: Controlling the tape drives, and the cardpunches, and the line printer, because IBM actually used to build the most reliable peripherals on the planet, and we'd had a lot of experience at the Genie Project with buying flaky peripherals because they were inexpensive. One of my first jobs there was to make a printer that they had bought, an old chain printer, actually work and so we decided to bite the bullet and rented an IBM Model 30. The IBM Model 30 was one of the very few computers that I have ever seen, in fact probably the only one that part of its equipment makeup was an air compressor. And the reason is, is that the capacitive storage had to be held carefully between the plates of this capacitor. The storage was actually IBM cards made out of Mylar, and the lower capacitance when you had made the whole, made the storage work. And so they wanted to gently push it, so there was a little air pillow on one side of the sandwich that pressed the card gently and evenly into the arrangement. It was quite crazy but we used diodes, but yes, I wrote a lot of microcode for those machines.

Kossow: So, when did Ed Fiala come out from the East Coast?

Thacker: Fiala came out from, I guess, BBN.

Kossow: So, he was at BCC, right?

Thacker: Was he at BCC?

Kossow: Or did he come out for MAXC ?

Thacker: It's possible, oh that's right. No, I think he came out for MAXC. That's right.

Kossow: Because there's some conflicting information on him. In some cases, he was with the BCC folks?

Thacker: Well, he was certainly with the BCC folks to the extent that they all went to PARC, but I don't think he was at BCC. So, the machine was working and we were beginning to commission it, and we were looking around for money to build more and we just couldn't get it. So, in those days--

Kossow: So, what's the connection between Shell Development and BCC?

Thacker: In the early days, we rented time on the Shell Development 940 machine for software development.

Kossow: Okay, so there was this disconnect by that time between Cal and BCC?

Thacker: Yeah, pretty much, and that's reasonable. I don't think there was any acrimony there. It was just that, you know, they still don't like university facilities being used by private companies. So, no, Shell Development was just a source of compute cycles, because we certainly didn't want to buy a 940. It was too expensive and we didn't want to program a 360.

Kossow: And there's just this whole body of code that gets dragged forward from Genie onto the BCC machine.

Thacker: Not really. The machines were actually quite different.

Kossow: But you could emulate a 940.

Thacker: That's true. That's true.

Kossow: But by that point--

Thacker: That turned out to never be a very big part of the whole consist. There was new software for almost everything and you actually have a lot of that software on your website. I've looked at some of those things, including memos that I wrote 40 years ago that I don't remember writing.

Kossow: Those memos are really fascinating because of the structure. Every startup I've been involved with, you'd never write memos like that.

Thacker: They look like RFCS don't they?

Kossow: Exactly, they look like reports that would come out of-- amazing history.

Thacker: That's just the way we did engineering in those days because you had to do it that way, because you had to keep track of everything by the documents. I mean, if you read Fred Brooks' book, he talks about what's needed for a large software project, and one thing is a librarian to control the documents.

Kossow: Did that follow through in the early days of PARC then? Because I've never seen the early PARC documents.

Thacker: Yes, it did, although probably to a lesser extent. But if you think about it, there were a lot of manuals written for a lot of things and usually the manuals were written by and large before the thing was built and served as specifications. So, right, so we tried to figure out how to move BCC forward and just really couldn't do it.

Kossow: And then the recession.

Thacker: And we cut a deal with ARPA to buy it and move it to the University of Hawaii, because they needed one. They needed a computing system for their computer center, excuse me, could I get a little bit of water here? So, in 1970 we, having talked to ARPA about getting rid of the machine, it turned out that Bob Taylor who used to be the IPTO Director, but by that time had taken a position at the University of Utah, but didn't really like it there--

Kossow: He was only there for, like, a year.

Thacker: He was only there for a year, yeah, that's right, and we started talking to him because he had been hired by George Pake to set up a lab for the Xerox Corporation. And he was looking around for people, and of course he turned to his old ARPA buddies, and so I was actually only in that crew peripherally, in the sense that I was never an ARPA principal investigator, for instance, as Mel had. But in any case, when he went to PARC finally and relocated, he came up to BCC with George Pake and interviewed many of the engineering staff. I'm not sure exactly how many, but certainly I think all the major players.

Kossow: Inside BCC?

Thacker: Yeah, at BCC, because it was pretty clear by that time that the company was winding down, and some of us actually decided to take the leap. I was a little bit nervous about moving to the South Bay, because I'd never been there very much. And I'd actually just had my first child, my wife had our child, and so uprooting her would have been a problem, but I decided that the opportunity was probably pretty good. Best decision I ever made. So, I actually went before most of the others because the software was not quite done, which is the usual situation in these matters.

Kossow: Okay, and they had to deliver something to Hawaii?

Thacker: And so the people who were actually going to do with the machine to Hawaii were the ones who packed it up and moved it, and I moved to PARC. So, I was actually one of the very first people to arrive in the lab.

Kossow: So, is 3180 Porter still the same building?

Thacker: No, it's been destroyed.

Kossow: How big a building-- ?

Thacker: 3180 Porter used to be two buildings, which were about, one of them was about 100 feet on a side and had sort of an open garden in the middle, sort of. So, it was offices in a ring around the central court, and then there was another non-descript building which was not used for much of anything, but it came with the deal.

Kossow: So, you were one of the first people to go over and work for Bob?

Thacker: I think I was the first. I'm not positive of that, but very shortly thereafter, Jim Curry and Bob Flegal came from Utah. Curry was subsequently influential in the development of the BCPL complier and Flegal went off and worked with Dick Shoup on graphics. And Dick Shoup came over from BCC. He was originally from Carnegie and then the rest of the people began to arrive, Butler and Peter. Charles Simonyi had elected to go off and work for Mel at Ames, which was setting up a center for the newly hatched ILLIAC IV. And so Charles worked there for a while as he was going to graduate school at Stanford, and then he came to PARC later.

Kossow: So this will be early 1971?

Thacker: This is early 1971 now, and we looked around for what to do about computing and we thought about buying a 940, and that would have been very politically correct, because Xerox had just bought Scientific Date Systems. But SDS didn't really like the 940. They were trying to replace it with another family called the Sigma Family, and we looked at the Sigma Family with the idea of what it would take to build a timesharing system on it, and decided that it was just not the right system. I mean, the Sigma was originally architected as an attempt to compete effectively with IBM in IBM's markets, not realizing that if you're going to compete with IBM's markets, you have to have all the IBM software, because the software is what matters. And so it was never a very successful machine, and we finally decided that it wasn't really what we wanted because it was not what the university research community, and particularly the

ARPA community used for doing research. They used the PDP-10 suitably modified by a set of hardware that had been built at BBN called the BBN Pager.

Kossow: So, you had never thought about doing another BCC?

Thacker: No. Well, I think we were all pretty tired and we were investing a lot in sort of setting up the new lab, you know, sort of building the necessary infrastructure to make it actually tick. And so after quite a bit of thought, we concluded that the right thing to do was, since it would be unseemly to buy a PDP-10 from an SDS competitor, that we would build one. And so we designed the MAXC system.

Kossow: What does MAXC stand for?

Thacker: Multiple Access Xerox Computer. It is also a play on the name Max, as in Max Palevsky, who was the President of SDS. And that was actually planned and designed very rapidly. I mean, it was an interesting machine because what we decided to do was rely on commercially available packaging technology, primarily. And that meant wire-wrap cards that you could by from Augat, and backplanes, and so on and so forth. So, everything was wire-wrap.

Kossow: Was Dave Poole or Pettit, or any of the guys from SAIL involved in the design?

Thacker: No, they were not. They were up on the hill doing other things at that time.

Kossow: So, it was you and--

Thacker: Pettit actually joined SRC later for a while, until he went off to Pittsburgh to be a mystery writer, right after they did the Foonly.

Kossow: So, it looks like from the specs that, I think the first MAXC came up in the summer of '72.

Thacker: Something like that, yeah.

Kossow: Because there's an 8.1 version of the spec that says it's going to be and then by July it says it is. So, it's probably, you know--

Thacker: I mean, that was a long time ago.

Kossow: One of the bits of folklore is the story about doing your own 10 and then how it comes up so fast.

Thacker: Right, well we were very careful because we knew about the second system effect, right. BCC was the second system. It took a long time to get right.

Kossow: So, yeah, this is the third. You did Genie, you did BCC--

Thacker: And by that time--

Kossow: You knew how to build timesharing systems.

Thacker: We knew how to build systems and so the only thing that was kind of aggressive that we did was we said, we will not use core memory. We will use this newly invented semiconductor RAM called the 1103.

Kossow: Which was a total nightmare.

Thacker: Which has an interesting story. So, we designed the memory board for the machine. I remember it well. It had 96 1103 chips, and they were semiconductor memory chips with external sense amplifiers. And these were packed in a rack with four back planes, each of which contained, had to get it right, 12 of the cards and controller, or maybe, no, it was more than that. And then there were four cabinets, and they were strung together by cables. We designed our own cables and connectors for them. Our association with Jack Hawley began at that time, because we had him make the connectors. Very nice engineering job for low price, but they were all twisted pair and transmission line terminated at the end. So, electrically the thing was quite conservative and it worked very well, except for the 1103. But we knew that there was a potential for problems with the 1103, so we put single error connection and dual error detection in the memory. And when the system came up and the CPU started being able to run memory exercises, the memory worked flawlessly. And so later, when we got ready to do the Alto, we said, "Well, let's just use the MAXC boards as the memory for the Alto, because we know how they work. They work very well, and by the way, we don't really need error correction. We can just put parity on the memory." Mistake, because the thing that I didn't realize is that although I had put extensive logic into the memory system for correcting the errors and reporting the existence of single errors to the system, the software guys never read. And so the thing was actually sitting there making a lot of errors, but it was fixing them and so we never noticed. And when we put it into the Alto, the same result happened. It looked very reliable, very rarely got a parity error until we started running more complicated software that exercised the memory more thoroughly and then suddenly we would see various failures. And so this turned into an arms race between me and Intel. I would write a diagnostic to track down some particular perverse flaw in the 1103, and then we'd run it. By that time, we had e-mail and we ran it overnight on the Altos, and the machine would automatically detect errors and send e-mail to the technicians. So they could come in, in the morning, and change out the bad boards, and then run them through another tester that we built, that could by that time do very comprehensive testing of the board. After that, we always used error correction and I'm amazed that present day PCs don't. Servers do but desktop machines do not, and I actually suspect that many of the bad events that happen on the PC are actually memory errors, because memory's not as reliable anymore and it's getting less reliable. So, in any case the MAXC came up and we used it. We subsequently built, I think, two more, one or two more, which were slight improvements, but nothing serious, because by that time we finally understood that we ought to have built the Alto and we did it.

Kossow: So, there's that run up to that in '72 when there is the TFS, all the timesharing stuff that was going on. VTS.

Thacker: Yeah, there was another system being built in the laboratory, not in our lab. There was also a thing called the System Sciences Lab, which was directed by Bill Gunning of GTL fame, and they had built a system based on interconnected Nova mini computers.

Kossow: So, is that really where the affinity for Novas starts?

Thacker: POLOS, no, actually it started before that. They chose the Nova to some extent because we did it, and we did it because we actually did a bakeoff between the PDP-11 and the Nova, and for our purposes, which was programming in higher-level languages, the Nova actually worked out to be better because it was faster. And it got through its instructions faster, and the 11 was better if you wanted to write assembly language program, do assembly language programming because that's what Gordon Bell designed it to do. And so there were always several different ways to do the same thing on the 11, and so the compilers always had to figure out, well, which one do I use in this circumstance? And it made the compilers more complex, and we didn't like that. So, we liked the Nova and of course we patterned the Alto on the Nova with a few improvements.

Kossow: Pretty much all of your tools for doing MAXC and everything were running on Novas?

Thacker: Correct. All the debugging stuff for the Alto itself ran on a Nova, but we never really thought about using them as part of a distributed system, and the POLOS, PARC Online Office System Group, which built the system out of a multiplicity of Novas had that idea in the beginning. The Nova had kind of a clunky network for interconnecting something like 16 machines.

Kossow: So, that was like Chris Jeffers and Bill English?

Thacker: It was Bill English, and was Chris there or was he with Alan at that time? No, I think he was with Alan Kay in the learning research group. He had a third group. So, we had been thinking about smaller machines. We had a couple of IMLACs and really liked the idea of interacting with this screen in a way that you couldn't really do with glass teletypes. But the IMLAC was kind of bad because it used a magnetic calligraphic display, where the beam is driven around to various positions and then turned on and off, so you can draw lines, and you can draw characters that way, but they're kind of ugly. And you have to draw them pretty rapidly because you have to repeat 30 times a second. So, a much better solution is the raster scan display, and in fact the POLOS people had designed one, which used a custom character generator to generate more or less arbitrary fonts, and it looked really nice. And we kind of liked that, but it was expensive. We said, "Well, if we're going to build a lot of these things, it's got to be cheap."

Kossow: And that's where the industrial design for the display came from?

Thacker: The industrial design for the display was done for the POLOS project. That's right. We just took it, and their display actually had a little bit higher resolution than ours, but we thought about it a bit and decided that 606 x 808 was probably good enough. And it was good enough that if we could produce an arbitrary bitmap, then we could really do a lot of pretty interesting things with that display, and we did. The display controller was extremely simple, because the Alto was arranged in such a way, and this was something that had actually been invented once before but we didn't know it, on the TX-2. When we heard about it, Ed McCreight and I were talking to Wes Clark and he told us that he had done something similar on the TX-2, and we said, "Oh."

Kossow: The task multiplexing?

Thacker: The task multiplexing, that's right, and Wes said, "Yeah, the paper appeared in FJCC in 1965 or something," and Ed said, "Well, Wes, the only excuse I can make is that at that time I was five years old, or nine years old, or something like that." But in any case, the task multiplexing was very fast. You could switch tasks in one cycle, and the result of that was that we kind of turned around the way computers had been architected until then.

Kossow: But you had done similar things with MAXC already, with the disc controller.

Thacker: Not as thoroughly. I think that was kind of an exploratory probe, but what we wound up doing was making all the hardware for the device controllers very, very simple and doing the hard work in microcode.

Kossow: And by that time, you had bipolar PROMS, so you could actually have reasonable sized micro stores.

Thacker: Reasonable sized, but still not great speed. So, where reasonable, I think, was 4K instructions rather than 1K. And so in a normal minicomputer, the memory is multiplexed between the IO devices plus the processor. Now, that's not the way it works in the Alto. In the Alto, the underlying microcoded machine is multiplexed between a lot of tasks and it owns the memory. So, we multiplexed the computing rather than multiplexing the access to memory and that leads to an enormous simplification in the device controllers, which makes the whole system cheap, particularly if you like complicated devices, which we did.

Kossow: Well, maybe this is a good time to break then.

Kossow: It's August 29th, 2007. This is tape number two, oral history of Chuck Thacker. So, we were talking about the early days of the development of the Alto.

Thacker: So, we designed it. It was actually based on a memo that I had written and then a follow-up memo that Butler had written to the laboratory about why we ought to do this. And finally decided to do it and built about a half dozen of them and see how it worked. And so we used all the same technology that we used at MAXC, and the memory boards, and the back planes, and the so on and so forth, but with a much smaller design. And I had worked out the design based on this idea about multitasking. And so we divided the machine up between Ed McCreight who did the disk controller and disk control microcode. I did the CPU, and the CPU microcode, and the display controller, and that was really all initially. A little bit later, like six month, Dave Boggs and Bob Metcalfe started, when we realized we needed a network, Dave Boggs and Bob Metcalfe started doing the Ethernet. But the original machine was just the display controller, the memory, the CPU, and the disk controller. I think that's right and later there came a few other things added. The Ethernet controller was one and more complex controllers for things like laser printers were also built a little bit later. But the early machine was just that, it was very simple and the microcode was all in ROM, Read Only Memory, not programmable and so that posed a bit of a problem for doing things like debugging. We built a large gizmo that you strap on the back of the Alto to give it read-write control store and also run a debugger interface so you could do a credible job of debugging the micro program as it ran on an attached Nova. And we brought the machine up that way. The whole thing took about, literally, three months.

Kossow: So, fall of '72?

Thacker: We started in the winter of '72 and finished all six machines by the middle of '73, I think. The timeline sometimes fades into oblivion, but in any case that first phase was very, very quick and the machine was ready, of course, as is usually the case, ready before the software. So, we had a fair amount of time to sort of hone things in preparation for having to run the real software.

Kossow: Gene McDaniel was working on the initial-- ?

Thacker: Gene McDaniel worked on, yeah, a lot of the initialization stuff and some of the microcode, and debugging it. We worked very closely together debugging it. Yeah, that was when I first realized that we really had something with bitmap display because we got, in a micro program, no CPU program, we got the Sesame Street Cookie Monster to eat a cookie. And that was one of the few aha moments I've ever had in graphics, the others being what, when I first saw a PC running color graphics, when I first saw Doom running on a PC, amazing piece of work. And what recently? The other stuff is all evolutionary. In any case, pretty soon we were beginning to use them for computing in the laboratory and then, of course, all of the good software that was built for the machine was actually written.

Kossow: So, how did that end up happening? The people just needed a tool and they just started writing it?

Thacker: Well, we were lucky because we actually had a goal for the laboratory, which was essentially to create the paperless office. We didn't realize that that was not what we were going to do at all, and in fact we were going to contribute to the death of a lot of trees, but fortunately trees are relatively renewable if you're using the kind you use for paper. But in any case, so we needed things like text editors for program editing. We needed, you know, compilers to compile programs. We needed a little operating system just to run the IO system. We needed a disk file system. Most of that stuff was all stuff that we needed or wanted to use to satisfy our own needs for various things.

Kossow: What was the size of the group? Was it a few dozen?

Thacker: Oh yeah, a couple dozen. And so the high quality text editing and the laser printer grew out of our desire to produce camera-ready copy for papers. I don't remember if you remember how that used to work but you started out with a large piece of poster board, essentially, that the journal sent you, with the display area outlined in light blue and then you used rubber cement and a pair of scissors to format your text. A very time consuming and error prone, and actually ugly process, and so we stopped all that. We could produce camera-ready copy immediately. And the same thing with laser printing, I mean that grew up because we wanted it. So, we were lucky because what we could do is we could do work that was actually new things in the world because we needed them and the motivation of being able to do things that you actually want is extremely high. And it's hard to do today because you have to sort of cast yourself into the representative of a much larger community. I mean, and if you're in Microsoft you're thinking about how 100 million people or half a billion people are going to use what you do.

Kossow: That's the problem that all operating system architects have, that you have to project yourself out. How is the system going to be used?

Thacker: Right, and we knew because we were the users, so we were in a fortunate position, which is very hard to do today. But, let's see, where was I, wandering around. The machines worked. We basically started. We built a follow-on. We built an Alto II. I didn't have too much to do with that, actually. I was moving on to other things by that time.

Kossow: And so it was John Ellenby.

Thacker: That was John Ellenby. Yeah, John Ellenby actually did the engineering and he brought on board a team in Los Angeles called the special products group that had done engineering within SDS, and so they designed the Alto II. Fairly nice implementation. By that time, we thought the Alto was getting a little old and, you know, it was now maybe three years old from start.

Kossow: So, you were sort of sucked into the office system, and the D0 then?

Thacker: Not immediately, no. The thing that came first was the Dorado [ph?].

Kossow: Oh, really?

Thacker: Yeah, because we thought, I had actually been doing some work on caching and I realized that caches were really going to make a big difference in computing performance. And so I thought, and also, you know, ECL Technology, emitter coupled logic, had become relatively inexpensive, and so I thought maybe it would be possible using ECL technology to make a Nova-like machine, or an Alto like machine that could run with a 40-nanosecond clock rate, which would be unprecedented. I mean, that's a really fast machine, particularly in a personal machine. And it would have a cache, and a little bit later we decided it should have a virtual memory system as well because we were beginning to push up against the limits of the standard problem that architects always have, not providing enough address bits.

Kossow: Then there was the whole Mesa--

Thacker: And the Mesa Development, Mesa needed more space and, you know, we were just getting more sophisticated. And so we started working on the machine and it evolved over a fairly long period. This was not a fast track project, it took several years to actually get it working, and it had its up and downs, because at that time some of the members of CSL had gone off and SSL primarily, had gone off and formed actually a business division within Xerox called the system development division. This was run by Dave Liddle and Dave said, "How would you like to come over to SDD and help us by designing a machine that we can use for an office system that we can sell?" And I said, "Well, the only thing I have at hand is the Dorado." And he said, "Well, that might be okay." So, I went over to SDD and I was there a for a couple of years, but it turned out that the Dorado was way too expensive for general consumption at the time, so we investigated a little bit using it as a controller for very high speed laser printers, because those things were actually beginning to be made by Xerox commercially at that time. They were just getting into the business and as usual, they did a real good job at the high end first with a thing called the 9200, which was based on the upper echelon of Xerox copy machines. And that would have required an extremely fast controller but they decided that the 9700 should be operated with a controller that was more specialized, more custom. And so the Dorado was out of that picture, and so that was when the D0 started. The first idea with the D0 was to build a machine that was much more Alto like, use TTL technology, rather than ECL, which could be built relatively inexpensively with the technology of the time. And so Brian Rosen and I, Brian had come to work for us from CMU, I believe, designed that over a period of probably six, eight months.

Kossow: So, did you split the design between Brian and yourself?

Thacker: Yeah, I don't remember exactly how we split it up because I remember I did the memory system and I probably did part of the processor, and Brian did most of the IO.

Kossow: D0 is weird because it supported multiple displays.

Thacker: Yeah, it did a lot of interesting things.

Kossow: And the boat anchor, the three-wire interface.

Thacker: Well, we wanted a machine that would run the CAD systems. We were beginning to get into LSI in part of the Mead & Conway thing. I mean, Lynn Conway was down the hall in SSL working on the book with Carver. And so one of the things we wanted was the ability to run a graphics editor that you could use for chip layout, and that requires color. So, the D0 actually had the first color display that we built which ran a lower resolution color monitor, but it was okay for that purpose. But the D0 was never really a commercial success, although there were a few of them sold as LISP workstations.

Kossow: The 1100?

Thacker: The 1100, just as a few Dorados were sold as LISP workstations.

Kossow: 1132.

Thacker: 1132, right, but the problem was those machines were really too expensive and they were beginning to be, the performances that you could achieve with a thing like that were beginning to be achievable with micro electronics.

Kossow: The D0 was odd. It doesn't have any removable storage either. It just had the Winchester, so you had to have it on a network.

Thacker: Well, we had networks so that wasn't actually a problem. And so we built a few D0s and, you know, we used them within CSL, but the Dorado came along so quickly in such a short time after, because when we had stopped working on the Dorado and SDD, the project went back into CSL. At which point Butler and several other people kept working on it, and in fact morphed it into something quite different than we had originally designed, and wound up with something that was actually quite nice in terms of its characteristics. I mean, you could read the reports and--

Kossow: Yeah, I was just looking down the list of names. Ed is on there.

Thacker: Yeah, Gene McDaniel, Ed Fiala, Severo Ornstein worked on it. Quite a few people worked on it.

Kossow: It was a big machine.

Thacker: It was a big machine.

Kossow: Loud too.

Thacker: It is not the loudest computer that I have ever worked on, but it's close. So, let's see, right, so I moved back into CSL along with the D0 and CSL then built a number of other small machines sort of branching out from that. They built the Dandelion, which was used finally as the basis for the Star Workstation.

Kossow: So, that initially was a CSL design?

Thacker: Yeah, Butler designed it. It used a somewhat different form of tasking than the Alto did, but it was pretty simple. It also used a bit slice processor available from AMD. So, it was much more economic. So, that was the Dandelion. There was a machine built in SSL, or no, not in SSL. It was built by Dave Boggs called the Dicentra, the purpose of which was to provide a D-class machine. By that time, we actually had a name for the class, the D machines, which had a standard IO bus, so you could run standard peripherals.

Kossow: Do you remember if the three-megabit Ethernet at the time, came out of Stanford? Or did that come out of-- ?

Thacker: The three megabit Ethernet? No, it came out of--

Kossow: For the Dicentra, the multibus?

Thacker: No, I don't remember.

Kossow: It shows up at the Sun workstation, so I wasn't sure who did it.

Thacker: Well, a lot of ideas that came out of PARC showed up in the Sun workstations, it turns out. I remember visiting with Andy Bechtolsheim in the top floor of Margaret Jacks Hall, because they actually wanted to buy displays from us, so they didn't have to make a display of their own for the Sun-1. The problem was, we couldn't figure out how to sell it.

Kossow: In that time frame, you're into kind of the world starts hearing about it. There's the university grant program.

Thacker: Well, we had done the university grants, which were extremely successful.

Kossow: That happened in '79?

Thacker: Something like '79.

Kossow: So, it MIT, the University of Rochester, CMU, and Stanford?

Thacker: And Stanford, right. And they got a half dozen, or eight, or something Altos and a Dover printer. And of course that's what they wanted. They all wanted the printer.

Kossow: Because at that point in the 1980s, all those institutions start distributing their papers.

Thacker: Right, and in fact there were Altos as a probe by a little organization that Jerry Elkind had started. I can't remember what he called it, but Charles was there, Charles Simonyi [ph?]. They put Altos in the Carter White House and they were fairly successful too, you know, for electronic mail and document preparation. Of course, the White House produced a lot of documents.

Kossow: And then there's the GIN publishing.

Thacker: There was a thing done at Ginn called Gypsy and again, a lot of that was leveraged by the existence of laser printers, which were primarily, you know, designed by Starkweather and Sproul, and Butler to some extent, and I helped. So, yeah, by now the 70s are winding down and we're moving into the 80s, and we began to look around for something really new to do. And Taylor had gotten into a tiff with the Xerox management because he didn't think we were being well enough supported, and wound up in a lot of us departing. And quite a few people did stay on, and of course PARC kept going until Xerox recently spun it off, I think. Excuse me, I'm going to grab some water. So, in '83 a few of us left and looked around for something to do, talked to several companies about setting up a lab for them and finally wound up setting up the lab for DEC. We went back East and talked to Sam Fuller, and Bill Strecker, who was sort of the CEO of DEC at the time, and I think Gordon Bell, and they finally decided that it would be a good idea to have a research lab. They hadn't done much of that in the past and then they had gotten the Western Research Lab, which had been started up by Forrest Basket well before, about a year before. But we wanted to start another lab and it was a little bit confusing because WRL [ph?] was just down the road, and why weren't they the same, and well it because of the research management styles of Forrest and Bob were quite different.

Kossow: So, had DEC started up with Cutler and those people by then, or is that later?

Thacker: That's later. That's quite a bit later. That's '89.

Kossow: So, at this point, Forrest and Smokey are doing Titan?

Thacker: Neil Wilhelm.

Kossow: Neil Wilhelm?

Thacker: Yeah.

Kossow: Okay, so Titan is going on.

Thacker: Titan is being built now and we had, in the last couple years, been looking very closely at multiprocessors, because that seemed like a good way to sort of get a step in performance. Which, if you did it right, it was a sustainable thing. You know, Moore's laws carrying you along this curve, and you step up to a difference curve.

Kossow: So, McCreight stays behind and does Dragon?

Thacker: That's right, he stayed behind and elaborated, and finally implemented, along with Sun in some way that I could never quite fathom.

Kossow: Robert Garner is over there.

Thacker: Right, Bob Gardner, the Dragon, which was a machine that I had started the project and named it, but the main thing was the idea of exploring the effect of multiprocessors. But then we left, and so some of them stayed and continued it, and by that time I was real busy starting up SRC, and so I don't know much about what happened to it. You'd have to talk to those guys. So, at SRC we actually decided before we started it what our first sort of software environment would be like, and hardware. And on that basis, Sam Fuller and Butler, and Bob and I sat around in a meeting room at the Concord Inn, in Concord, and sort of planned out the first year of SRC. And then we went back to the West Coast and did it, got the building on the corner of Lytton and Alma and began to hire people, not just people from PARC, although we would, but we never approached them. I mean, we didn't really recruit our colleagues from PARC, because Xerox told us that if we did, dire results would--

Kossow: Ed Satterthwaite comes over then, or was he already-- ?

Thacker: So, Ed came over and Butler came over. Butler actually came over a little later. But by that time, Butler was living in Philadelphia because his wife had a position there at UO P, just as he now lives in Boston because she has a position there. So, he was a commuting member of the staff. Let's see, Charles had gone to Microsoft by that time, and who else moved here immediately? Well, Phil Pettit joined us right about that time. I have a photograph of the first few people, Roy Levin, Jim Horning, Mike Schroeder were all part of the early crew. Boy, I wish I remembered exactly who's on that photograph, because it's on my sort of kitchen wall. I should. So, we began to do the Firefly, which was an up to six-way multiprocessor, initially using 68000. But before we built them, we switched over to the MicroVAX because it seemed like it would be a better thing. That was a new chip processor from DEC that was a single chip fax.

Kossow: The MicroVAX II, the one that DEC worked on?

Thacker: Well, it was MicroVAX actually. The MicroVAX II was a little bit later. But, in fact, I think the MicroVAX II was the one that actually went into the machines, you're right.

Kossow: But you ended up doing a lot work on snooping protocols than on the-- ?

Thacker: That's right, we did. Actually, it's interesting. We have some of the earliest patents on snooping protocols, but nobody knows it because we didn't write a whole lot of papers about it. But in any case, that came up and along with it came a programming environment that allowed you to program the multiprocessor, and do a good job at writing programs with threads, which is something that programmers today still don't do. But in 1985, roughly, we had a full-bore operating system that used threading very extensively and ran quite well on six processors, thank you.

Kossow: The environment is interesting, because Niklaus Wirth does his sabbatical, goes off, builds Lilith, does Modula 2, and then you end up using Modula 2.

Thacker: Well, we, of course, knew Niklaus very well and still do. We vacationed together in France last year.

Kossow: These interesting things, like Rosen does the PERQ.

Thacker: Right, yeah. Brian actually went off to do the PERQ deliberately on Three Rivers , and basing it a lot on the D0, although he had the much nicer display because he was working with a guy whose name I do not remember. I never met any of them who had designed this marvelous 600-line resolution, no 300 lines per inch resolution display, beautiful. Put up a drawing of it and it was marvelous. But in any case, we got that machine working and then we used it quite successfully. I don't think we built very many extensions to it, and then DEC began to think seriously about the Alpha. And actually--

Kossow: The PRISM/Alpha Wars?

Thacker: Yeah, the PRISM/Alpha Wars happened at that time, but a little before that, we essentially changed our focus. We were no longer solely interested in computers and we started getting interested in networks, because it was pretty clear-- it was not clear then that the Ethernet, which was then over ten years old, was going to survive the way it has, and it has incredible longevity. In fact, it's really like George Washington's axe because it's nothing like the original, it's just the name and the idea of a 1500-byte packet. But, we explored some networks that were pretty interesting for their times. We built a network that we first called Autonet until the company that ran some network for selling used cars threatened to sue us. And that had some pretty interesting properties as a local area network, because the network switches were actually active. They had programs in them and the thing could organize itself to be self-healing when something broke. So, all the workstations were dual homed to two switch ports, and if a switch broke, the entire network would recognize that and reorganize itself into a potentially new spanning tree that allowed you to route messages from everywhere to everywhere else. And it was a switched network, so it was actually quite fast. The signaling was actually done in a slightly weird way. It was done bidirectionally on a 75-ohm coaxial cable, which turned out to be kind of a bad idea, but I was always trying to do interesting electrical engineering things. But, that network worked out very well, and so we, at that time there was a lot of discussion about, you know, what's long haul networking going to look like? Is it going to be faster Ethernet? Is it going to be ATM? Is it going to be a network called FDDI, which DEC was pushing, for fiber distributed data interconnect. And we decided we liked ATM networking because it was pretty simple, and we decided to build an ATM switch. And so we built a switch that would switch 16 622 megabit per second ports, and this was a big crossbar connected with my very competent line cards that had fiber optic interfaces on them. Fairly substantial piece of engineering, all controlled by a small control processor for the switch just to set it up, and the fabric was actually scheduled by microcode. It ran in a Xilinx Chip and set up the crossbar at one ATM cell intervals, and it did the necessary calculation to discover how to configure the 16 x 16 crossbar to most efficiently route the messages between the input ports and the output ports. And that was kind of nice, but I still remember inventing the algorithm because I invented an algorithm to do that while lying on the beach in Hawaii. But the problem was, I couldn't quite figure out how rapidly the process would converge. So, I went back and I talked to one of the theorists, and he said, "Well, I don't know, you know, this is a matter of bipartite graph-matching and that's NP-complete." And I said, "Well, I don't care about getting a perfect solution, I just would like to get a good solution. So, does this do that and how fast does this converge?" And he came back a couple of days later and he said, "Oh, the proof was trivial. It converges in four cycles." So, you can run this algorithm for four clock ticks and then you have your switch schedule, and this was subsequently elaborated by a guy named Nick McKeown who's now a Stanford professor, into an algorithm that was used to build a terabit router. So, I was quite pleased at that. So, we built this thing and it eventually worked, and we used it, replacing by and large the AN1 network, and we participated in a test bed that was being run by Sprint around the Bay Area. They actually put fiber into our buildings and we got a lot of traffic statistics on that network, and one of the things we did, that we were actually pushing quite heavily was ATM does not use flow control. And as a result, you have to be very careful about arranging connections through the network, because otherwise you can lose packets. And I have always believed that loosing packets is a sin. The user, after all, the customer paid you to deliver those packets and dropping them on the floor is sort of a violation of the trust between you and your customer. And so, we actually worked out a scheme for doing credit based flow control along a link that meant that the network never dropped packets, I mean, unless there was an error. We actually tried to sell this to the internet community and they weren't buying. They believed in TCP, you know. Congestion control is signaled by loss, and of course that's not true at all. Congestion control should be because of congestion, not because of loss. Loss is because of errors. So, in any case that thing actually became a DEC product. It was called the Gigaswitch/ATM, but at that time there was a lot of interest in networking and which protocols, and, you know, what underlying technology would win out. A lot of different contenders, and of course, Ethernet just came in and swept the field. And so, it all looks kind of quaint from the 21^{st} Century to look back at that time, which was by now the late 80s and see the sort of discussions that went on in the various communities that were interested in these issues. Because, of course, the Internet was going by that time, haltingly, but going, and so, you know, this now because a question, it had a lot of potential economic value. And it was no longer just a lot of people sitting around in IETF working group. So, in any case, that was our little foray into networking and then we built another computer, because Bob Supnik, who was at that time the project manager for the overall Alpha program, including the production of the chips and the generation of the necessary software, came to me and said, "You know, Chuck, I've got a problem. The programmers aren't programming because they don't have a machine to program and they never write code for simulators. And so we are going to wind up in the enviable position of having our first products and not being able to sell them for a year. How can you help mitigate this?" And I said, "Well, you know, maybe we can do this," because they had been building a chip that was never actually sold, called the EV3, which was a test chip for the Alpha, which included a fully functioning processor, but very, very small caches and it had the desirable property that its IO pins could be either ECL or TTL, or CMOS interface levels. So, we took that thing and we built a machine around it, which was actually another one of those--

Kossow: Larry Stewart

Thacker: This was Larry Stewart, Dave Conroy, and myself.

Kossow: So, Dave [Conroy] is out here by that point?

Thacker: Dave came to work for us. He was thinking of leaving DEC and we talked him into coming to the West Coast instead Similarly with Dick Sites who was in roughly the same boat. I'm glad we did. So, we looked at building this machine and we built a machine called the ADU, which was the nosiest machine I ever made. This machine was built on printed circuit [ph?] boards that were about this big and a board was either a memory board, or a processor board, or an IO board. And we decided very early on not to build a lot of IO devices, but instead to build a high-speed channel off to a VAX that could run the IO devices for us, just the way we ran the tapes on the BCC-500. So, we designed this 50 megabyte per second channel, and Larry Stewart designed that board, Conroy designed the CPU board, which included the Alphas and its caches, and I designed the memory system, and sort of architected the overall thing. And it was a stellar success. It really worked. We built about 50 of them, distributed them around the company and the programmers started programming them because they could, and Supnik has told me subsequently that he believes that the existence of that machine cut one year off the ship time of the Alpha. Which sort of in the long overview of history is probably not very important, since it's dead now, but there you are.

Kossow: Let's take a break for a while.

Kossow: It's August 29, 2007, this is tape number 3 of the oral history of Chuck Thacker. So we're just to the point where DEC is about to come out with the Alpha so you're doing prototyping machines at SRC.

Thacker: Right. So we built these ADU machines. We actually wrote some nice papers about that, which appeared in ACM, communications of the ACM in '93. But we never built very many of them for our own use because they were very expensive. ECL is an expensive technology. They actually used, they had a lot of memory by the metrics of their time, which was very expensive in those days, and we really couldn't justify the use of those things for research.

Kossow: So most of the research machines at that point at SRC are UNIX machines?

Thacker: They're uniformly UNIX machines, yeah. DEC, they're standard DEC products running UNIX with, you know, somewhat larger DEC machines in the basement in the machine room.

Kossow: So sort of the last gasp of the Fireflies was early 90s or?

Thacker: Yeah, '93, '94. Some people still kept them running, but not for any good reason.

Kossow: We have a couple of them in the collection.

Thacker: Right.

Kossow: So let's see, so when the ADU wound down—So did SRC end up getting involved in the PRISM alpha wars?

Thacker: A little bit. A little bit. We were involved in another war--

Kossow: Do you want to talk about the background at all?

Thacker: Well, I can in a second but yeah, the earlier war was a thing called the Bird Wars, where we were trying to decide whether to build product machines that were representative of the machines that were being designed in Marlboro, which was the VAX 9000 or the machines that were being designed in Littleton, which the VAX 8200 is an example, a much kind of sleeker machine. And DEC was just about to take an enormous beating on the VAX 9000 because they didn't work fundamentally. And they were dominated by the lower end VAXs in CMOS silicon. Dan Dobberpuhl had finally you know, worked out how to build an extremely fast VAX in a chip. Which seemed fairly magical at the time but of course it's ho-hum today.

Kossow: Right, and then Dobberpuhl goes on to do lots of other stuff.

Thacker: Oh, he's done lots of other things. He did the StrongARM, he went and founded the company to do PowerPCs called CyByte, which was an insult to Broadcomm. He started a company called Palo Alto Semiconductor which he's at now, producing, I beg your pardon, the Broadcomm thing was a MIPS, maybe it was a MIPS because PowerPC is being done by the PA Semi .

Kossow: Right, so it was Where Satterthwaite goes?

Thacker: And that's where Ed went, it's where a guy named Mark Hayter that was interesting guy who joined us out of Cambridge, joined SRC. Those are the two primary people, although several people from the Hudson Enterprise on the East coast joined up with Dan. But,

Kossow: So you have the same us verses—

Thacker: Right, so the Bird Wars, I was the judge in the Bird Wars, and I finally said, you know, these mainframes don't make much sense for the same reason that they had made no sense for SDS, you know, IBM is just, if you want to compete with IBM you've got to have the IBM software. We didn't, and so that coupled with the fact that the machines were vastly expensive and fundamentally didn't work because we used this weird packaging technology.

Kossow: So you said the Bird Wars?

Thacker: Bird Wars.

Kossow: So which were the Bird Wars?

Thacker: Raven and Eagle.

Kossow: Ah.

Thacker: So in any case, later when we were doing the Alpha, there was a lot of discussion about the operating system that Dave (Cutler) was working on for it and the machine that ought to run it. And Dave wanted the operating system PILLAR to into in a machine whose architecture was called PRISM.

Kossow: This is Dave Cutler?

Thacker: Dave Cutler, right. And Dave was the primary software architect of the VAX family and very highly respected guy in Microsoft.

Kossow: Still at Microsoft?

Thacker: Now within Microsoft, still at Microsoft, still highly respected. He's a senior technical fellow, one of the few people above me. In any case, he had set up this place in Washington called DEC West in Seattle in fact and Redmond, and when the company finally decided that it was not going to pursue that, he decided to leave and go to Microsoft. And that worked out very well for him. But, so we moved forward on the Alpha. At that time, I don't remember, I guess we were, actually we were building the AN2 network at that time. And then that wound down and I was looking around for some other things to do and by this time it's '96 and Microsoft had been talking to me about joining them for awhile, since about '91. I'd always believe that there was probably no place for me because they're not a hardware company and I'm primarily a hardware person. Although, I have done a lot of software. I know how to do it and all of that.

Kossow: When did Butler end up going over?

Thacker: Butler went about a year and a half, two years before I left. And so he was highly instrumental in recruiting me, as was Jim Gray, who had been there for awhile. And, of course, Rick Rashid I knew very well because one of Rashid's early claims to fame was Alto Trek .

Kosssow: Right.

Thacker: Which has become sort of a cult classic in the field. And so I had talked to Rick and to Nathan Myhrvold and to Butler and to Jim and ultimately to Bill and, what I was doing was talking, trying to talk them into setting up a lab in the Bay Area. And they decided they didn't want to do that because they were worried, Microsoft was at that time, and today even still is a bit, very ,Redmond-centric All the development is, most of the development is done in that area.

Kossow: So this is 1996?

Thacker: Late 1996.

Kossow: Okay, so there's the Mac products and had Web TV started? So there's a few things going on.

Thacker: I don't think Microsoft had bought them by that time but--

Kossow: Yeah.

Thacker: But they had started. And so, finally, this must have been in the late summer of 1996 because, yeah because Roger Needham, who had been visiting us for a month a year for 20 years roughly, and I were both talking to Microsoft at the same time. And what would transpire is some Microsoft people would fly down from the north and rent a hotel suite in San Francisco and in the morning I'd go up to talk to them about a lab in the Bay area and in the afternoon Roger would talk to them about a lab in Cambridge.

Kossow: This is Cambridge, England?

Thacker: Yes, and we would, you know, cross paths through the door and then, and then Roger would come back to SRC where he was based and we would compare notes. But finally the company decided that it didn't want to set up a lab in the Bay Area so I began to look around for other things to do because I knew that, I knew that DEC, the writing was on the wall for DEC and I was just looking around but, you know, by that time--

Kossow: Ed had gone off to General Magic and--

Thacker: Yes. By that time my kids were grown and gone and my wife had retired and so we had very little holding us and we said let's just take a year off. Let's do a sabbatical somewhere, because we've never done that, you know, let's either go to either to ETH and do a sabbatical with Klaus Wirth (ph?) or go to the Cambridge Computer Laboratory and do a sabbatical there. And she, thought as that might be an interesting idea, was a little nervous about Cambridge but--

Kossow: Had she spent much, or had you spent much time in England?

Thacker: No as a matter of fact, I had but she hadn't. And so one day in the early Winter I think of '96, I heard a rumor that Roger's lab was actually going to come to pass, they'd continued to talk after I dropped out of that discussion and one day I got a call on the phone, I know, what happened was I heard about this and I sent Roger an email at the computer lab congratulating him and asking him what was going on? Roger is always very cryptic in his email responses and he sent back something that said, "Well, yes, it looks fairly positive and you'll hear more about this soon." Thirty minutes later the telephone rang and it was Nathan Myhrvold and he said, and he said, "You know we've been talking about this and one of the things that you know a lot about how to do is to set up a research lab, how would you like to move to Cambridge for two years and help Roger set up the lab, because he's never actually done that and you've done two?" So I said "Well, that sounds pretty good, let me talk to my wife and I'll call you back in a few minutes." And I called my wife and I said, "Love, how would you like to spend a couple of years and live in Cambridge and I will start, I will help Roger start a lab for Microsoft?" No, I didn't say that, and help start a lab there, and she said "No, I don't like Boston. It's too cold." And I said, "No, the other one." And she said, "Oh, well, that might be interesting." And so I called back Nathan, I called Nathan back and I said "Sure, I'll do it." And that was how the whole thing, my participation in the whole thing got started and it was a two year assignment, so I went over there and helped set it up, help do a lot of the initial hiring and set up the you know, policies you need. Industrial labs are a little different than academic places and so you know, you have to work with things like the Human Resources Department and all of that. And Lord, with the Public Relations people and Press people because of course, Microsoft going to Cambridge was a big deal in the British mind, they thought. They actually thought that we were going to drive the house prices in Cambridge through the roof. Now they have subsequently gone through the roof but it's not our doing. It's just that Cambridge has flourished, as a center of technology and innovation. But I remember talking to a reporter from Cambridge Evening News and he said, "Well you guys are going to drive up the prices because all you Americans," and I said "Wait a minute, this is not an American lab, it's a University, this is an European lab, I'm the only American here and I'm gone in two years." So, right, we were also dunned when McDonalds set up a separate, a second facility in Cambridge, thought it was for us. And I said "No, we eat at the pubs just like you do." But in any case, that was a very pleasant and I think quite productive time. I didn't actually do a whole lot of technical work during that time. But I did some. I worked a bit on electronic books, how to organize an electronic book and how to build one. And that was where some of the early thinking for our Tablet PC was done. So when my wife and I moved back at the end of the two years, we had decided that we really didn't want to live in Redmond because some people are affected by the disparity between the days and the nights and the various seasons and it bothered my wife quite badly. So, I said to Rick Rashid, "Well, I'm not going to move to Redmond, am I still working for Microsoft?" And he said "Oh, sure, you know, we'll find something to do."

Kossow: But by that time Gordon [Bell] is working down here isn't he?

Thacker: By that time, Gordon may have been here but he was not, I don't know what the deal was. Jim had his thing in San Francisco at that time, but Jim really didn't want to grow his, his organization that was just an office for him, because he lived in San Francisco. When Gordon moved to San Francisco he started working for Jim. So, yeah that is about the same time. But instead what I did was I talk to the electronic book people and I had been working with them a bit on some of their prototyping work and when the guy who was running the group got the charter to build a Tablet PC, I went to him and I said "Why don't you hire me because I know how to do this, because we built a Tablet PC at SRC in 1993." And so, Mark Hayter had done that. Mark Hayter and Satterthwaite. And, yeah, and then, and Andrew Birrell as well. Andrew Birrell had bid it, he was in that photograph that I mentioned. So, so I got embroiled in the Tablet PC for about two years.

Kossow: Okay, and that's the time when you and Butler gave the talk on the Alto for the museum?

Thacker: That is correct. Yeah.

Kossow: Yeah, I remember you were working on that at the time.

Thacker: Right, and that became sort of a main line thing in the industry actually. I mean Tablets are still, the Tablet PC has not been the kind of commercial success that the laptop has but it's gratifying that it's the only such machine in that ilk that has been in the market for six years now. And it still survived, I mean, all the earlier ones died. So more recently let's see, what have I been doing? So I went back to Microsoft research when I finished all that up, I worked on a couple of other little projects in an incubation group run by Craig Mundie, who's now the CTO, CTO then for that matter. But when there began to be a discussion within MSR about actually doing computer architecture research, I decided to rejoin the organization because of course, that's what I do. Or what I did for a long time. And I've been there now for almost, a little more than a year and a half, yeah. And so we're setting up a group, we're doing some things; we're doing the thing that I mentioned to you.

Kossow: Right, with Patterson.

Thacker: With Patterson.

Kossow: So, are you going to retire next year?

Thacker: Probably not, I mean, I find myself still enjoying going to work and I figure if you can that why retire?

Kossow: Are you working over here? Or do you have a—

Thacker: Yes, I'm two blocks away from your Museum. And in fact I'll be a little closer because shortly we will move into this building on Pear Avenue because the Mountain View campus is filling up and we're growing.

Kossow: Were you involved in the Xbox at all or any of the other--

Thacker: Yeah, I actually was a bit, I helped.

Kossow: Dave Conroy had worked on that a little bit.

Thacker: Dave actually worked on hardware for the group that became the Xbox group. But by the time the Xbox started, he was no longer in that group, he was working with me. And then he decided to go off to Apple, so he's doing that and apparently happy as a clam.

Kossow: Yeah, my office was two offices down from him.

Thacker: Well, Dave does love to build things and of course he had--

Kossow: Dave is very good. He's extremely good.

Thacker: Dave is one of the few people I know that's actually a better engineer than I am. Because he's careful, he's also extremely smart, we have complimentary skills in many ways, but there are things he can do that I, you know, can't. And I'm sure the same thing is true about the other direction.

Kossow: He's very smart, he's very fast.

Thacker: That's right. Now I enjoyed working with Dave. As I said, I hired him at SRC, we worked together for almost 15 years and, so in any case, I've just been doing this and I probably will not retire, I mean, I might. I threaten to from time to time but if I did I would go off and find something else sort of to do in the technology area, I don't know what it would be but.

Kossow: Yeah, it's hard to do systems things now.

Thacker: Well, that's one of the reasons that I'm having fun these days because I believe this RAMP project actually has a hope of revitalizing system architecture research because if you look at the papers that's been published over the last 15 or 20 years, they're all very incremental in the architecturearea and there don't seem to be many breakthroughs and the reason is, is it's just very hard to build systems that are very different from what's in the main line because you can't build chips any more. And so the, one of the key ideas behind the Ramp program is that FPGAs are now big enough that you can't build something that is both as good, is as good as something you can implement in silicon. But you can actually build a credible emulation of a big system that runs at a sufficiently high speed that it will run real software. And therefore, you can do a class of experimentation that you just simply couldn't do without machines like that. And, so I'm quite enthusiastic about it as I mentioned earlier, I will be, I'm working with them on getting their next machine going and--

Kossow: So how many software people do you have working with you?

Thacker: Oh, the lab is mostly software people except for me, I mean, we're trying to hire more architecture people, we have hired one very good person from Stanford and another sort of co-conspirator who came to Microsoft after getting a PhD, after being essentially the CTO of SGI. And so the three of us are trying to attract more architecture people but that's a slow process because our standards are very high and, so what else can I tell you?

Kossow: Oh well, there are a few questions that they wanted me to ask.

Thacker: Okay, sure.

Kossow: While we grind through those. What were your most important life lessons?

Thacker: Life lessons? I would say it's always a good idea to follow Einstein's dictum and make thing as simple as possible. But no simpler.

Kossow: And what was your proudest moment?

Thacker: Oh, I don't know. I could think of a number of things, you know, birth of my first child, although I didn't have nearly as much to do with that as my wife did. You know, any of the nice awards, you know, but—

Kossow: So what might you have done?

Thacker: What might I have done? If I hadn't chosen the path that I did? I would have probably have been a retired fairly mediocre Physicist.

Kossow: So, looking back, what do you see as you're most important contribution?

Thacker: I have tried to figure that out and I can't really do it because there have been several different things that I've actually been quite proud of. I've been extremely proud of the Alto of course. Because it basically launched a whole line of, line of stuff that we are still developing today. And I'm quite proud of some of the larger machines just because they demonstrate that if you do careful engineering, and keep it simple, you can build things that are very high performance. I'm proud of the Tablet PC because I like eventually it will be, it will take over and all PC's will be tablet-like. That might be very slow, it will require a lot better power consumption, and a lot of sort of attention to the low end in power which of course is now one of the problems that is facing us as an industry. We're not going to be able to go faster because we cannot get the power out of the chips anymore. And so we're going to have to figure out ways to deal with that and I think some of those things are pretty exciting because it's going to require looking at systems in slightly different ways than we've been doing for the last 20-30 years, is to say we've been quite incremental. So now we have to do a lot more thinking out of the box.

Kossow: So what do you think of the convergent devices or you know, mobile phones and that kind of thing?

Thacker: I think they'd be nice if they worked. I haven't seen any that I really like. Usually people who try to build fancy phones forget that a killer application for a telephone is making phone calls and also, it's very hard to have a sort of convergent device world when the market is controlled so heavily by the carriers. This is something people frequently don't realize, which is that you can't walk up to anybody's cellular network with a different phone and necessarily run it. So they control very carefully what goes on their network and that while that is happening it's a some what different thing than happened in other sort of computer related industries. I'm not quite sure how it will play out but we'll see.

Kossow: What advise do you have for current and future generations?

Thacker: Well, that's an interesting question. I'm very worried about the fact that fewer and fewer kids are being, are interested in technology. And I've tried to figure out ways to help with that but I don't know, I don't know exactly what to do. I mean we don't have the very, we almost don't have you know, the ham radio operators and the radio experimenters that happened when I grew up. You can no longer buy Heathkits. So what is going to be the motivator for young people? And of course for awhile, people thought it would be the PC. But once you've written a few Basic programs, what then? You know and you've manipulated the sprites on the screen and written your little video game, what then? Well, one possible answer is Robotics. Allowing a kid to build something that actually does something is a pretty interesting thing. And I know Lego has been relatively successful about, with that. We're actually making a robotics tool kit to try to make some of that simpler, but I don't know, we need a sort of a killer app. for sucking people into technology and of course when I grew up, when I was in, Sputnik happened when I was in high school and of course that was a galvanizing moment for the nation. And so a lot of people who were deciding on what to do made a decision to you know, to basically go out there and beat the Russians. We haven't had anything like that. So,--

Kossow: Well, there have been galvanizing things that have happened but they haven't been positive.

Thacker: Well, but they haven't been, it's not the same kind of thing. It's not something that you would actually aspire to.

Kossow: Well, this has been fun.

Thacker: Yeah, thank you, my pleasure.

END OF INTERVIEW