Oral History of Alan Kotok

Interviewed by:
Gardner Hendrie

Recorded: November 15, 2004
Cambridge, Massachusetts

Total Running Time: 2:36:36

CHM Reference number: X3004.2005

Q: Today we have Alan Kotok, who's agreed to an oral history for the computer history museum, and its Oral History program. Maybe Alan, you might start off-- I think it's always interesting to understand a little bit about sort of where you were born, and your family.

Alan Kotok: Okay.

Q: And the first recollections of what you thought you might want to do when you grow up whenever those occurred.

Alan Kotok: Okay. I was born in Philadelphia. My family lived in Southern New Jersey. That just happened to be where my mother went to give birth. My father was in the retail hardware business, a family store. And I was always interested in tinkering around in the store with tools and various things. I got into model railroading, because they sold toy trains at Christmastime, and from sort of an early age I was always involved in setting up the Christmas displays. There's a family legend, which I don't personally recall, that my engineering career began at a tender age when I stuck a screwdriver into an electric outlet, and managed to put a notch in the screwdriver while I'm trying to pry it out, as well sending myself to the other side of the room. So, as I say, this is a family legend. There's this screwdriver still around with this notch in it, but it looks like it might be appropriate age. So I guess I would say that I was always interested in electrical engineering. And I used-- when I was a kid, I used to build little circuits with batteries, battery-powered circuits, and I got-- wasn't really into ham radio, but short-wave radio listening, and that's SWRL, or something like that. And so my desk was always cluttered with gadgetry of various sorts. And so the thought I would go into electrical engineering sort of came from when I was five or six years old, I think that was sort of evident.

Q: Did you do any work on controls for the model railroad, or anything like that?

Alan Kotok: Not before I came to MIT. I mean, when I was in high school, I was in various science clubs, and other such things. And I think I recall maybe building a few relay gadgets and things of that sort. I first got interested in computers in high school when-- this was in the mid-1950s.

Q: Now what high school was this?

Alan Kotok: This was the Vineland High School. Vineland, New Jersey, right, yes.

Q: I grew up in Philadelphia, so...

Alan Kotok: Oh, okay, so you know the area.

Q: Yes.

Alan Kotok: Yeah, so the science club, or one of the various science clubs, went to visit the SOCONY [ph?] Mobil Research Lab in Paulsboro, New Jersey. And they had some, you know, giant thinking machine of the day, which is now-- this is the day, it's like 1956 or '57, so. And the visit included someone, rather than saying, "Here is our giant thinking machine, isn't it wonderful?" Actually took the class, the group, through a programming exercise, where we programmed, I don't know, some converging series problem or something like that, and we punched up the cards, and we put them into the machine, and you know, the printer clank-clanked and we got the answers. And I get a little bit emotional about this sort of stuff. <laughs> But as sort of the spark that triggered me, and said, you know, "Computers! This is it!"

Q: Yeah, "This is fascinating!"

Alan Kotok: This is really fascinating! And so I then, you know, the library in my school and in town had sort of nothing on the subject, and so I would head up to Philadelphia on the bus on the weekends and go to the Free Library of Philadelphia, and the Franklin Institute, I guess, had a library or something, and you know, found books on computing, and sort of read up on it. And, excuse me. So anyway, I applied to MIT, and...

Q: Now did you do it? Did you get your hands on any computers?

Alan Kotok: I really didn't, no.

Q: You just read about them?

Alan Kotok: I just read about them, and sort of tried to learn as much as I could, given the lack of any resources. So I applied to MIT to matriculate in 1958. And I also applied to a few other schools and got in.

Q: Where else did you apply?

Alan Kotok: CalTech and somewhere else, I don't remember.

Q: Okay.

Alan Kotok: So, anyway, I showed up here in the Fall of '58, and immediately wound up at the MIT Model Railroad Club. Because I was interested in model railroads. And they had this very sophisticated-- at least at the time-- relay control system that helped operate the trains. And so <clears throat> I immediately got interested in that. And met Jack Dennis, who was a graduate student at the time. He's now an emeritus professor, I guess, here.

Q: Yes.

Alan Kotok: Who I still am friendly with. And he had access to the TX0 computer, which was the first transistorized computer. It was built at Lincoln Lab, and it had been moved over here to the Cambridge Campus, over in Building 26, sometime in the Fall of 1958. And he offered to teach programming on that to the people at the Model Railroad Club. So there was this group of us who went to a classroom nearby, and he-- talked about this, and the way that this lab that had the TX0 operated, was that it was just a sign-up for time. So as distinguished from the computation-centered model, where you had to submit tasks and decks of cards, and you'd come back literally, you know, a day or two later to get your results, you could actually sit at the console of this computer and it had a display, it had a typewriter, it had paper tape-- input/output was the primary means at that time-- but you could do your computing! And so he asked the management there whether it was okay if his friends, you know, these undergraduates used the computer, and they said, "Well, no one else has signed up. You can have it." So from the Fall of '58 I was, you know, in computing heaven, so to speak. <laughs> So yeah, so anyway, so that was one track. And then in the spring term of '59, John McCarthy first offered a class, a programming class for freshmen. And previous to that there was a junior level or something programming class. And so I, and again several of my friends, signed up for that. And it was focused on the IBM system in the computation center. Which I guess at the time was a 704 or some, you know, room-filling machine, vacuum tube machine. And I, you know, really enjoyed that, and went to McCarthy along with a few friends, and said, "You know, we'd like to do a project here. What do you have? Is there anything you could give us?" And so, you know, he said, "Well..."

Q: Yeah, you're still a freshman.

Alan Kotok: I'm still a freshman, right. And so he said, "Well, I've been working on this chess program. I haven't really gotten very far. You know, I have some move routines, but you know, if you guys would like to work on this, you know, you're welcome to." So there were three or four of us who picked up this project, like the Summer of '59, I guess it was. And we worked on it as a sort of an undergraduate research project for the next three years. And as it came time for me to graduate, there was a thesis requirement in the EE department. Bachelor thesis requirement at that time.

Q: Cool.

Alan Kotok: And among the group it was agreed that it was okay if I wrote the thesis.

Q: You could use this.

Alan Kotok: So, I could use this for my thesis. And so I did write that up. And then some time shortly thereafter, although I cannot recall exactly when, the body of the thesis got published as an artificial intelligence memo, from the Artificial Intelligence Lab, and that has resurfaced, but only sort of in nth generation copy form. So what I have here-- so I have a printed copy, which you can see is pretty awful, and I scan-converted it, and then edited like mad, until I got it into editable text, so there it is in HTML.

Q: Oh.

Alan Kotok: It's on my website. So.

Q: Oh, very good.

Alan Kotok: So anyway.

<crew talk>

Alan Kotok: So then I, you know, when you called, I said, "Well, so this makes me think, so where is the thesis and you know, it has-- there're various appendices that are referred to there with various results that don't appear in the memo, and so I did a library search, and so they do have the thesis in the archive somewhere. And I just signed up to have them scan it and turn it into .pdf, which will be delivered by email to me sometime in the next week or two. So.

Q: Okay, well, thank you very much!

Alan Kotok: So I'll be happy to supply that.,,

Q: Oh, that'd be wonderful!

Alan Kotok: ...to you, too.

Q: Excellent.

Alan Kotok: With luck, the original thesis is clear enough that scan-converting it back to text will work.

Q: Will work as it's supposed to,

Alan Kotok: Will work, most, you know, pretty well. And so I would plan to republish that in HTML.

Q: Okay. Excellent.

Alan Kotok: So anyway, we sort of jumped way ahead there, but the flow kind of led to that.

Q: Okay, that's good. Thank you very much! 'Cause you said those kind of documents archived?

Alan Kotok: Yes. Yeah, they're...

Q: All right. That will be wonderful. Good.

Alan Kotok: And if they, you know, you can have the machine readable versions of-- both of those are available.

Q: Okay, all right. Now maybe you could tell me a little bit about what you can remember about some of the trials and tribulations of the group as they were working on this program. You know, was there a general approach to how to do it that had already been outlined by McCarthy? Or what, whatever you can remember, I would be interested in any details.

Alan Kotok: Well, let's see. So at the time we started the state of the art in machine chess was very rudimentary. There was a program, I guess, Newell, Simon, and Shaw from Carnegie had written, I think it played on a reduced board without all the pieces. So, there was just a tremendous amount of work to do just to get all the rudimentary things you need to actually mechanize the playing of the game. Determining what moves were legal, trying to form some static evaluation functions of positions, and so forth and so on. And the state of computer languages in those days wasn't all that great either. McCarthy, of course, was a great promoter of Lisp, but Lisp in those days was fairly simple, and random, pretty slow. And he had started using FORTRAN, which was, of course, the primarily "higher level" language in the-- around 1960s or so.

Q: Certainly the language that's developed and available in 704, yeah.

Alan Kotok: So we started putting together, you know, just a whole lot of sub-routines. We soon concluded that FORTRAN had so many inefficiencies that at least the routines that ran very frequently were called, you know, thousands of times per move, had to be written in machine language, and so we dropped down to assembler language for quite a number of routines. As I recall, mostly to McCarthy's chagrin, one of the things that, again, a lot of this has been refreshed by reading this paper here, is that McCarthy proposed to us what is known as the alpha-beta heuristic, which you know, limits the necessity to search down directions which have already been demonstrated to be-- they're better alternatives and so forth.

Q: Yes, exactly.

Alan Kotok: And so we built that into the program. And then of course, along the line MIT replaced this computer twice, I believe. So we started on the 704, and then they went to the 709, and then a 7090, which were all nominally compatible machines, but not that things broke, but now there were new functions, there were new capabilities that by building them in would improve the program. So what I recall was that the program did not run very fast. You know, computing was in, you know, kilo instructions per second in those days.

Q: Yes, exactly.

Alan Kotok: And our project, we had this sort of informal race with the system software group. So everyone had a charge number. You had to charge your stuff to, and we tried to see if we could use more computing time on that machine than the system resource, the system support group, which was the other largest group. So for a couple years we were kind of the largest user of computer time.

Q: Even though a relatively small group of people.

Alan Kotok: Yes. And so we would come in, you know, in the middle of the night, after they had run their batch jobs for the day, or something like that, we would come in and take over the machine, because the chess program, you know, we would try things and see how it would work, and then try to tune the program a bit, and come back, and run more tests. So we burned a lot of midnight oil in those days.

Q: Oh, wow.

Alan Kotok: And oftentimes six or eight hours a day when we were ready to run.

Q: Yes. And by doing it in the middle of the night, you could get big chunks of time.

Alan Kotok: Right, we could get big chunks of time.

Q: As supposed to the "submit your card" bit during the day, and the operators will do it.

Alan Kotok: Right, yeah, the card deck approach to computing was-- well, we'd try to do like compilations and things like that with FORTRAN, the traditional thing is you'd submit this big deck of cards, and it would come back with a listing, "Program expects comma here. Abort." And so you'd lose another day! <laughs>

Q: Exactly. It slowly works its way down each day.

Alan Kotok: Yes, yes, right.

Q: Test, abort the program. Test for syntax errors.

Alan Kotok: Right, test for syntax errors. So but again, it was sort of someone's heady times of getting to use the giant thinking machine yourselves, and you know, the operator staff would sort of just go off and relax, and we would take over the console, the machine.

Q: Okay, they would let you. They had to be there, but they didn't...

Alan Kotok: Right, right, yeah.

Q: Okay, good. When you're-- did it initially-- how far-- how deep did you search, you know, go through in terms of trying to see <inaudible> possible moves.

Alan Kotok: Yeah, I think what we found was that going more than about four full moves was beyond the computing capacity.

Q: Yeah, because there's a geometric thing _____ on there, right.

Alan Kotok: And also, we would, as it says in the paper, we would-- what the basic scheme was that we would, at any given point, we would generate all the legal moves. So you'd look at each piece, and all the

places it could move and build up this big list of moves, and then we would-- we had a very, very quick evaluation function that would try to just order these moves in terms of plausibility. And then depending upon how deep we were, we would take fewer moves as plausible on each level. So we had this algorithm that tried to come up with plausible moves based on a simple evaluation. And then try that move and see where it would go. And again, later, apply this alpha-beta heuristic before going further. We talk about plies, which were half moves, so we would, you know, flip the algorithm over on each half to optimize for black or for white.

Q: Yes.

Alan Kotok: So obviously this has been 44 or 43/44 years since this was done. And so my firsthand knowledge of any of the details has, you know, faded into the fuzz, you know. <laughs>

Q: Those neurons have long since been co-opted for some other purpose.

Alan Kotok: There was once, as you mentioned, it's still an anecdote or something, but I recall one instance where-- so it's towards the end of the project when the program was working, you know, incredibly well, we would, you know, find a volunteer who wanted to play chess against the computer. And so there was one game where we were looking and the position of the computer was really dire. And we said, "Oh, we wonder what it's gonna do?" And I must say that none of the programmers were what you would call "chess experts." This chess group were computer hackers, not chess experts.

Q: You knew how to play.

Alan Kotok: We knew how to play, and we knew good positions from bad positions, and so forth, and we tried to hit the books-- and as it says there, books were not at all quantitative about evaluations. The advice as to how to play good chess was all very qualitative. But anyway, but we usually had some hangers-on who were pretty good at chess, and you know, so the question was, "What would the machine do in this case?" And it was pretty much trapped. And eventually it spews forth its answer, managing to get out of trouble by moving through another piece, some bug in the program, extracted its queen by going past the...

Q: Yes, do, and move right through.

Alan Kotok: Right, right through some other piece. <laughs> So we all got a big laugh out of that. That uh.. you know, it was some case which we obviously just hadn't considered that.

Q: Exactly. And it was a part of algorithm.

Alan Kotok: Right, right, yes. So.

Q: That's pretty funny.

Alan Kotok: Yeah, so we did actually, you know, try to play games, and there just wasn't enough computer time. You couldn't play by chess rules in terms of tournament rules of you have two hours to make 50 moves or whatever. If that's the right number.

Q: Yes, sure.

Alan Kotok: The computer was just much too slow. And so we would play for a while until it became evident what was going to happen, and try to reach some-- learn from the mistakes the computer made, and try to tune up the algorithms in the process. And so what I always characterized the ultimate state of that program was, "Beats rank amateurs." <laughs> And if you were at all a tournament chess player, you would have no problem with this computer.

Q: But just a total amateur, it might beat a total amateur.

Alan Kotok: It might beat a total amateur, right.

Q: Did you do anything like put in openings.

Alan Kotok: Book openings and things like that?

Q: Yeah, book openings or anything like that?

Alan Kotok: Not to my recollection.

Q: To your recollection it was all done with logic. Right, although I-- it seems unreasonable that we didn't have some starting sort of, you know, canned starting idea. So you know, "Pawn to king four, or something like that, and if you get this response then, you know, it's just...,"

Alan Kotok: so while I don't recall specifically that being in there, it seems somehow unlikely that we didn't have some kind of starting start-up.

Q: Yeah, at least in the first move or two.

Alan Kotok: Yeah.

Q: And I presume that you didn't have any of the backend analysis that Ken Thompson put in his thing.

Alan Kotok: No.

Q: Ten years later.

Alan Kotok: Right. Have you had any discussions with Ricky Greenblatt?

Q: No.

Alan Kotok: Okay, so I mean, he sort of picked up, not our program, but sort of the MIT chess effort in the mid-'60s.

Q: Now did he start working on it before you graduated?

Alan Kotok: No, no.

Q: He was <inaudible>...

Alan Kotok: Oh, I know him.

Q: Okay.

Alan Kotok: I mean, he was another Model Railroad Club person, and he-- I don't know exactly when he came to MIT, it was around the time I graduated, and he was interested in chess, and he built a-- maybe with some collaborators, a program known as, I think it was MacHack VI. It was done at Project Mac on the PDP-6. So he lives in town here, and I'm sure he'd be someone who you could talk to. By that time,

so our program basically reached whatever state it was by the Spring of '62, when I graduated. And it was sort of put to bed at that point in time.

Q: How many of you were there?

Alan Kotok: So there were a total of five people. There was the initial four were, besides me, Charles Niessen [ph?], Chuck Niessen, whose these days is some sort of director over at Lincoln Lab. And Mike Lieberman, who is on the faculty at Berkeley. And Elwyn Berlekamp [ph?], who is also Berkeley faculty, and fairly famous computer game theory person. Elwyn dropped out of this project at some point, and Bob Wagner, another-- so these were all sort of East Campus Model Railroad Club friends-- and Bob Wagner is at, I think, University of North Carolina-- what's in Raleigh-Durham?

Q: Raleigh-Durham? Duke and...

Alan Kotok: Duke, excuse me, Duke, yeah. He's Duke faculty. So and he's a mathematician. So we had this sort of mathematics, computer science. Of course, computer science was a word that wasn't invented prior to my graduation, I don't think. <laughs> So, the group shifted slightly, but I guess I haven't really had much contact with Mike Lieberman, but I keep in fairly close contact with Elwyn Berlekamp, and I see Chuck Niessen around once in a while. He's in the area. And Bob Wagner, I see once in a while. So.

Q: So you still-- they haven't all just disappeared.

Alan Kotok: No, no they haven't.

Q: Into some cloud.

Alan Kotok: No, no. And in this we've called ourselves "The Chess Group," okay? And I suppose in those days around the computer center people knew who The Chess Group was, because obviously we were always around there. <laughs>

Q: You know, there's some indication-- I'm not an expert yet, but one of our curators seems to think that this program may have been one of the earliest ones that actually played with all the...

Alan Kotok: Yes, I'm fairly sure it did.

Q: ...with all the board. And you know, allowed all the possible moves.

Alan Kotok: Right.

Q: Now did you do castling, and en passant? I mean, some of the more sophisticated moves of chess?

Alan Kotok: Yes, I believe-- again, I'm trying to refresh my memory from what I read.

Q: Yeah, it was 40 years ago.

Alan Kotok: But I think we didn't initially have those, but by the time we were done, we had added that capability. Yeah.

Q: Well, three years to do it. You can get a lot of work done.

Alan Kotok: That's right. Well, I mean, of course this was not the exclusive...

Q: You had other work to do.

Alan Kotok: Right, right. There was also this problem of getting an MIT degree, and doing physics, and all these other things. So. <laughs>

Q: Yes, doing the stuff to graduate.

Alan Kotok: Right, so not to mention spending a lot of time hacking at the Model Railroad Club on, you know, building the model railroad and control systems for that, and stuff like that. So.

Q: They must've continued to be relay-based. At least during your...

Alan Kotok: Yes. Yes, it was relay-based, and so after I graduated, I kind of hung around here for quite a number of years. Come in after work, and so there was a new version of the relay-control system that the original one had been designed by Jack Dennis back-- and friends-- back around 1950. And so by 1965 we had built-- designed and built a new one, which was noticeably more reliable and capable than the original one. Again, all out of telephone relays. <clears throat> And that ran in Building 20, which is where the club was, which is right on the land we're currently sitting in, until Building 20 was torn down

like five years ago or so. At which point riggers were brought in to pick this humongous relay thing up, take it over to the new location of the club and reinstall it! And it ran there until a couple of years ago as well.

Q: Oh my goodness.

Alan Kotok: So and now it is at the New England Museum of Telephony in Ellsworth, Maine. So. <laughs> So, this relay control system lives on.

Q: Lives on, your model railroad.

Alan Kotok: Right. I was also very much into telephone systems. And we designed and built a telephone system for the model railroad, called the RotoStrowBar Telephone System back in the early '60s as well. So.

Q: Oh my goodness.

Alan Kotok: So there-- I've often joked that if some alien force came and tried to shut down the country by sending waves that destroyed the operation of semi-conductors, that I was still prepared to design computers out of relays. <laughs>

Q: Okay, yes.

Alan Kotok: It might not work very fast, but <laughs>...

Q: But they'll work!

Alan Kotok: They'll work, right.

Q: That's good.

Alan Kotok: Right, yeah.

Q: Just a couple other questions rolling back to the chess. You told me the story about when it ran, and it made an illegal move to extract itself from dire straits (?) [ph?] . When you were writing the thing, what

were the most difficult things to do? Were there any challenges along the way as you guys slowly built this up?

Alan Kotok: Well, we had a lot of ideas that we tried to implement that are mentioned in the paper that just got to be too complex, and we had to back off. Various complex evaluation functions, and as I recall, you know, there were data structures were whatever FORTRAN tables would allow you. There were a lot of things that Lisp structures would've helped with, but we didn't have the underlying Lisp structures base. You know-- again, this is speculation-- but the Lisp system was, you know, being sort of co-developed back in the early '60s as well, and by the time we might have moved over to it, it was too late to do that. What else? Well, data structures. Oh, also recursion was not-- a chess program would really like to push stuff on a stack and go in, you know, sort of recurse [ph?(ok)] to do the next level. FORTRAN had no recursion capability. So we really had to manage all of this stuff in a sort of linear function, which was complex. There were a lot of tools that were sort of being developed outside our project, but sort of we couldn't keep switching tools underneath the program.

Q: Because you had so much code that you wanted to preserve.

Alan Kotok: Right.

Q: You remember how much code there was in the program when you finally sort of put it to bed? How big it was?

Alan Kotok: No, I don't. But that answer will become evident when I get the copy. Because the listings of the program were attached as an appendix to the thesis.

Q: Oh my goodness!

Alan Kotok: So it will all become evident soon! I expect.

Q: Okay! Oh, very good! Very good.

Alan Kotok: And if anyone has an old 7090 FORTRAN compiler, and the FAP [ph?(ok)] program, they might be able to run it. <laughs>

Q: Okay. That would be interesting.

Alan Kotok: Yeah.

Q: I think there are software emulators for most of those old machines. Though, I'm certainly not an expert on what's out there.

Alan Kotok: I was quite taken when I first discovered a Java applet that would emulate a PDP-1 and run Spacewar on a web browser. So. <laughs>

Q: Oh my goodness! <inaudible>

Alan Kotok: So that was...

Q: As an aside-- it probably doesn't belong in the oral history-- but the PDP-1 at the reconstruction at the museum, PDP-1 now works. And they're getting the-- they're working on getting the display. And so you will have Spacewar _____.

Alan Kotok: Right. I've been sort of following that a little bit at a distance. <clears throat>

Q: All right, so you're tapped into that.

Alan Kotok: And was planning to come out for whatever the opening celebration is.

Q: Yeah, okay. Oh, that would be great! Good! I think we should probably change the tape.

Q: Let's continue. Probably what we ought to do for the time being, let's leave the chess world. I guess I had one other question. Did you ever play any formal competitions with any other chess programs, or any other people?

Alan Kotok: No. No I believe that Greenblatt's program was the first program to actually compete in a computer chess tournament.

Q: Okay, yeah.

Alan Kotok: Obviously, there must've been another one if it competed. But. <laughs>

Q: Yes, exactly.

Alan Kotok: Right.

Q: There was a fellow by the name of Bernstein-- frankly, I can't remember his first name-- that wrote some sort of very simple chess playing program for the 704, and there's actually a video of him controlling it using the switches on the keyboard.

Alan Kotok: So let me see if there's-- I thought maybe there was an illusion. So there is an illusion to Bernstein's program here.

Q: Oh, okay!

Alan Kotok: It says, "Before commencing work on our test program, we studied a report by Newell, Shaw, and Simon, covering previous attempts, such as the Los Alamos program, and Bernstein's program at IBM." So maybe it was secondhand through the Newell paper.

Q: That you heard about.

Alan Kotok: Heard about Bernstein's program.

Q: Okay.

Alan Kotok: We actually-- the way we controlled the program during the games that we actually played was that because there was no typewriter, there was no online keyboard on a 709, the-- we would print out on this humongous line printer that they had, the list of all legal moves with their number, just numbered from one to "n", okay? And then you would key that number in binary in the console switches, and push the start button. So the computer would end its move by doing this printing, and just stopping, waiting to be restarted to read the number out of the console switches. So someone says, "Okay, so I'm going to reply to your move by, you know, knight to king bishop three," and we'd just go, "Knight to king bishop three, that's move 27. Now let's see, 27 in binary is...," you know, it's "hm, hm, hm," you know, we keyed it in and pushed the start button.

Q: Oh, that's fascinating. Okay. Well, that's not a bad-- hey, it works!

Alan Kotok: Works, right?

Q: Exactly. All right, cool. Very good. After you left, had graduated, what'd you do next in your career?

Alan Kotok: So I probably ought to go back a little bit. On top of all this chess program stuff, I was also working in the Research Laboratory for Electronics, which had acquired by donation, a PDP-1 from Digital Equipment. I guess it was about '61 or something like that, during the-- and Gordon Bell was at DEC at that time, and prior to that he had been a graduate student in the RLE at, you know, TX-0. And he had, I don't know, his master's thesis project or something was to design a magnetic tape controller to run one of the big IBM-style, you know, 75 IPS tape drives. And so he built that and connected it to the TX-0, and I knew him from those days, because we were TX-0 users at the time.

Q: Yes, exactly.

Alan Kotok: So I worked for the lab on top of doing this thesis, I was on the sort of...

Q: And doing your course work.

Alan Kotok: Doing my course work. And this was also the time that Spacewar was just being developed and I was involved with that as well.

Q: Oh!

Alan Kotok: You know, to keep the record clear, I didn't write any of the code but I was probably the first manager of the project in the sense of keeping Steve Russell going by providing him with motivation and some software bits that I had gotten out of various libraries and so forth. But anyway. So I was involved in system development for the PDP-1 here at MIT. And so Gordon wandered through one day, says, you know, "How you guys doing?" and all that. And so he said to me, "Well, what are you doing this summer?" And this was the summer-- because I had been admitted to the master's program here. And so this was the summer of '62, and the previous two summers I'd worked for Western Electric Engineering Research Lab in Princeton, New Jersey at the time.

Q: Oh!

Alan Kotok: And so I said, "Well, I don't know, I'll probably go back there, and you know, spend another summer." And he said, "Well, why don't you come out and work for us, for DEC?" And I said, "Well, that's cool!" You know, I probably didn't use that terminology since it hadn't come into the language yet. But anyway, so he said, "Well, good! You're hired! Call personnel and tell them I hired you." <laughs>

Q: That sounds like Gordon(?)_____.

Alan Kotok: Right, so I started at DEC in the Summer of '62 as a summer job. And when the fall came-- and so what I started to do was my first project was to write a FORTRAN compiler for the PDP-4.

Q: Okay.

Alan Kotok: And come the fall, I was, you know, pretty into that, and I don't remember how far that project was along. I mean, in those days, you know, "Oh, you wanna compiler? Well, that's three man-months," or something like that.

Q: Exactly.

Alan Kotok: We didn't know that you-- well, I mean, the machine was so small you couldn't write a lot of code, it wouldn't fit. So you had to do it in little code, okay?

Q: Yes.

Alan Kotok: But I said, "Well, you know, I can arrange my classes so that I go to school on Monday, Wednesday, and Friday, and I could come out and work for DEC on Tuesdays and Thursdays." So he said, "Okay". Well, you know, so I went sort of to part-time status, and did that. Come the spring term, I said, "Well, gee, I could probably get the classes I need on Tuesdays and Thursdays, and I could work Monday, Wednesday, and Friday." And he said, "Oh, why don't we just put you on the payroll full-time? So, and you can go to class." So we made that arrangement. And well, so I had completed the master's course requirements fairly rapidly, but then there was this master's thesis. And I had to come up with this project, since I had been working both in the PDP-1 lab, and with the computation center, I had, you know, connections both with-- my master's thesis project was under Corby [ph?], Fernando Corbató [ph?].

Q: Yeah.

Alan Kotok: And we were trying to make the PDP-1 be a remote display terminal for his time-sharing systems, CTSS, that was on the 7090 at the time. And so I had to design the whole interface controller that would speak some wire protocol to whatever the thing was. And well, I hate to say it, but this project sort of dragged on, because I got sucked into what seemed to be more of the real world at DEC. Although, I finally got it done under great pressure from Corby, Gordon Bell, my mother, you know, and

others. You know, the final thing my mother said, “Well, I’ve told everyone you’ve gotten your degree, so you damn well better get it!” You know? <laughs>

Q: I love it! Very good. That’s great.

Alan Kotok: Yeah, so I finally finished off that degree in ’65. But in the meantime, you know, sometime not very long after that summer, and the FORTRAN project, Gordon came running into my office-- and since you probably know Gordon, if Gordon does anything, he comes running in, okay?

Q: Yes, right.

Alan Kotok: And he says, “Oh, I’ve got this great idea. We need a yet smaller computer than this PDP-4, with an 18-bit machine.” He says, “I’ve got this idea for a 10-bit computer.” And this was not in the serial mode of the old IBM 1620 kind of thing. This was parallel scientific 10-bits. And I said, “Well, okay.” He says, “Let me tell you about it.” So he gave a rundown, and he said, you know, “Would you like to do this project?” And I said, “Oh, sure!” So I took that on, and soon expanded it from 10 to 12 bits. And basically came up with the instruction set for the PDP-5. And was just beginning to get into, you know, the detail design, when Gordon came running in again, and he said, “Oh, I got this idea! We’re gonna build a big time-sharing system!” And I said, “Oh, wonderful!” And he said, “Well, you got to get rid of this PDP-5 thing. Here’s Ed de Castro [ph?(ok)]. You know, he’s ready to take that on.” So I sort of turned the whole thing over to Ed. And this is now, must be some time in ’63, we started on the PDP-6. And Gordon was the chief designer, and I was sort of assistant logic designer for that. And Dick Best, I guess, and Russ Doane [ph?], I guess Russ Doane was the circuit engineer. And there- there’s a photo of this PDP-6 crew, which has recently appeared in-- there’s a book that’s just out of a photo history of Digital Equipment Corporation, and I supplied this photo to them.

Q: Oh!

Alan Kotok: But anyway, so here’s this photo. I think there’s also a negative or something that sort of shows up here and there.

Q: You know, I have actually seen this photo.

Alan Kotok: Right.

Q: It’s, yeah, it’s in a book on Digital.

Alan Kotok: Right, right.

Q: That I have.

Alan Kotok: Yeah.

Q: Okay.

Alan Kotok: And most people can still pick me out from this photo.

Q: Yeah. Okay!

Alan Kotok: So here was the crew. This was after the PDP-6 was completed, I guess, at the end of ’63 or so, beginning of ’64.

Q: Okay. Can we roll back a little bit?

Alan Kotok: Sure, go ahead.

Q: You know, you were saying you did the FORTRAN compiler for the PDP-4. Did you finish that?

Alan Kotok: Yeah, yeah. It was shipped and all.

Q: How long did it take-- when did you actually end up getting it done?

Alan Kotok: It must’ve been late in ’62, because I started in the Summer of ’62. So I graduated, my bachelor’s degree, June ’62. I started at Digital, and that was my first project.

Q: Right.

Alan Kotok: And I’m pretty sure it worked, and got shipped. But some six months. I really don’t know.

Q: Yeah, you don’t remember. But...

Alan Kotok: I don't remember.

Q: ...it didn't drag out for months and months.

Alan Kotok: No, it didn't drag out forever. And really, when I started there, I basically thought I was gonna do programming, okay? That was-- the thought of designing hardware wasn't really in my plan, but why not? You know?

Q: Yeah.

Alan Kotok: So opportunity came knocking.

Q: Yeah, because you've done this chess program.

Alan Kotok: Right. So I had been involved...

Q: Had you done other programming?

Alan Kotok: Yeah, yeah.

Q: Was it at MIT?

Alan Kotok: Well, basically, a lot of sort of system tools for the PDP-1 that we had in the lab, and the TX0 before that. And there was this amusing-- while we're telling anecdotes-- amusing PDP-1 story. So after we had-- soon, after we got the PDP-1, the software that came with it was extremely minimal, and included a very poor, at least by our standards, assembly program, a somewhat weird compiler that Ed Fredkin had written over at Bolt, Beranek, and Newman. He was there at the time. They had gotten an earlier PDP-1. We had been developing...

Q: What kind of compiler? What language?

Alan Kotok: It was a one-of-a-kind. It was-- it seemed to me it was a somewhat ALGOL-ish [ph?] kind of thing, if I were to call it...

Q: It was a peculiar one.

Alan Kotok: It was a peculiar...

Q: It wasn't a language maker?

Alan Kotok: Well, no, it wasn't one of the ones that you'd ever heard of before.

Q: Yeah, okay.

Alan Kotok: He called it DECAL, as I recall. DEC Algorithmic Language. So maybe that suggests it was in the ALGOL group of languages or something. I really don't remember much at all about it, except that a) it didn't catch on here at MIT; and the worst thing was that it didn't buffer its input off of the paper tape. And us elegant programmers were very offended by the fact that it sort of clattered the paper tape reader reading a few characters at a time as it would digest the input. And we thought that this was the nadir of programming elegance, okay? Independent of anything else.

Q: Okay.

Alan Kotok: So anyway, we had an assembler that had been developed for the TX0, by a number of our crew. I don't claim to be one of the authors of that original assembler. Dennis had put a lot of work into it, and I think Bob Saunders, and Bob Wagner maybe. Anyway, so we had this thing we called the Macro Assembler for the TX0. And macros were a very big thing in those days because the instruction set wasn't wildly rich, and so you'd like to have shortened forms of instructions to substitute arguments into the macro thing. And Jack Dennis, who ran the lab, kept saying, "Well, you know, we shouldn't really go off on our own and do this. We should really use the stuff that everyone's using." Well, we were more and more frustrated. So one day we hatched this plan. We went to Dennis and we said-- and we were the paid staff, the paid student staff-- now I specifically remember that in 1961 we got paid $1.70 an hour, which was the same price that you got-- that all under-graduate students got paid for like bussing tables at the cafeteria.

Q: Whatever it was.

Alan Kotok: Whatever it was. So we were the highly paid student programming staff, and we said, "If we start Friday after work, and you come in Monday morning, and we have macro running on the PDP-1, will you pay us for the weekend time?" And he said, "That seems fair." So...

Q: Oh my goodness!

Alan Kotok: So we-- what we-- the trick was that we-- I guess the word would be "transliterated," the TX0 version of the program. So we had four or five of us, and we sort of parceled out routines, and we said, you know, "The interface remains identical. Here's the sub-routine calling conventions. Same input, same output, recode," okay? And so-- and the test was that it had to assemble itself by Monday morning. And sure enough, it passed!

Q: Oh wow!

Alan Kotok: And we made our 15 bucks or whatever. <laughs> Whatever we got. So I think that was some kind-- at least at the time-- some kind of a record for writing a significant piece of software.

Q: That's pretty amazing. I think that'd be pretty amazing today.

Alan Kotok: So that was one of the more fun stories, so.

Q: Oh, that's good, great!

Alan Kotok: Oh! I have another chess story. Now that-- you were saying about chess stories.

Q: Oh, good!

Alan Kotok: So the fact that this chess program as being developed was pretty well known around the computer-- I would hate to call it computer science-- but the computer part of the department. And so one night we decided to set up a hack-- we were really into hacks-- that was reminiscent of these 18th Century computer machines. I mean, chess playing machines. The man in the box kind of machines.

Q: Oh yeah, sure!

Alan Kotok: Okay? But this was a high-tech version of it. And we had decided we were going to try to do inter-machine communication, and so we had set up a coaxial cable that ran back and forth from the PDP-1, which was in one room, with the TX0, which was in the adjacent room. And we said, "Well, now we've got this working, what're we gonna do with it?" And we said, "Ah! We're gonna have a chess program!" And so we put our chess experts in one room, and I-- and all this program did was you would type a line on the terminal and it would print it on the other terminal, and vice versa. And so I was the person who sort of watched the hallways until an appropriate sucker went by. And so I finally found someone appropriate, and dragged him in, and said, "You gotta see! We've got this new chess program on the PDP-1! You know? And you know, it's really good. You know, you wanna try playing it?" And he

said, “Oh, yeah!” So, he-- we had the chessboard set up, and so he would play, and I would type the moves in, and you know, the printer would in due time print the other, you know, the response. And we got quite a ways into this, and he was really impressed. Until at some point, the chessboard in the other room somehow got out of sync with the board that we had in our room. And so there was, you know, an illegal move, and you know, he said, “Illegal move.” You know, and then he started arguing with us. Okay? <laughs> And so he, “Hey, what’s going on here?” And I said, “Get out of there quick!” You know? He said, “What’s that wire?” You know? <laughs> And so he goes running into the other room as they go running out. So, but he had clearly been taken in. <laughs>

Q: Oh, very good. But he did figure it out.

Alan Kotok: He eventually figured it out. But only because we kind of...

Q: Fouled up a little bit.

Alan Kotok: ...fouled up a little bit. <laughs>

Q: Well, that’s a great story. Very good.

Alan Kotok: So anyway.

Q: That’s a good story!

Alan Kotok: Uhm hm.

Q: Okay. Let’s see...

Alan Kotok: We had digressed when you went back to finishing the PDP-4 FORTRAN compiler.

Q: Yeah, exactly. And then the next thing you did?

Alan Kotok: I’m pretty sure was guiding into the PDP-5 design.

Q: Yeah, okay.

Alan Kotok: So I think I really only did sort of one software project, at least in that round of my connection with software at DEC. And quickly got into the hardware side of the business.

Q: And worked on the instruction set.

Alan Kotok: I worked on the instruction set, PDP-5. So came up as I recall with the way how to handle the fact that the addresses would be, you know, too short for the various indirect structure and so forth to handle, handle longer addresses.

Q: <inaudible>

Alan Kotok: I don't think it was...

Q: It definitely was the classical problem with the short word like machine.

Alan Kotok: Yeah, so, which I suppose dogged us for the rest of our lives. But was short word-- the memories always got bigger faster, then the word lengths got bigger, so.

Q: Yes, but the same thing happened to actually E6. Started out really short word lengths.

Alan Kotok: Yeah.

Q: The kluge lived on.

Alan Kotok: So I went from that programming project to the PDP-5, and not many months later to the PDP-6. And I can't recall the exact date we started on each of those projects, but I recall it was-- so we would've been kind of into the Fall of '62, on the compiler, and then the end of '62, beginning of '63, on the PDP-5, and started on the PDP-6 fairly early in '63, I guess. Because we had it pretty much ready to go early in '64. I remember in the beginning of '64, we made a sales trip to Australia to sell the PDP-6. We hadn't sold any in the United States, and the machine was just barely running at this point in time. But the Australian government had put out a request for bids, or request for tender-- is what I think they called it-- for the five major universities to upgrade their computing infrastructure. And there was a guy who Gordon knew, Ron Smart, who was, I guess, on the faculty at the University of New South Wales, or something like that. And who was interested in this. And I guess in the process came on to the DEC staff, and ran the Australian office for a while. But anyway, so he clued us into this. And I remember Gordon and I, and Harlan [ph?(ok)] Anderson, who was the vice-president of DEC in those days, put together a big boilerplate kit of stuff to build proposals from, and went to Australia. And we traveled

around and visited these various universities, and then holed up in our hotel room, with typewriters produced proposal kits, which we duly submitted. <laughs>

Q: After you talked to them.

Alan Kotok: And we won one of the orders in Perth at University of Western Australia. So the big joke was that the very first PDP-6 sold was to the city at the furthest point in the earth from Maynard, Massachusetts that was populated, you know? <laughs>

Q: That's pretty good, very good.

Alan Kotok: So I guess we kind of finished getting the PDP-6 up and running in the beginning part of '64.

Q: What piece-- or how did you fit into?

Alan Kotok: Okay.

Q: I know Gordon sort of was the principal designer of it.

Alan Kotok: Right.

Q: And I know I've seen his name on the Logic diagrams.

Alan Kotok: That's right. Well, so I did Logic design. I mean, you know, Gordon and I would sort of hash things out together. But we'd say, "Okay, so the accumulator is gonna have these inputs and these outputs, and do these functions." And so...

Q: "And this is how we're going to _____. It will go this fast, you know, but you can't afford to put anymore stuff in there," or whatever.

Alan Kotok: Right. So you know, I was the, you know, Gordon was the primary designer, I was the assistant Logic designer for that machine.

Q: So then you...

Alan Kotok: Right, and we would go back to-- so this guy here, Russ Doane, he was the circuit designer.

Q: Yeah, okay.

Alan Kotok: And so in those days, you know, we had certain logic cards that were standard DEC issue cards, but we decided that we needed to build special boards for like the main registers of the machine in order to get enough interconnection, because the connectors were not very dense in those days. And so we came up with this bit-slice board idea. And so, and we had these ten-and-a-half inch boards which the normal, in those days we, five-and-a-quarter.

Q: So it was a double-height board.

Alan Kotok: A double-height board, and it was double-depth as well.

Q: Ah!

Alan Kotok: And we came up with the idea that we would distribute the timing pulses on the backside. So there were nine and nine side-by-side. There were two groups of nine and nine, so it was a 36-bit machine. And in the middle there were the clocking system boards that fed the signals out the back, and then they were bussed across the backs of these other boards. Well, this made a very difficult situation if you need to remove a board, because you had to pull all of these connectors off that had, I guess, the initial design was they were...

Q: So these you're talking about were on the handle side.

Alan Kotok: The handle side. Right. There were no handles, there were connectors instead. <laughs>

Q: Yeah, there were connectors, but on that side.

Alan Kotok: That side, right. And every time you would go to remove a board, because of a fault, you would run the risk of, you know, breaking some of this weird bussing structure that was sort of free-hanging connectors. Eventually, I guess we went to flexible printed circuit technology.

Q: Flex cables.

Alan Kotok: Flex cables and so forth, which I think were just coming in at that time. And the design style of that machine was sort of not to be seen again, maybe for good reason. It was not the traditional synchronous clocking design. It was, what we call today, asynchronous. It wasn't classic asynchronous in terms of the completion of a function was determined by the logic, which then propagated something in most cases. It was that functions were triggered by pulses that would go through a delay line that would be set to cover the logic delays that were necessary.

Q: Of the logic it needed to set up.

Alan Kotok: Right, right. Yes.

Q: The transfer, from _____ register, _____, and stuff like that.

Alan Kotok: Exactly, exactly. And so depending upon what you were doing, more or less time would be allowed. And so this delay line pulse generator thing was sort of the style of that design.

Q: Okay.

Alan Kotok: And the one thing that really was totally asynchronous was the carry propagation logic. In that machine we didn't have carry look-ahead. But what we had was the carry signals were literally pulsed from one bit to the next. So if there were a carry generated within one bit, a pulse would go to the next bit. And the pulse was controlled with pulse.?? And there was this giant OR gate that looked at all 36 carry pulses, and of course, you could've done an add in which there were no carries.

Q: Right.

Alan Kotok: But if there were any, it would look for the trailing transition of the OR of all these pulses.

Q: So it would look for the last trailing edge.

Alan Kotok: Right, the last trailing edge, and then it would move on to the next step. <laughs>

Q: Oh, my goodness, okay.

Alan Kotok: Yeah, so there were a number of interesting quirks in that design.

Q: Exactly, that's interesting.

Alan Kotok: So.

Q: I know Seymour Cray [ph?] had a somewhat similar, he would just add inverters in the line-- in the signal lines to buffer everything out, so it would come out at the right thing. I mean, he had a clocked system, but that's how he would, you know, sort of manipulate it. Sort of reminds me a little bit of the delay lines, the block pulses.

Alan Kotok: Yeah, right. So I think this delay line design approach went back to the TX0, which I guess Dick Best, who was with Ken Olsen [ph?(ok)] over at Lincoln Lab, was one of their early starters of Digital, he was sort of the chief circuit designer for Digital back in the early days. And this kind of pulse circuitry was one of his things. Ah, I think by the time we got to the PDP-10, we went to more standard clocking-type, synchronous clocking circuitry. Although, just where we made that transition is a little fuzzy in my mind.

Q: Okay. So, you know, I know there were some problems, you know, some reliability problems in the 6.

Alan Kotok: Absolutely, yes.

Q: Talk to me about some of those experiences when you first tried to get machines up and running.

Alan Kotok: Yeah, the 6 was not a winner in that regard. And oh, I remember there was a-- one of the machines was sold to Charlie Adams, Adams Associates. It was right across the street here over in 575 Tech Square-- or 585-- whatever that company-- wherever they were. <clears throat> And we were down there all the time trying to solve flaky problems with it. And it turned out that there was a lot of electrical noise sneaking into these systems from things like printers, which are excellent-- the old mechanical printers, which basically scraped along the paper with a drum, were Van de Graaff [ph?] machines in disguise. Even some of the mag-tape machines were essentially Van de Graaff machines in disguise.

Q: Uhm hm.

Alan Kotok: And the circuitry was just too noise prone. So you know, uptime, continuous uptime was not at all good. And I guess sort of the dénouement of all that was when Charlie said, "I'm throwing this machine out! Take it back!"

Q: Oh-oh.

Alan Kotok: And so there was this crisis at DEC where Ken Olsen in essence said, "You know, I've been skeptical of these big computers all along. And <clears throat>...,"

Q: Of course, this is with all the factors. But there are some side issues here.

Alan Kotok: And sort of, you know, "I don't want anymore big computers!" And I remember, well, whether this is faulty memory or not, I don't know, but it sticks in my mind, him saying to the large computer engineering group assembled, you know, "At DEC everybody gets to make a mistake. You've made yours!" <laughs>

Q: Ooh!

Alan Kotok: So in the meantime, we were well into the design of the PDP-10. So the 7 and the 8 and the 9 had sort of-- other groups were working on-- and the 6 group moved on to what was going to be the 10. And the, you know, our approach was, "Well, okay, so we think we understand the circuit problem, so you know, onward to bigger and better computers!" You know? <laughs>

Q: Yes.

Alan Kotok: With, you know, we liked the instruction set. It had gotten a lot of rave reviews. And so we were just gonna do it right this time. <coughs> And so Ken said, you know-- so the PDP-6 is sold for like $300,000 or something in that general price range-- and he said, "$100,000, that's the computers that we want here." So there was a great regrouping, and somewhere in this process, Gordon decided he'd had enough, and headed off to Carnegie-Mellon. And so...

Q: He left for Carnegie-Mellon.

Alan Kotok: He left for Carnegie-Mellon.

Q: PDP-6.

Alan Kotok: After the PDP-6. But some time into this-- I think he was around for this crisis, okay?

Q: Yep.

Alan Kotok: My chronology is not 100 percent here. But we said, "Well, you know, well, gee, we really like this design, how can we somehow meet Ken's objectives and still continue with this basic approach?" So I guess somehow we managed to talk Ken up to $150,000. And we sort of looked at this machine and said, "What are all the things we could conceivably make optional?" You know? And so we pulled out the floating point. That became an option. And we pulled out the byte string instructions-- we called character instruction at that time. We made the-- well, I guess in the PDP-6, we had the first 16 addressable registers were the-- served multiple purposes, they were accumulated. They were addressable in multiple fashions. And they were actually implemented on the memory BUSS of the machine in their own cabinet.

Q: Oh my goodness.

Alan Kotok: And well...

Q: They were hardware registers?

Alan Kotok: They were hardware registers. They were hardware registers. In those days, you know, the cycle time of core memory was the dominant thing, not the wire-time to get over to these registers in another box. So that didn't seem like a bad idea that the time. But we kept this-- so there was some PDP-6 option where you could run out of main memory. I don't know that we ever sold a machine that didn't have these registers, but you know, you could change one logic signal and the main memory would pick up those register functions.

Q: Right, as opposed to...

Alan Kotok: As opposed to the, yeah.

Q: Addresses that went to those first 16 point. Yeah, it diverted to off to the-- all right!

Alan Kotok: So that was maintained as an option. And then we had the 8,000 word, 8192 word main memory option as well. And we knew damn well that the operating, even the most rudimentary operating, system that we had developed would not fit. [I see your blinking again.] (refers to camera)

Q: Yeah, that's okay.

Alan Kotok: And so we basically crossed our fingers, and said, "You know, if anyone actually buys the 8,000 word machine," to the salesman, "Talk them out of it quick, okay?" <laughs>

Q: <inaudible> ...the computer, okay.

Alan Kotok: And eventually, Oxford University said, “No, that’s the one we want!” And we said, “Oh, I guess we’re gonna have to make a donation of the extra 8,000 words.”

Q: And that’s what you did.

Alan Kotok: That’s what we did, yes.

Q: But nobody else realized that you might do that.

Alan Kotok: Right.

Q: Okay, let’s stop.

Q: So that’s how you?

Alan Kotok: So we managed to get this project approved and crossed our fingers that we weren't going to have to support too many poorly configured machines, most of the customers figured this out and bought the options. Since we had an instruction trapping feature so if a program used one of the instructions that wasn’t present in the machine it would trap it and we could emulate the instruction, so you didn't have to have special software that avoided the missing instructions. So that--

Q: So that made the software pass manageable.

Alan Kotok: Yes.

Q: As opposed to all these different versions.

Alan Kotok: Right.

Q: Tell me about the development of the machine, what role did you play in this new machine?

Alan Kotok: So the first PDP10, which was known at DEC as KA10, I guess when Gordon left I became sort of the chief architect honcho, and there was a fellow named Bob Clements who worked for me, he sort of took the role I had in the PDP6 realm. And I'm not sure exactly when these people showed up, but Dave Gross and Alan Kent were also involved in the design of if not that machine maybe the next one. But anyway, so anyway the first PDP10 came out in '67 I believe, so we learned how to take longer to design computers than we used to <laughs>, there may have been some false starts in that process as was noted. And that machine was technically a pretty good success, that's really the machine that made the reputation for DEC in the, whatever reputation it had in the high end, a lot of computer science departments had gotten these machines and I still run into people who remember programming on the KA10, and say "Are you the Alan Kotok who designed that machine?" Yeah, so--

Q: Tell me a little bit more about what you did differently to make it a more reliable machine, what did you fix?

Alan Kotok: I think there were-- we went to a much more robust mechanical arrangement that no longer had these flex circuit connectors on the back of boards that required, I mean we soon concluded that that was a major reliability issue. And we also moved to what at the time DEC was promoting as the Flip-Chip modules, which were these half, instead of five and a quarter, two and seven-eighths or whatever size, boards, and went to a much smaller card format, although we used the five and a-- you could still make five and a quarter inch, it was five and a half and two and three-quarters, whatever those numbers are. Yeah, but we had some larger boards but they didn't have rear connectors on them anymore. And--

Q: This must've been a new generation of circuits.

Alan Kotok: So this was a new generation of circuits, this was not integrated circuits, this was still discrete transistor logic, but I'm pretty sure we went, and I'm going to sound pretty stupid if I've got this wrong, but I'm pretty sure we went to a more traditional clocked arrangement in this machine, rather than-- we didn't have a lot of this very-- we didn't have all these delay tuning things.

Q: Got rid of the delay lines.

Alan Kotok: Yeah, if I remember correctly. So for whatever reason we managed to pretty much solve those reliability problems and the speed got noticeably higher over the time as well. It was still core memories in those days.

Q: But core memories kept getting faster.

Alan Kotok: So as I recall like the original PDP6 memories were like five microsecond cycle times and we were down into like two microseconds by the time we went to the 10. Yeah, and the 16 high speed registers were integrated much more closely into the basic machine. Of course we were also moving on in operating systems as well, we'd improved-- so we had this timesharing system that we had built and it got more robust. Yeah, so.

Q: So it was similar fundamentally.

Alan Kotok: Fundamentally it was the same instruction set.

Q: Same instruction set.

Alan Kotok: Same instruction set and we carried that several more generations. We then-- that machine didn't cause Ken any embarrassment, we sold quite a number of them and so we then moved on to a first generation integrated circuit version of the machine, it was called the KI10, and it used small scale integration circuits primarily. The physical packaging was very similar, it was these small boards, small board packaging. It's going to be hard to get these facts straight but I remember that the first prototype of that machine when we finally powered it up amazingly enough sort of ran within the day or something like that, I mean we had improved our CAD systems sufficiently to wring out a lot of errors before we built it this time. And yeah we'd also moved, the PD6 was built with back panels that had soldered wires so the connectors were just little solder lugs and people physically laid the wires in and soldered them all.

Q: And soldered them in, okay.

Alan Kotok: I guess, I think yeah, I'm pretty sure, the first PDP10 used a wire wrap back plane, and I don't remember whether we had automatic wire wrap at that point or whether it was hand wire wrap but by the next generation at least it was all programming control wire wrap, so we were able to generate all the wire wrap tooling control stuff straight out of the logic diagrams.

Q: No transcription errors and no truth errors.

Alan Kotok: Right. I remember it took quite a bit of time to do all the computation to, we called it "run the wire list," and Dave Gross and I and Alan Kent, who were the main crew on this KI10 machine, had this cart, a little wheeled cart and on it we had big stacks of magnetic tapes and those big sort of cake cover type disc packs, the big--

Q: Sure, I remember those.

Alan Kotok: And at night we would rumble around the plant looking for a computer that wasn't in use and commandeer it and stick all of our tapes and discs on and run the wire list.

Q: This is around the manufacturing plant?

Alan Kotok: Around the manufacturing site, yes, that was sort of the best way to get computation resources, yes. Part of this process I recall was punching cards to run the wire wrap machine or something like that, and the only reliable card punch was on this Burroughs computer that accounting had, and so we had to write tapes that would be- we could take over to the Burroughs machine and run the punch card operation to drive the wire wrap machine <laughs>. So a lot of this was ad-hoc, night-time project computing. We still in those days drew the logic diagrams by hand, we had graphs for them still and then we would key in the connections into-- so we had this thing where we could say this signal connects to this, this and this, and that would generate the wire list, but there was still no hard connection between what was on the paper and what was in the machine. So we would-- remember the listings in those days were fanfold and wide paper and we spent a lot of time <laughs> leafing through listings to make sure that the design was as we intended.

Q: So it wasn't totally manual transcription.

Alan Kotok: Right. Yeah, so that was-- we did that, that was still in the mill, so we were in the Maynard Mill from the time I joined through the KI10 and so I guess that machine came out in like, '70, '71 or so, so three and a half more years or so had gone by. And then we said well we're going to do the next machine you see, and the next one we called KL10, I don't know why it was called KL, over the years people have tried to back form reasons for these acronyms, and as far as I can tell well, at least between most, the early time, Dick Best's office was in charge of giving out product numbers, and so he had this registry and he would sort of give you the next thing on his registry, and K were processors or something like that, and the first would be A and the next would be B and so forth, but then the marketeers sort of got involved and they wanted specific ones and blah-blah-blah, so.

Q: And all the machines were in order.

Alan Kotok: Right. So those were the simple days. Yeah, so.

Q: So the same sort of key crew did the first two 10s.

Alan Kotok: Right, and yeah there were always people coming and going and I guess I was the primary continuity through that whole line of machines. So after '71, we started on this KL10, and by this time we decided it was time to do micro-program control, rather than hardware logic, and so we set off in that

direction, I guess my main compatriot there was Paul Guglielmi [ph?], and we also took another step forward in the CAD world. I'd always kept a pretty close connection with the group at the Stanford Artificial Intelligence Lab, McCarthy's Lab, and they'd been dabbling in designing their own PDP10 because, well I think really, because they wanted to design a computer, but sort of the justification was "Oh you guys are too slow, and we're going to build this machine, it's going to run circles around your machine," and so forth and so on. But like a lot of academic projects it never quite came to fruition but they did develop some really cool CAD software. And this project that was the computer development project at Stanford was sort of the chief designer was a guy named Dave Poole who met an untimely end some time ago, I don't know, an accident or something like that, I don't recall precisely what happened to him. But he called this computer Foonly [ph?(ok)], I don't know where he got that, a whimsical name, and the CAD system that came along with that, gees, I don't remember, it might've had a better name than the Foonly CAD system, but anyway, we hired the guy who was the primary programmer for that, and so he came over to Maynard, it was Dick Helliwell [ph?] is his name, and continued to develop that for us. So now we were into a graphic interface CAD system where we actually drew the circuits and the logic elements, the logic elements on a CRT and when you drew a line that connected two terminal points from one gate to another that meant it was logically connected. And so all the wiring--

Q: All the wiring information now comes automatically.

Alan Kotok: Comes automatically, yes.

Q: All you have to do is get the logic correct.

Alan Kotok: Right. But of course the number of gates in the machine had gone up with every generation to--

Q: Because you added functions.

Alan Kotok: Right. Well not so much because we added functions but to make things go faster the usual speed of light didn't get any better and the way to make things faster was to have things work in parallel processing multiple stages, little pipelining kind of things, this kind of stuff, and we had added much more sophisticated memory management as well and the PDP6 just had a base register and an offset that you could apply to memories so the timesharing system could-- it was a base and a limit register. And then we went to various sophistication of paging systems and so forth. So the machine had become more complex although the instruction set-- sort of each time I guess we succumbed to adding a few instructions but mostly it was the same 36 bit machine. We did, by this point, have to add some sort of extended memory addressing hack to it, the original PDP6 had 18 bit word addresses which in that day we said well, gees, how much memory--

Q: How much could the customer afford?

Alan Kotok: How much could the customer conceivably afford. The Artificial Intelligence Lab here bought one of those early PDP6's and immediately went out to somebody and bought a quarter megaword memory which was the "full house" memory. So it didn't last anywhere near as long as we had thought it would <laughs>. So there was an extension to 23 bit addressing which allowed you to use up the index and indirect fields somehow, or maybe just the index field and I don't remember, but anyway, there were various things to extend the virtual addressing, the physical address mapping stuff had less limits so that whether there might've been still limits on individual programs the total system memory could grow beyond this. And I guess somewhere along the line in that project, and maybe in '75, we moved out of the mill over to Marlboro to the old RCA buildings over there where were--

Q: I remember when you finally bought the RCA building.

Alan Kotok: Right, I guess Ken waited until he could get them for a song or something, or the city of Marlboro gave him free taxes for the next 10 years, I don't know. But so we moved over to Marlboro and finished the design, I remember doing the debugging shifts over in Marlboro.

Q: The KL10.

Alan Kotok: The KL10. And of course so the micro-code development was a project unto itself.

Q: That was micro-code done by the logic design crew, or a separate crew?

Alan Kotok: There was I believe a separate group who worked closely with us obviously, but I don't recall personally being involved in that--

Q: In the micro-code.

Alan Kotok: Micro-code.

Q: Then there's suddenly a terrible temptation to expand the number of instructions.

Alan Kotok: Right, well we had almost totally populated the instruction field from day one, because I remember the PDP6, the claim to fame, one of its claims to fame was it had 360 instructions or something like that, and we got that by sort of decomposing the instruction field and so there was a lot of regularity

so in whole large classes of instructions there was a two bit field that said where the destination went, did the destination go into memory, did it go into a register, I forget the options, and there were source options, so you could have immediate addressing, direct addressing, indirect addressing and so forth. So there were-- we got a large number of-- and there were also half-word options, so you could do left-half, right-half, both halves, and so forth. So there was a lot of multiplication that went into making--

Q: Went into getting this huge number.

Alan Kotok: Right. But in the end it effectively filled up the instruction field and so there wasn't a lot of temptation to sort of, there were little holes here and there you could fill in but--

Q: There wasn't a lot of room.

Alan Kotok: Right, there wasn't a lot of room, so. And there have been a lot of software that people wanted to move up to the next machine, so we didn't want to take anything out and prevent those.

Q: It would always be backwards.

Alan Kotok: It was always backwards compatible.

Q: And each machine sold more--

Alan Kotok: Yeah.

Q: And was more successful as DEC grew.

Alan Kotok: Seemed to be. I don't know that Ken ever became really comfortable with the high end of the business, he really-- well, he may have gotten suckered in years later by Bob Glorioso [ph?(ok)] and the high end VAX business, this was the "DEC taking on IBM" phase of the world, which was probably an unfortunate beginning of the end, but. So this machine, the KL got delivered in '75 and we were doing some preliminary studies for future machines, and a friend of mine, well the aforementioned, Elwyn Berlekamp [ph?], called me up and said "Would you like to come up to Berkeley and teach as a visiting faculty for a year?" and this seemed to be a really good time to do this, since I was just, you know, I'd just completed a project, I wasn't too far into the next project. And so, well there was a little hassle that I almost didn't respond to him because I wound up with a kidney stone and I was in the hospital and blah-blah-blah and he almost gave away the position to someone else before I <laughs> before I got a chance to say yes, but it all worked out. And so I took a leave of absence and went out to Berkeley and taught

computer science for a year, and it was a reverse sabbatical kind of thing and it really worked very nicely to change, get into a different environment for a year and--

Q: Rather than just cranking out the next PDP.

Alan Kotok: Right, right. So eventually I came back and--

Q: You just did that for how long?

Alan Kotok: It was nine months, so the academic year, '75-'76.

Q: And what did you teach?

Alan Kotok: Well I taught logic design, I taught some courses on telephone switching, telephone system principles, I always had this side interest in telephone systems, and I actually guest lectured courses here back at MIT in those days too on telephone systems. Yeah, anyway, and some of the basic computer science 101 kind of courses that everyone has to teach and a couple of graduate courses and what have you, so it was-- there weren't any texts for anything but the most basic stuff, and so I remember having to-- there would be the is sort of panic, I would have to write my course notes and I'd do this on some text editor and print them out on ditto masters and run them through <laughs> and come running into class with "Okay, here's the notes for today's lecture," sort of just in time, a just in time production of subject matter <laughs>.

Q: That tends to happen when you're teaching a course for the first time.

Alan Kotok: Right, and everything I taught was definitely for the first time <laughs>. So also I continued to have a lot of connection with the friends at Stanford, which is just down the road, and some other friends who had a little company that was building computers as well, so.

Q: What was that?

Alan Kotok: This was Stuart Nelson and Mike Levitt [ph?(ok)] and it was called Systems Concepts, and they decided to go into the PDP10 clone business or something on their own. No-one has ever made a dime doing that, but there was always a temptation to do it, especially when soon thereafter Gordon decided that we no longer needed the PDP6 and we no longer needed 36 bit computers and 32 bit was the way of the future, and the VAX was the answer to this. I guess there may have been some--

Q: Gordon did come back by now?

Alan Kotok: Yeah, Gordon had come back and I guess the VAX 780 had come out somewhere around the timeframe as the PDP10-- the KL10.

Q: The KL10.

Alan Kotok: Yes, it was in the ’75.

Q: Yeah, so you had two.

Alan Kotok: So we now had two architectures that were reasonably similar in capability if not in programming style and so when I came back in ’76 we had again gone back to design a successor PDP10 and maybe it was six, nine months to a year, I don’t remember how long into that Gordon said “Stop, we're not going to build any more PDP10’s.” And so at that point I said, “Gees, I've been in this computer design business too long, I'm going to go do something else.” So I went off into the networking group for a while and also there was a technology office that the head of research, it was the head of research at one point, he lived down the street from me in Harvard, oh it’ll come to me, [Sam Fuller] ran and we did a lot of stuff like specifying some of the new BUS designs. And I got involved with bringing the Ethernet into DEC from Xerox, there was this Xerox-DEC partner-- and Intel partnership, and so I was involved in that and some of the system planning for Ethernet. At the time we were into calling-- we had various interconnects, that was the term, and we had the CI was the computer interconnect and so this was a high performance storage access, storage processor connection bus. We were moving more from parallel to serial in those days and the Ethernet we called the NI, the network interconnect. So I worked with the network group and so forth for a year or two.

Q: So this is after you came back.

Alan Kotok: This is after I came back. So I came back in ’76 and we spent a-- started another 10, abandoned it when we changed strategy, and then I went into this interconnect networking area for a while.

Q: Who was running that at the time?

Alan Kotok: Which?

Q: The networking.

Alan Kotok: Well the networking group was run I guess by Bernie LaCroute [ph?], and this interconnect technology group, the guy whose name just won't pop to the front of my mouth. [Sam Fuller]

Q: This isn't Strecker [ph?]?

Alan Kotok: No, no, no this was not Strecker, this sort of soft-spoken guy, another ex-CMU guy, he went to-- eventually went to Analog Devices I think. Sam Fuller. (at long last)

Q: Oh yes, of course.

Alan Kotok: Yeah Sam, I mean Sam, he lived down the street from me, and I've known Sam for so long it just sometimes--

Q: That's called anomie [ph?].

Alan Kotok: Yes.

Q: There's actually a word for that.

Alan Kotok: Oh well that's good.

Q: Anomie is the inability to remember nouns.

Alan Kotok: Uh-huh, yes.

Q: And it's a symptom.

Alan Kotok: Well what I know is that--

Q: Like schizophrenia.

Alan Kotok: Well, having thought about this, and I concluded that at least my brain's associative memory knows things that it will never come up with as distinguished from there's this intermediate category which is I know this information and I know in time it will come forward, and then there's the stuff that just pops

out. So I know I knew this guy's name and some background process runs and eventually it pops out. So, interesting brain, we can have Brain Science right across the street, maybe they'll figure this out. So I worked for Sam Fuller during that interval. So two things happened in the meantime in the processor business, one is the Vax 8600 project was started, which was the new high end Vax, there had been a 750 project that was--

Q: I remember it went 780.

Alan Kotok: Yeah.

Q: And there was a 750 which was sort of a mini Vax.

Alan Kotok: Yeah right.

Q: Poor man's Vax.

Alan Kotok: Poor man's Vax, and there was even a lower one than that.

Q: It was Micro Vax.

Alan Kotok: It was Micro Vax was in that timeframe too, so there were a number of Vax things and one was this high end 8600 thing. But the customers basically for the 36 bit line beat up Ken Olsen and said "You can't do that to us, we'll never buy another computer from you if you stop developing the 10 line."

Q: And of course if you look at the names of these customers it isn't something you necessarily want to have happen.

Alan Kotok: So a new 10 project got started up, I think it was called Jupiter, it was known as Jupiter, and they hired a guy who had been a architect in Gene Amdahl's [ph?] company, remember there was time when Gene Amdahl had started this company and they were going to build some sort of VLSI based thing, I've forgotten.

Q: Not Trilogy.

Alan Kotok: Trilogy, yeah. Yeah, so they hired this guy from Trilogy and he was going to build a super pipelined, I don't know, 15 stage pipeline PDP10.

Q: With incredibly sexy integrated circuits.

Alan Kotok: Yes, right. Now I don't know whether the DEC version, the DEC version I think was just going to be ECL custom, what do you call it, Gate Array, ECL Gate Array, the technology. And that was the same technology that was being used by the 8600, Vax 8600 people, and they were side by side in Marlboro. And sort of my view of this was I wish you luck, I wish you all luck. And some time, I don't remember when it was, so let's see, it was '77, '79 timeframe or so, Gordon called me and he said "We're going to have a design review of the 8600," which was known as--

Q: Venus wasn't it?

Alan Kotok: Venus, Venus right.

Q: They were planetary things.

Alan Kotok: Right. And would I be willing to take a week or two or whatever, and go to this design meeting, and I said sure. So and this project--

Q: Who were the people running this project, or doing the design?

Alan Kotok: Well the guy who was running it I think was Ulf Fagerquist [ph?] if I'm not mistaken, or maybe he was on the Jupiter side. I forget, gees, that's a name I'm not necessarily going to come up with. But there was this group down there and we had this design review, the reason Gordon called it design review is this probably didn't seem to be converging, it was the deadlines were slipping and slipping so--

Q: It was missing the schedule dates.

Alan Kotok: And missing the schedule dates.

Q: And there wasn't anything running.

Alan Kotok: No, there was nothing running, it was- there wasn't even a chip sent for fabrication, these ECL Gate Array chips. So the design review, as I recall, resulted in the conclusion that this project is not going to succeed, and so Gordon came running in and said "Alan, will you take over this project?" and--

Q: This was a lightly larger commitment.

Alan Kotok: Right. So I said “Well, I'll make you a deal, I'll go over to Marlboro and I'm going to take a month to assess the situation, talk to all the engineers and get a much--

Q: Get dirty.

Alan Kotok: Get dirty.

Q: Find out what's really going on.

Alan Kotok: What's really going on here.

Q: Not what they're presenting.

Alan Kotok: Right, and at the end of that time, whatever this period was, if it was a month or whatever it was, I said “I will give you a report, either I will conclude that this is hopeless and I will have nothing further to do with it, you can do with it what you want, or I will take on,” become the chief architect, we called that.

Q: Technical project.

Alan Kotok: Technical project lead, right. Well, I don't know, it could've been that I suffered from a burst of ego and “Gee, I can fix this,” or something--

<Crew Talk>

Alan Kotok: The 8600. So my feeling at the time was that the engineers were basically competent but that the machine had been divvied up into several functional units and no-one in fact was ensuring that the functional units could talk to one another, because the design styles, the whole clocking system, there were just major disconnects in the style of the design. So the timing would just sort of never work, so I figured if I could merely just get these people to talk to one another because they really had sort of developed into a bunch of fiefdoms and sort of get together in a written down, in a common approach to many of the important features of the machine that we could actually make it work. And since I was well known as not being a VAX person, here I am in the middle of this giant VAX project, I would periodically

threaten to say “Well, if you guys don’t come in with the answers to how this is going to work I'm going to tell you.” And so that was a big enough scare for them <laughs> to actually go and--

Q: Work hard to figure it out.

Alan Kotok: To figure it out. So there was a fairly significant reset back to the drawing boards kind of thing, again with a much better integrated set of design styles and parameters and what have you and the clocks would actually synchronize across the machine and all this sort of stuff. I don’t remember the details but I do remember clocking was kind of the major issue.

Q: The most obvious thing that would never work.

Alan Kotok: Right. Yes, so the algorithms within each unit were reasonably well understood by the engineers, I did persuade a few of my old engineering buddies to come into the project, people I trusted, Tom Eggers [ph?(ok)] and a few other people to help with this, and--

Q: Who'd been in the old--

Alan Kotok: 10 groups yeah, and again it didn't move as fast as we wanted, but it did succeed in the end, so we did succeed in building the machine and shipped and--

Q: It did get reset to a plausible place where you then went forward in a reasonable fashion you'd have something that worked.

Alan Kotok: Right. So.

Q: So how long did you stay with that project?

Alan Kotok: Oh it must’ve close to three years I would guess, yeah.

Q: So you said this is what needs to be done and you'd help see that it gets done.

Alan Kotok: Yeah. No I stayed with it until it shipped, or until well yeah, manufacturing was well underway and all the big bugs were out of it. Well so by that time, and I don’t even remember where we've gotten, so we must be some time in the early 80s, RISC architecture was being touted, and there was a group out at the DEC western-- well DEC had these warring research labs in and around Palo Alto.

Q: Can we back up, can you finish what happened to Jupiter.

Alan Kotok: Oh Jupiter. Oh well Jupiter went through a number of design reviews that I sat in and it had a complexity explosion that the conclusion was it would never succeed and the project was cancelled. So that poor guy left in disgrace I guess. I don't recall enough to know sort of kind of where it went wrong, possibly the chief designer just was in over his head in complexity. I mean pipeline machines are just unbelievably complex things, and so we basically went back to the customers and said "Sorry, we're not just going to do this," and we gave it the old college try but it didn't work. But I wasn't, other than some design reviews I wasn't involved. Yeah.

Q: Were you involved in any of the 20 work?

Alan Kotok: Yeah, so the 20, the 20 was a piece of marketing, not a piece of engineering, well I mean, to do a first approximation. The 20 came from, well the number wasn't-- somehow marketing came up with the number 20, but along the way we had adopted the operating system that BB&N had been developing over many years known as Tenex, which was a more modern thing than what we called Tops 10 had been. And marketing decided that to create sort of a separate market and not cannibalize the 10 market or something, they would create this thing called the 20 which actually did have a somewhat different- had some tweaks to the memory mapping system, that made it more flexible than the 10, so it was a little bit of redesign, but the major thing was it was packaged in tan cabinets that were low rather than blue cabinets which were tall. But it was pretty much the same machine. There was a--

Q: It was KL?

Alan Kotok: It was a KL. There was a follow-on smaller machine known as the KS10, the 2020 or something like that, but there were 2040's, 2060's, 2080's, there were all the KL10 in various configurations, and the 1080 and the 1090, were all the same basic guts, some differences in micro-code that had to do with handling page faults and that kind of stuff.

<Crew Talk>

Alan Kotok: So where we were was the--

Q: So that sort of wraps up, you weren't really involved in that.

Alan Kotok: Right, no.

Q: So moving on, you started talking about the Warring Research.

Alan Kotok: Right, Warring Research Labs, and VAX was the ultimate complex instruction set machine, and one of these groups, what was known as WRL I guess, Western Research Lab, had been working on a RISC design, whose name escapes me at this moment, but they were going--

Q: This was David Cutler's operation?

Alan Kotok: No, this was prior to Dave Cutler.

Q: Cutler was still working on software.

Alan Kotok: Cutler at this time is in the-- working on VMS, he's in the VMS, he's up in New Hampshire working on VMS. I remember Jud Leonard was part of this lab, Jud had been part of-- had worked on various, some of the machines that I'd been on, and so forth. And I can't remember the name of the lab director, but they had this project to build a RISC machine, which bore no relation to the VAX at all, but basically as a research project to demonstrate how fast you could-- how this design would be more effective. And I don't recall how far they got, whether they actually ever built one or prototyped it or what. But anyway, it caught my eye as being, I thought, a good idea, and I guess Bob Glorioso [ph?] had taken over the large computer group by this time and so he set up an advanced development group which I ran, I was the chief hacker there, and our idea was to build a reduced instruction set computer that was data format compatible with the VAX, so it would handle-- you could both emulate-- so you could emulate VAX instructions, it would get the same results, or you could compile directly to its native reduced instruction set. So part of the idea was to be able to emulate the VAX through RISC instructions, and we called this machine SAFE, as opposed to RISC. So, we did a lot of the ground work on that as an advanced development project, maybe spent a year or so at which point-- now at this point in time Jack Smith is running engineering, formerly of manufacturing, and I mean I like Jack but--

Q: But we know what that must be like.

Alan Kotok: Right, and somewhere around while we were doing this Cutler went to Jack Smith and said "I'm tired of developing operating systems, I want to build a computer, I want to build a RISC machine, and I want to go to Seattle and start a group and do this." And Jack said "Whatever you want Dave," and so Dave went off to the West Coast and formed his DEC West group in Seattle and started to do this. And sort of this word came to us and we said "Hey Jack, did anyone tell you that we're doing this?" and he said "Well, er, uhm.. gee, I told Dave, you know, and I guess you guys are out of business," <laughs>. So, now again, this might be a self-serving view of recollection, but I don't recall that there was ever kind of a bakeoff or any kind of evaluation by no design review kind of thing, it--

Q: A serious technical.

Alan Kotok: Right, it was kind of Dave is our star, he brought us VMS, and if Dave thinks it's a good idea we can't afford to do it twice so you guys go find something else to do with your lives. So I said-- I'd sworn off designing computers once, I'm now swearing off it twice, I'm getting into something else. And so at that point I went over to the storage group and became Storage Architect. Storage was doing these sort of early version of smart storage interconnected devices, and so I had--

Q: Was Grant Saviers [ph?] running it?

Alan Kotok: Grant Saviers, Grant Saviers ran Storage and this little wiry guy, sort of old wiry guy, ran the engineering, or the advanced engineering, another name that'll come to me. And so there was Richie Lary, who was well known in that group, became part of this architecture group of mine, or maybe we had parallel positions, I don't know, but I worked with him. So I had a small group in Colorado Springs, and a group, they were in Shrewsbury [ph?] at the time, so we basically worked on storage interconnects, and cache, how to do caching algorithms for storage devices and stuff like that, and this, whatever the hell the proprietary bus interconnect that we had. We had a thing that competed with SCSI, and typical DEC fashion we couldn't just do SCSI, we had to do SCSI one better but it never caught on.

Q: One better and non standard.

Alan Kotok: One better and non standard. So maybe that kind of pushed me towards standards, but <laughs>. So I spent a couple of years in that Storage group, I don't know that I felt wildly rewarded, the major memories were of a problem employee who I spent a tremendous amount of time dealing with, he tended to be mercurial and occasionally abusive and so there was just personnel problem after personnel problem, and so I don't look back on a lot of that with great fondness. Then DEC decided to go into supporting the telephone industry with well, sort of a two-pronged thrust, one was with administrative systems for trouble ticketing and all that kind of stuff, and secondly was the cell phone industry was cranking up, this was in the I guess the late 80s by now, middle to late 80s, and there were a number of infrastructure components, one of which was called the Home Location Register, which is the thing that tracked where your phones were as they roamed and so forth. And it seemed that we were going to get into the business of building these things, and they formed a telephone business group and Mahendra Patel [ph?(ok)] was the-- was sort of associated with the networking group and Mahendra Patel was the technical director of the networking group. And so he pulled me in to run the architecture stuff for the telephone group, and since I had been an old telephone person this all seemed like a lot of fun.

Q: You said "Oh this could be fun."

Alan Kotok: Right. So there we had a lot of the development being done over in France, and Sophia Antipolis [ph?], so we had a group over there, and we had a group in Merrimac doing some of the work, and some people in Littleton. And so I would coordinate these various groups and had a lot of fun going to France, enjoyed that part of it, learned enough French to be dangerous in the process. So, but DEC had, as we got into the 80s, they could never sort of stay focused on any thing very long.

Q: Long enough to make a real impact.

Alan Kotok: Exactly.

Q: Build a customer base.

Alan Kotok: Right, right. So at some point they decided maybe this telephone group should just flog All In One systems or something like that, and not be in the operational infrastructure stuff. And so they gave up on that, and so I headed off to the software group and worked for David Stone, the late David Stone, again doing a lot of sort of strategic product planning, I'd sort of gotten out of getting my fingers too dirty anymore, it was coding and this sort of stuff. So I was, I don't know, sort of chief troubleshooter for projects that weren't going right or needed "what should we really be doing" kind of stuff. At some point we had gotten into a deal to supply a huge transaction processing system for Nippon Telephone, Telegraph, I guess it's Nippon Telegraph and Telephone, to be precisely correct, and it needed somebody who could somehow pull this together, it was scattered and so I volunteered to do that, wound up in that for six or nine months, or something like that. And so now all of this has taken us up into '72 or so.

Q: You mean '92.

Alan Kotok: '92.

Q: Gordon is long gone.

Alan Kotok: Gordon is long gone.

Q: But Ken is still in charge.

Alan Kotok: No Ken left I think some time while I was in the software group, but I did get-- so the next move I made was into a group with that one could immediately tell was going to have no influence whatever, it was called the Corporate Strategy Group, and--

Q: By definition.

Alan Kotok: By definition. It was run by a woman, let's see, she was the granddaughter of the Luce's [ph?] of publishing fame, yeah. So but that was her middle name, she was something-Luce-something, I forget. But anyway, so there was this group which was trying to come up with new opportunities, and the management was all focused on Alphas okay, so they sort of didn't want to hear anything about other than how to sell Alphas, okay. But we plugged on anyway, figuring that they needed to hear some alternate views, and one of the things that had been gaining some enthusiasm in this group, although I confess I didn't have the enthusiasm for it, was this video on demand, that was going to be the big thing, and it was- there was this, remember that the telephone companies, the entertainment companies and the computer companies were all going to merge and this was the grand convergence, was the term, and DEC actually started a project to build a video on demand server for the cable industry. But sort of my epiphany was discovering the web, one of the guys from the network group who I knew came, you know, the traditional, came running into my office and said "Oh you got to see this web thing," this was like in '93 I guess, and I guess Mosaic had just come out. And I sort of looked at it and I said "That's it," <laughs>, "this is going to take off," and--

Q: This is something really big.

Alan Kotok: Right, and DEC has a great opportunity to be in this, I mean it was more at that time it was the internet in total, the web was viewed as one of the things, but DEC had sort of wallowed in this proprietary networking for a long time, and so sure enough, another one of the West Coast research labs had been championing the TCP/IP world for a while, getting beaten up, beaten about the head and shoulders by--

Q: That isn't DEC Mat.

Alan Kotok: Right, that isn't Dec Mat(??). But see, DEC was so big that there was someone around who knew about any technology you cared to be interested in. So tying it, sort of tying now towards this organization, so we started putting the idea for an Internet Business Group, to develop software products because our conclusion was that, guys, hardware is becoming a commodity, you really have to learn how to build volume stuff at low prices, get off of this proprietary kick, but they weren't listening to that. So we said well, maybe at least we could, you know, we had some people who knew about networking software and we could do networking infrastructure and networking applications and web applications and so forth. And so around some time in the spring, March, April, 1993, a group of us, a couple of marketing types and me and another engineering type, decided to take a tour to visit key customers to see how they would react to us going into this business. And our tour took us to Europe and one day we found ourselves in Geneva with some time and one of the guys said "Well Tim Berners-Lee [ph?] is right over at CERN [ph?], why don't we go talk to him," and so we said "Cool idea, we'll call Tim." So, "Hello, we're here, we're from DEC and if you have some time we can come talk to you," so he says sure. So we go

over there and Tim has recently had discussions with MIT and Michael Dertouzos [ph?] about forming this consortium because it turns out that Michael Dertouzos, who is the lab director here, was sort of coming off the-- he was at the tail end of this X Consortium which had brought in a lot of money frankly, and some fame to the lab, and he was looking for the next big thing for the lab. And so he was enthusiastic about starting this web consortium, and had talked to Tim, and Tim was not sure that this would get a lot of support from the industry. And so what our conversation amounted to was what would you think of such a consortium, would you support it, do you think others would, and so forth, and we were very enthusiastic. And in Tim's book [Weaving the Web] he recalls that event as what persuaded him to actually start the consortium. But amusingly enough he didn't remember who was at this meeting, it was this bunch of people from DEC, and so he walks into my office and he says "Well, you were with DEC, who was at this meeting, do you know who was there?" and I said "Yeah, it was me," <laughs> and the gang, and so, anyway, so I get to tell this anecdote at the tenth anniversary celebration of the Web Consortium which is next month.

Q: Oh very good.

Alan Kotok: So anyway, so the consortium got started, we did persuade Bob Palmer was the CEO at the time, that starting this group would be a good idea, we recruited Rosanne Giordano [ph?] who was one of the sort of marketing senior manager type VP's around the time to come run the group. And then things didn't go well, and I became the technical director of the group, and the word from on high was "come up with some marketing programs to sell Alphas", and we said "Well gee, we'd really like some engineering money to develop some products," and we had some ideas and firewalls were big at the time and so forth, and it was like pulling teeth to get any money. One of the West Coast groups had by this time come up with Alta Vista and trying to get any money behind it was pulling teeth, eventually they brought in some other manager but by this time, which was now-- so we started this group in '94, we did join the Web Consortium and I became the official person, liaison, representative, and got involved with some of the work, and by '96, the beginning of early '96, my wife was saying "You just don't come home looking happy," I mean, you know.

Q: You're supposed to have fun at work.

Alan Kotok: Right, right.

Q: You used to have designing computers.

Alan Kotok: And I said "Yeah, maybe it is time to get out of here." So I actually scouted around through some of my W3C connections and there was-- I was at the time enthusiastic about online payment systems that were going to be micro-payment systems, which never came to fruition, but at least at the time seemed as if it had some promise.

Q: What do you mean by micro-payment?

Alan Kotok: Well, so a system for buying content from the web in amounts smaller than credit card transactions could support. So this was before advertising became the primary funder of the web, and back in the mid 90s it was felt that that wasn't the way people would deliver things. So if you wanted a weather forecast you could pay one cent and get a weather forecast, if you could figure out how to pay the cent efficiently. And there was a guy at DEC who had some good ideas on that subject, and there was this little company that was pursuing this business, although sort of a step up in the dollar range kind of payment thing. A little French company and they were in the W3C working groups and I'd gotten to know them, and so I said "Hey, I'd be interested in joining your company," and they said "Oh wonderful, wonderful," and I had this meeting with their CEO and he made me this wonderful looking offer and so I went back to DEC and I said "I'm going to be 55 in November, I'd like to get out of here now, will you make me a deal where I can officially retire," and so forth and so on, and in light of the fact I had been there 34 years they said yes, and so. So I left but interestingly enough the employment contract that this French company wanted me to sign, my lawyer never became satisfied with, and so I said well, after my sort of last paycheck had been cashed, I said "You got to pay me, how about I just be a consultant for you, I'll sign a consulting agreement." And so they said okay, and so I started work with them. Again a lot of sort-of product strategic stuff kind of thing and they were French and they needed a little bit more American slant to their business approach and so forth. But this didn't last long because by early '97 they said "We're running out of financing, we've got this second round financing coming," but it seemed at least apparent to me that the original investors were never willing to cede enough of their stake to the second round investors to get a deal. So it was in April of '97, it was just at the time of the World Wide Web conference that was out in Santa Clara, "Well, we're going to shut down, we're going to file for bankruptcy next week," but the US guy who I had been dealing with said "We've got some money in the bank, turn in your time, you know, your invoice expenses, and we'll wire you the money before this happens." So I got out perfectly clean out of this deal, I got every penny that was owed me and then they were talking about various reorganizations and so forth and I said "Well, you know," I said "I'm not broke, I can hang in here for a while if you were to give me some assurances that if you reconstituted I would be in and reconnected."

Q: You could be part of the team.

Alan Kotok: Right. And the French management couldn't quite get its act together and so I walked into Michael-- well I walked into Tim's office. It was it over the other side of the street at the time, and said "I could use something to do," and he said "Oh wonderful," go talk to Michael Dertouzos, and I knew Michael from years and years so it was the old--

Q: From whenever.

Alan Kotok: So in no time flat I was here. And initially I had a part-time contract because the shock of what MIT was willing to pay compared to what I'd been paid from DEC, but I hadn't burnt any bridges at DEC so they were willing to give me part-time consulting so I spanned that for nine months before I became a fulltime MIT staffer, so that sort of transition helped. So here I am.

Q: Here you are working on the--

Alan Kotok: On the web <laughs>.

Q: Say a little bit more about what you're doing here.

Alan Kotok: Right, so when I first came here Michael Dertouzos said “This organization is growing, it really needs someone to attend to the membership,” and I said “Well, I can do that too.” So what I again had to figure out was that the problem, why the membership wasn’t rising as fast as it could was because no-one gave a damn, so all I had to do was--

Q: Was give a damn.

Alan Kotok: Was give a damn and grease the skids and the membership went from 300 to 500 or so over three years or so. And I wish I could claim credit, maybe I have some credit but it wasn’t that I mounted this grand marketing campaign, it's just that I made it easy for people to join <laughs>. So, I took a short stint as the head of the Technology and Society Domain when the guy who ran it left and I didn't find that was totally satisfactory and so I went back into running the System Group and the business of the site. So that’s what I do.

Q: That’s what you do.

Alan Kotok: Yeah.

<Crew Talk>

----------END OF INTERVIEW----------