An Interview with

Bobby A. Creech

OH 249

Conducted by Anne Frantilla

on

27 June 1987

Pasadena, CA
Bobby A. Creech Interview
27 June 1987

Abstract

Creech, manager of Product Technology Operations at Unisys' Pasadena plant at the time of the interview, discusses his career at the Burroughs Corporation since 1962. Subjects discussed include: his management of software and programming on the B 6500; coding for data communications on the B 5500; his responsibilities as the Terminal Systems Group director; and his management of the Medium Systems Group at Pasadena.
Bobby A. Creech Interview
27 June 1987

Tape Index

Tape 1/Side 1..........................1
Side 2...............................18
BOBBY CREECH/ANN FRANTILLA INTERVIEW, JUNE 27, 1987

C = Bobby Creech        F = Anne Frantilla
// = Unintelligible passages

SIDE ONE

F: This is Anne Frantilla talking with Bobby Creech in Pasadena on the

C: The 205 was the first product when Burroughs bought ElectroData. It was a
drum machine, 4,000 words. A comparable IBM machine with its 2,000-
word drum. The first machine developed under the Burroughs ownership
was the B300. On the West Coast, which is where I claim the commercial
computer industry got started, this is the school of people that pioneered
the circuit technology. The B300 was the first transistorized machine.
There was a 220 which is a vacuum tube machine, decimals oriented, much
better functionality than the IBM 7070 with the same kind of architecture,
similar architecture. One of the things it was very well known for though,
was its ... Burroughs ... its algebraic compiler, ALGOL, which was ALGOL
58. That was a very, very well-done compiler. As a matter of fact, when I
was being recruited, that's one of the things that was demonstrated to me.
I first started being recruited in '61. It took me a year, took them a year to
get me to come out here because I would never move to Southern
California.

F: Where were you from?

C: Dallas. In any event, they showed me this compiler that would compile in
one pass at cardreader speed, a very fast cardreader in my experience has
been as an IBM customer, and a very, very slow, bulky, cumbersome
FORTRAN compiler. It just blew my mind, I knew that they had something, software technology. Now the B300 was again ... well, first it was the first transistorized, first solid state commercial computer in the company. It was quite well developed as a financial application package. Modifications of that are still running and it's called a Thrift package for handling the transactions of savings and loans. They were originally stored on the B300 by machines. Now there was a tangential development ... the 500, by the way, is just a 300 with new paint, more or less. The B5000, however, was a radical departure from that kind of machine. It was a radical departure from everything in the business. However, it should be pointed out that it used the technology that was pioneered on the B300. // where you see the two paths separating, two schools of thought, two schools of computer architecture and development.

So they used the same transistorized logic, discrete component logic as was used on the B300. And by the way, the B300 was the first to use the head-to-track disc that was developed in really thinking of the B300 as the host machine. The head-to-track disk was developed in the Pasadena plant, tape drives, everything for printers, ... our proprietary tape drives and the original head-to-track disk was developed in this plant.

The B5000 was a scientifically-oriented machine. Certainly everybody knows it was an ALGOL machine. It was so much patterned after the needs of the ALGOL language that it basically neglected everything else. The people who had to implement the COBOL compiler for this machine, the B5000, had such a painful experience ... and oh, by the way, their compiler was better than any in the business, in spite of the fact that it was an ALGOL machine. They a one-pass COBOL compiler. It was very fast, it was very efficient, and did a tremendous, big hunk of the
language. It was very complete. But it was very difficult to bend that
language to an ALGOL machine. And they said, gee whiz, wouldn't it be
nice to build a machine that was hospitable to COBOL? Those kind of
sentiments coincided with problem in the marketplace with the B5000,
which wasn't selling all that well, and there was actually a planned
shutdown for that machine, and the management was ... it is going to be a
commercial flop, what can we do? We were wrong. Therefore we will go to
the other extreme and we will go develop this COBOL machine. That was
the beginning of the V-Series. In fact, the guy that ran the department for
the COBOL ... actually, it was the section, I should say ... he ran the section
of a dozen people or so to do the COBOL compiler for the 5000, became
the head of the software department for the B2000, 2535.

F: And who was that?

C: John Hale. And he supported that stone. And he headed up the
architectural, the software side of the team that designed this whole
architecture which we still use. Once again, that hardware group also
pioneered a new circuit technology, circuits and packaging with the
introduction of the first integrated circuit.

At the same time they got this going, the B5000 took off, and IBM
announced that the concepts were o.k., like multiprocessing, virtual
memory and all those sorts of things. So that product did pay for itself and
make some money.

F: By that time the 5500 was introduced?

C: Now the 5500 was a minor variation of the 5000. It was a very minor
variation of the 5000. The real significant change that happened on the
5000 was the software. Not many people know this ... the original MCP that
was developed at that time never really made it. It was called a drum MCP.
It was written in Assembler. Not many people knew there was an Assembler on this machine. You know, got to hand it to those guys. They were trying something that had never been done before, and there were a few loose ends, I think, that never got tied together. The good news was that the head-to-track disk came along and we needed to integrate it with the host and we did that by developing what was called a disk brace MCP, which we wrote in high-level language. By the way, that was done by four or five people. The ALGOL compiler was done by four or five people, so the legacy of very small groups of people being able to program very significant systems software packages I guess speaks not only for the people, but also, it mainly speaks for what a difference architecture makes in being able to program the system, and what a difference the right kind of high-level tools makes.

Now those guys, since these guys were all what are now the V-Series people, were all busily engaged in the 2500 and 3500 product development, the 5500 people were looking for an encore, and proposed doing a 65 (what else?) a 6500 and first proposal out of the box was a "well, we'll just make it with the new technology and we won't change anything. We'll just use the same software. There'll be no software investment, and we will basically" ... A mental picture was drawn to the management of lifting up the software, unscrewing four bolts, lift up the software, slide in a new processor, and there you go. It didn't work that way. As a matter of fact, one thing led to another. There were several versions of the 6500 proposed, one of which was to make a 64-bit machine and it was very, very big oversimplification made of what it would require to change the software with it. Well, when you stretch all the tables by a third, 48 bits to 64 bits, it's really quite misleading. And the management bought that, and after they bought that,
we said, well, if you buy that, changing the format of the descriptor, which was letting the cat out of the bag, then there's no reason to make it 64 bits. We can squeeze it back into 48 bits if we ... and that means total new software. Now the real key there, though, the real fundamental reason for changing was that the 5500 had a 6-bit character orientation. And that was passe, that was not going to fly, so we had to go to an 8-bit character representation for the character manipulation aspect.

F: Who was drawing up the proposal for the 6500?

C: Well, I was head of the software side of that effort. I was the department manager. Highest level programmer in the place was the department manager. The engineering people ... there was a manager of engineering, and he had ... I had a counterpart for the ... on the hardware side ... head of the hardware department, guy named Merv Halleck. And he and I and our people spent some of the most enjoyable times of my life designing this machine. We got outside consultant, Bob Barton, who basically conceived of the 5000 in the first place, and he was a ... certainly the guiding light, the genius behind the 6500 design, of course which is now really, fundamentally unchanged, which is the A-Series. I happen to have the written proposal on the 6500.

It really tickles me to look back at these and see how we proposed products in those days. You notice there's Plan A, Plan B, and Plan C. Here he was. You always had to give those guys in headquarters a choice. Let them think that they had a choice. Now the head of engineering used to tell me, he says, when you get right down to it, we estimate the cost of the machine by the front foot. He says if you can ... because it doesn't matter what's inside there, what really matters is how big is it? If you measure the front of the cabinet, you can tell how much it's going to cost. And I guess
he's right. One of the big surprises from ... we always talk about the fact that we had an integrated hardware/software code. We didn’t. Not before the 6500. The 6500 we really had what I would consider the ideal climate, the ideal attitudes on both sides. And we really worked together without any turfsmanship, without any conflict really at all, to develop what we all considered the best machine. Before that, on the B5000, the hardware guys didn’t know the software guys. In fact, the software guys were in Marketing. We were resident in the same building, but we were not close. We did not even know who the hardware people were when we were developing that software.

F: I was Talking to Roy Guck yesterday afternoon. He had a solid experience in hardware and then he moved to software, and that really helped him out in a lot of programs.

C: That guy is phenomenal. Roy Guck is phenomenal. We tell war stories about that guy.

F: But it didn’t seem like a whole lot of people have that expertise in both areas of hardware and software.

C: No. No. I guess that I would have to say, making a editorial statement, software is a separate discipline. It’s a different discipline than ... well, I don’t like to say // hardware engineering, // engineering, and there are some who say soft ... that's because software is not a discipline, they're not disciplined. Discipline is discipline whether it's, no matter what field it's in. But it is a different job that really needs to be recognized as a different job. And one of the things that really bothered me is the notion of software engineering. It misleads so many people as to what's involved. And people go down the primrose path and I can point out project after project
that was basically a failure because people thought in software engineering terms, and thought they could bring it to be ...

F: Without weighing what the major problems would be, what ...

C: You can't partition the job the way they do in hardware engineering, let people go out and let people work on this little piece and put it all together and it hang together. That has never worked. It never has.

So we proposed several versions // This is probably the 64-bit version, I don't know. I haven't really read through these ... lately. I have to say one other thing. We could have designed these things forever, and that's one of the problems ...

F: // to get a perfect machine.

C: Irv Houck, I have to give him the credit for one day saying ... well, // Another thing I forgot, all the I/O controls and development for how you hooked I/O into the system is always pioneered by this group of product, this family //.

So all the peripheral controls were developed by the so-called medium systems guys and the A-Series, what came out of the A-Series guys, simply concentrated on the process there in the architecture and software and pulled the pieces together off the other developments.

The head-for-track disk was so successful that they built a 400,000 square foot plant in West Lake to build those things, and they thought they were going to even build that many more, so they built a place called Mission Viejo to build disks. I don't know, but that was a very short-lived plant at Mission. I think it was developed to build packs. It was almost brand new and it was quickly ... all the production from there went back into the West Lake plant. That thing set down there a white elephant.

Meanwhile, here in the Pasadena plant, we've had thousands ... three thousand people in this building. Now we've got 450 now. The factory just
moved last year. We would have had maybe 800 and it would have been full. But we had 3,000 people here. You couldn’t park within blocks of this place. It was unbelievable ...

F: I’ve seen pictures of it after the acquisition and it just seemed like it was looked upon as sort of a state-of-the-art facility.

C: You just marched out across the lot back here. There was the A building built first—this is a split-level place. If you’d like, I’ll tour you around, because I want to show you the original board room. The original GM’s office is still up there. But at that time we had all the A-Series type machines being done here, the 6500 was ... I was in that school, and we were doing that all in this building. Meanwhile, the B2000, 2500 people were doing all their things in this building. The B1000 was being thought of, and it started here, and all at once it was like an explosion happening. Somebody finally realized, my God, we’re out of space. And there was no way that all these things are going to happen if we all try to do it in this building. You were not getting the management attention; the general manager had his pet area that he paid attention to; the other two were coming out on the short end of the stick, the resources. So all at once they just spun off in all directions. There was a move to move to Santa Barbara, which was attractive for all kinds of environmental reasons. It’s a nice place to live, etc. Those guys, one group of guys got there way. As a matter of fact, it turned out that they were going to move what is now the V-Series group up there. In fact, they did move some of them. While Ray Macdonald was on ... they did this so fast, it was done so fast, that when he got back from Santa he said, “What are you guys doing? You’re crazy,” and actually moved some of them back, the ones that had gone up there already, because he said you’ll do the new thing up there, the B-1000—-it
was then called, I think it started as the 1500. Those guys go up there and start an engineering organization. Don't move an already-existing engineering organization up there. So the guys had to come back to here. Meanwhile, the A-Series guys, and I was in that group, decided ... well, we had to get space ... we had shipped the first 6500 from this building. And of course they were built here and were shipped out of here, but we were out of space. The software crew had a long way to go. At that time I had moved on to be what was called the Manager of Programming, which included a functional responsibility for the systems software, but also they gave me the CAD stuff, what we called design automation in those days. And so I had a more general job than just the A-Series by that time. They moved off then to a warehouse, a converted warehouse, Venice City. Venice--I don't know if you've ever seen that place.

F: No. I've heard of it.

C: But it is an industrial warehouse district, with all the images that come to mind. And that's how that group ... that's when that got split off. So these guys went up there, and these guys went to Proctor. The guy running Proctor ... the first thing that comes to your mind when you go in there is how do we get out of here? And he saw this white elephant called Mission Viejo sitting down there empty. They'd been trying to sell it and they had no takers. He made an economic case, which you can make an economic case for anything, to move everybody down there, he drove the people down there kicking and screaming. But he moved the whole operation. And to his credit. And these guys are ... this is before the big real estate sky-rocketing prices, so people who went down there have houses they can't afford today. If they sell them, they're rich. Anyway, that's how the Mission Viejo plant got the A-Series.
Meanwhile, they closed down the ... they made an announcement to the world, no more B1000s, no more development.

F: That's what they thought.

C: Right. They're still building, they're still selling the new ones. They moved the manufacturing into here at that time. They also moved it over to Liege in Belgium. They're still building them. They're still building them in Liege. Now I heard just last week that there's only reconditioned--they're not building new ones. But I just heard that.

Meanwhile, the B-Series manufacturing was moved down to Mission Viejo late last year. I've got another set of cells which basically say this: Twenty years ago the ratio of manufacturing space to engineering space was about ten to one when we were building machines out of vacuum tubes and such and so forth. Ten years ago there was a major ... maybe a little bit more close ... somewhere between ten and fifteen ... there was a major realignment in the corporation. I remember when we realized, and there was a lot of publicity we got // we were no longer in the electromechanical business. We didn't need all the space that we had, and there was a big shift and they said that nowadays the ratio is 50:50.

F: That was about what time?

C: I'd say 1975. And now it's 10:1 engineering to manufacturing. The ratio is completely reversed. And, let's face it, the semiconductor guys are building the machines. We're working on a machine right now, where six little chips are replacing what filled a room when I came here. And that doesn't make me feel old, but it makes me say gee whiz.

F: It's probably the industry which has changed the most. It's just incredible the way it changed.
C: This build ... I ... we had a retirement party last October or so for the people that ... we had it upstairs for the people that were retiring early--there was a bunch of them. It was pointed out some of them came from Electrodata. They really have seen the entire computer history in their own lifetimes, and been involved with it. It's astonishing. This is kind of more of a breakdown of the products specifically and it's probably not all that complete. We are now, of course, just the Pasadena Development Center which does all the B-Series engineering and software. And by the way, we're still the pioneers, the leaders in circuit technology in R&D. The circuit technology that we're doing for the B-500, for example, that we're going to deliver this year, is ahead of anything else in the country. In fact, Sperry is going to use it on their Super 1100, the big high-end 1100-Series. We still have that going. We have a very strong user base, very loyal user base.

This is last year's survey. There should be another one out momentarily--customer satisfaction done by DataPro. You've seen that and it showed Burroughs versus Sperry. This shows the B-Series versus our internal Burroughs. And I underlined the places where we came out on top. Well our customer base is extremely loyal. We made our mark in the financial area. The people in the plain vanilla business data processing world loved this machine, this architecture. It's easy for them to understand. There's no fancy things about stacks, although there's a stack in the machine. It wasn't sold that way. It was sold as a COBOL decimally-oriented machine, even to the extent of having decimal addresses in it. Everybody now will admit that there was someone who said that back in the days when it was started that it was wrong, but customers think decimal is natural. It also is a very easy-to-understand system. We just had a visit this morning from that bank in South Africa. They have four of our biggest
machines right now, and they were one of the first customers for the B500. The manager of the place has told me that he has enormous programming staff. They’re one of the biggest banks in South Africa. He’s got 50 programmers, including all the management to run his whole shop. His competitor across the street, about the same size place, has 600 some people to do the same thing—guess what equipment they’re using? IBM. So it does take our people very much ... our customers very much less resource. That’s an order of magnitude of course, less resource to do their job. And they’re very happy with it. It does what they want to do, it’s no mystery, it’s a performer, it just grinds away. He said he was told in Detroit that our B500 was three times more reliable than the B300. And he said he had a problem with that because his B300 has a 99.8% availability record. How’s it going to be three times better than theirs? It’s almost like three times 100%. They’re very pleased with it.

F: How far back can customer loyalty go?

C: How far back?

F: To the first B series?

C: I think it probably changed over the years, but we have a lot of ... well, let’s put it this way ... one of the things that came out here is over half of our systems are over five years old, which is good news and bad news. One is that people have stuck with it, but the other thing is they’re ready for a change. They’re a ripe market for picking, and if we don’t pick them with our own replacement products to modernize them, somebody else is going to come after them. So it ... they go back a long ways. There’re a lot of customers. We have still in his presentation that was made on strategy two weeks ago to Blumenthal and his people, it was pointed out that this base, the installed base of B-Series is still ... has more dollar value than the A-
Series, and they spent ... almost quadrupled ... in R&D their spending more than quadrupled the amount of products // in spite of that, and has been a strong marketing emphasis on A away from B, and yet we still have the bigger base.

F: The B Series must, per unit, cost a lot less, though. //

C: Well, there's such a wide ... because they have developed so many A-Series models--there're not that many B-Series models because they haven't funded it. But in the last ... there's A-Series models that are much smaller, and they're ... up until the B500 there's some that are much larger although the B500 will be is an A15 class. They get almost double the price the price on an A15 that we're charging on the B500. So we haven't had anything out that expensive. There has always been, well, for the last decade at least, // has always been a more expensive A-Series model, and that's where the real revenue comes in when you sell those big, expensive //. But you can't blame it on that. People just like this one. In much the same, and as far as units, I guess the A-Series is just about overtaking in numbers of units installed, but again, they got these little ones, the little A-3s and 2s and so forth. So that makes it ... but then you look at that. When you combine the A and the 1000.

It's a funny thing, I find myself talking about this because technically and emotionally I feel much more a contributor to the A-Series architecture. I feel like I have to invent that. I feel like a very key contributor to that architecture and design. I can still go in there and read the code in that operating system and write code for that machine. I can't do that in this thing. [a B20?] But I like to think I'm openminded enough to face the reality that this product line is very successful.
F: Well, after hearing about the products over the years, I'd like to know how you fit in ...

C: How I fit in?

F: Your different tasks and projects and // what you're major contributions were.

C: On the 5000 I came to work here as a senior automatic programmer. We used to laugh and say automatic programmers were the best kind. Strange job titles. To work as one of the five-man teams, the five-man team that did the ALGOL compiler. We had a section manager and four workers. The section manager also had responsibility for FORTRAN. So we had one guy, a different guy, writing a FORTRAN translator. We did that project, and as soon as we finished that, and the drum MCP was all over the floor, and the machine had been shipped but it wasn't really working. We all signed up to do a new operating system that describes a disk-file base. And the same guys except for one person--we had one new guy and one of the old guys left--did that operating system in the next year or so and delivered that. It was a very successful ...

F: It sounds like a short period of time to do that.

C: It was, it still is a //. We did everything. My management would ask me later on when I was in management to do a compiler, I had a rule of thumb. I'd say, o.k., it takes four guys one year, four people one year, and that's what it costs to beget a compiler. Today the numbers are dramatically different. So somewhere something's been lost. I'm trying to find ways to regain that around here and do things ...

F: So it takes more people to do the same amount of work?

C: Across the line. Now in Pasadena we haven't lost it. I guess if you looked up and down the coast you'd say that Santa Barbara was just disappeared.
Probably had the best productivity and had a small group, a very small group, like 50 people, and they were all programming and management. They did pretty good stuff, but it was a very ... they were probably below critical mass for doing anything dramatically new. And you come down to Pasadena, we have less than a fourth of the resources to do programming systems as they do in Orange County for the A-Series, and yet you can't really look around and say there's four times as much stuff. We're doing quite a few things with much smaller numbers of people, although from my point of view, compared to back then, we still have more people than I think is necessary to do the job. There are some extenuating circumstances why it takes more people, but not that much.

We had a meeting yesterday with some people that are doing a machine, on the engineering side of the business, for example. I couldn't help but note that we were talking to people from Mission who came up here because we're doing two machines, a B-Series and an A-Series, the A-8 and the B400 that are roughly the same size, but have different architectures. Couldn't help but notice that they had a department for each piece of their system, where I have a department manager that has sections for each one of those. Interesting contrast. Anyway, I'm getting back off, wandering.

After we got that done, shortly, that was about say, elapsed time of 1962 to '67. In '67 I went to a very brief period of time as the equivalent of a section manager, although I was not given that title. I was called project systems designer, but almost immediately I ended up being the department manager because the department manager who was here was promoted to a job in Detroit. I had at the time the department for all of the 5500 software. And I use the words 5000, 5500, 5700 interchangeably. I
probably say 5500 more. When we had ... with that department which was probably 20, 30 people, we had all the systems software for the B5500. We did the first file security system in business. We did the first data communications system on the B5500. In fact, I personally coded all that. I put the first data communications up, implementation on the 5500. I remember Ben Dent and I, as a matter of fact, teamed up to go up in ... we had a sale on DataCom for Chrysler in Detroit. They had Honeywell 120s all over the U.S. parts depots. They had a B5500 installed in Detroit. They wanted to hook all these together with modems or data sets from headquarters where the host 5500 was to nightly update all the parts inventories through these Honeywell 120s out at remote sites around the whole U.S. We thought that was interesting application and we needed to be able to stall it. Ben had never written a COBOL program, so as an exercise we invented the idea ... I claim we invented it, I never heard of it before ... of having ... since we had a multiprocessing machine, and maybe that's why it never had been invented before ... Ben wrote a program in COBOL that did what the Honeywell 120 was going to do. I wrote a program in ALGOL that did what the 5500 host was going to do, and we ... one program went out to the data communications subsystems, back in to the data communications subsystems ... talked to itself. So this program written in ALGOL talked to this program written in COBOL and for all practical purposes this ALGOL program really thought it was talking to a Honeywell 120 outside. Now it turned out to be invaluable because when we installed the systems, we made sure that worked in back-to-back mode, we called it, and sure enough, whenever the phone company ... I mean, this was in pioneering days when data sets didn't work and the phone company didn't even know how to make them work ... we knew there was going to be
finger-pointing, so that any time Honeywell says it's your end that's not working, we could run this program and say no, this works, our end works. You know. We could isolate the problem. And we had a very successful installation there. That's used all over the world now. I mean, everybody does that now. Back-to-back is the ... I mean, you've got your own world under your control then. Everybody does that now.

It wasn't a product. Not a standalone type. Just a way of doing business, which you could do if you had multiprogramming. When I think back of some of the things that we take for granted today that ... I remember being in the discovery process, and necessity being the mother of invention, when we first did the ... when we were trying to get the ALGOL compiler working on the original B5000, the very first one, we had unbelievable difficulties. We were building ... we were working on the first machine where the operating system had to work or you couldn't do anything. I mean, it was built to make the the thing easier to use, but it only worked that way if the operating system worked. Now if a tape went off the end of the reel, the whole system stopped working unless the operating system successfully handled that error condition. Or if the tape had a parity, etc., etc. If the card reader went empty ... it was the first machine, first system, where the operating system sensed when a card reader when empty and printed something out on the //. People take that for granted today, but that had never been done before. So what we were faced with, unless the operating system worked, how'd you get your work done? And the card reader was not that reliable, and it misread cards, and tape handling was atrocious without it //. And so we said, well gee, getting this whole tape to read is going to be impossible. We need a way just to merge in ... I'm sorry, the card reader was worse than the tape. We could not read
a whole deck of cards. We put the deck on the tape and said let's just read the patches in against that source file, and we invented merging patches. Nobody ever had to worry about that before. All those kinds of things. Just...
I'm sure I've forgotten more than I remember about the kind of little things that we invented on the fly because it made our jobs easier and in turn made the customers' job easier. It was an easy-to-use system. Job control cards--the drum MCP had horrible job control language. It was so difficult to make those control cards that people carried patterns around in their pocket because they didn't know how to make one...I got one that works, and you'd use that. So when we did the disk file MCP, we invented nice, reasonable, easy-to-understand mnemonic control cards, and we could put them in in different orders and they worked, and so on. All those kinds of things.

Anyway, then we invented the 65. We said, gee, we got to have a new machine. We got to have a follow-on. And we proposed the 6500 and I was department manager of programming, and that was just another one of the things we had to do. At the same time we did subcontract out the development of

END OF SIDE ONE OF CREECH INTERVIEW.

SIDE TWO OF CREECH INTERVIEW

C: He was one of the guys that worked on the ALGOL compiler for Burroughs and the MCP. He knew what he could do. And he and a couple others came in and wrote this time-sharing package for us. And that was successful, commercially successful // We then started on the 6500
development, mostly in the architectural--the design of the machine spent
... we formed two departments, then two sections. It was still only a
department for all assistant software. Ben was the section manager for the
6500 side of the business. Ira Burchison, you may or may not have known,
he was in Detroit, was the section manager for the 5500 software. We went
through the design. We spent, I don't know, a glorious year back in '66. I
must not have been ... I was a department manager, I kind of remember that
happening in October '67, so maybe I wasn't a department manager at this
time. I was still free of managerial responsibilities.

Anyway, I remember working on the design of that for at least, let's
say, a year. Very, very memorable pleasant phase of just designing and
talking on how was the best way to do things. I considered complete
freedom and the ability to do it right just a very pleasant experience.

We used to make a very, in fact I wrote it in a paper I wrote on the
architecture of the 6500, that--I think I put it in there--that there was no
conflict--I know I was department manager before that design was over,
because I remember there was no conflict, no technical decision that I
couldn't resolve with Mike // in the hardware side. We had to take nothing
to our common boss. We just trusted each other implicitly and neither one
of us was trying to win. It was the most cooperative effort I've ever been
involved in. And then we started out to building it, and started recruiting
because we did have to draw up a complete software package. And I
remember at one stage of the game Ben's section was 50 or more people in
it and we couldn't organize, we had no structure. I was the department
manager and there was a 5500 section and a 6500 section. And partly
because of my pushing to fix that, I was promoted to a higher level and that
let Ben be a department manager for the 6500. And I had functional
responsibility for that and the BA. I had the interesting BA experience for about a year, and we did go through to shipment of the 6500 in a year. Then they moved to Proctor and I went out then to La Jolla and started a research lab, called the Interactive Research Company. I proposed this and they bought it.

You could say in one sense it turned into ASDO, if you've heard of that. I had this reputation, I guess, for // researcher, because I was always interested in designing these things. And I guess I took it as a vote of confidence that they let me go off and be that, and I pulled a little fast one on them. It was kind of like going for the 64-bit machine and changing it back after they said o.k. I proposed that we do it in Salt Lake City, that we set this thing up in Salt Lake City. And I made all the reasons why. I went up and interviewed their ... they were going to start a research park up there near the university. And I knew professors in their ... Bay Levin of Levins and Sutherland was to be head of the electrical engineering department up there. I needed him. And // knew a lot of students there. We were quite closely coupled with the ... all the EE and computer science faculty and students, kind of one of those relationships we developed. And I proposed that we do it there, and that was fairly, and you can't be looking for a boondoggle if you're going to Salt Lake City. So they bought it. They approved it. After they approved it we said well, actually (we all had discovered La Jolla), and we said actually there's some publicity came about that time with the Mormon Church and Blacks. We took ... // you know, Salt Lake City might not be the most appropriate place for a Detroit-based corporation. // And in fact, since the University of San Diego was getting one of the first, if not the first, 6500s, // In fact, Ken Bowles, I think,
developed. So we said it might be a better place to go down there. So we went down to La Jolla and we set that up.

F: And what kind of research center was it?

C: Called Interactive Research. It was based upon the principle that we were going into an interactive environment where people would sit at workstations and do their programming and their applications which we take for granted today. In those days there any. There were no terminals, there was nothing. And we wanted to go off and explore audio, synthesized audio, // disk, etc. Video, the whole range. Nothing was happening. It was all just a batch-oriented. And we knew that that was going to change. And we did some good stuff. As a matter of fact, see, I left there in the early '70s and I came up here to Pasadena. I was not ... I worked for the general manager here and my job was called manager of new product development. We had an engineering manager and a programming manager, but again there was nothing new coming out.

And I came up here and we were scurrying around for new products. I got a group together and we finally split up into two parts. One was I wanted to really ... I had a personal realization where I had pictured myself as a scientist all my career, and yet we didn’t didn’t sell the scientists. I had a hard time giving up ... I had been on sales calls at Lockheed. They weren’t having any. I realized one day that we really sold most of our products to businesses—bankers, etc. Especially bankers. And I realized that I didn’t know anything about them. So I said, I’m going to find out. One of the things I set up a team of people to study the banking applications, to see exactly what made a bank tick, what kind of information processors they needed. We interviewed ...
'72, '73, '74. I know the dates because I've got a thing ... I wish I'd have thought to bring that to you today ... Ben may have one if he ... if you're going to see him. I've got one now that's burnt. We published ... I'll tell you about that when we get to banking. We sort of ... we interviewed ... we went into the head guys at the Bank of America in San Francisco, Wells Fargo, we interviewed branch managers, we interviewed Security Pacific branch managers to see what their life was like, what they needed. People from headquarters would that were responsible for developing the overall data processing //, the big banks like B of A. We actually got close enough to the B of A branch down here to let them let us have a guide, have free reign behind the counter in their operations department and we just studied the boo out of that. We also got in the credit union somehow. They did all their stuff. And one of the things we tried was ... I decided what we needed to be able to do was to find a way to represent what an application is, and banking was as good an application as any to study. We developed a representational system, and I would love to know where all that material is. It was taken--one of the guys that worked for me ended up in a ball-out job over at Croyden where we bought this automatic teller place. It had a programming job that was all over the floor and I recommended this guy. He went over there and he took this with him and through that representation was able to really get some order into what they had to put on that remote teller. And they used that very effectively.

I guess I'd have to say that that was the upshot of that side of the task force. The other side, we really talked about what was really the product. We really believed we were in the communications business, not the data processing business. All this is about is communications. And we tried to apply their communication models to business and we kind of
looked inside this plant, their manufacturing. How would you do it if you really viewed the problem as communication? There is not all that much arithmetic to do. Things don’t move as fast as electrons move and what is it that makes us think we’ve got to have //? So we looked up things from that angle. We finally came up proposing a product, and the reason I remember the date is that there’s a date in this book, December of 1974, where we proposed a business communications system. There are words bandied about in the world today like Voice Mail and all that. We had those in the system and we used our knowledge in storage of video of digital audio on disk, and all the things that Burroughs is good at--storage management, etc., and we proposed the very ... I mean this book was our product, you might say. We worked very hard at making a very professionally-presented pitch. When I say book, I mean a published book. All original art work, binding, the whole works. We knew we had to get their attention. And we did. Fred Meier will tell you about that, if you know Fred. He was ... I remember making that presentation to him in this board room up here. Way back there in the ... probably early ’75. He liked it. You might ask him if you do talk to him about it. Why he didn’t do anything about it. It was very ... as a matter of fact, when the new regime came aboard, Stern, Blumenthal, etc., somebody who knew about that drug that thing out and Stern has actually read that book, and Lee Chevelle was in Detroit at the time. His comment to me when he read the book was, "You ought to be fired." And why? "Because you didn’t quit and go through this on your own." And as a matter of fact, I think the Advanced Systems Group they went out in Boulder was trying to do the same thing. We proposed a plan in a product program in that proposal that would have developed a product
that in about, well, by '84 we would have been the premier player in that game.

Now say that maybe we shouldn't have it. A lot of people take and they pioneer something new. I said you take your lumps. You've got to be awfully lucky and you've got to spend a lot of money before you get there, but ... One of our strongest, one of our biggest weaknesses in the corporation is in the communication area. We don't have that //.

Well, what did I do after that? Went to work and they formed a new group called Terminal Systems Group. And I was asked to be director of programming for that group. But I didn't want to move to Detroit where the headquarters were, so I was allowed to stay here. I was allowed an office upstairs and I'll show that to you. Actually, it was the office I was already in. But it turned out to be very much of a travelling job because the place was scattered. It was in Croyden, England, and Villiers, France, and initially even Belgium //. But I worked in that group for a while and we tried ... basically as though Advanced System Development Organization was ... you say was a spinoff of // or whatever of Interactive Research Center. And it was down in La Jolla.

F: So how long did Interactive Research Center ...

C: Exist? That's a good question. I know it was there in '70 ... well, see, I have a problem between ... if you lump them together as one entity and they just moved locations, which I think is a fair way to look at them, ASDO was done away with just a few years ago in '80s when Blumenthal and Stern came aboard. I would have to guess about '82 or '83 that they were disbanded. Same time Austin Research Center, all that kind of disappeared. There was a basic strategy in the corporation--they don't believe in research.
Well, Paoli was a research place, originally. That was called the Great Valley Labs and the Paoli Research Center, and all those sort of things. But they have turned into ... from my perspective, they have turned into Tredyffrin, which is a computer plant that is, by the way, building the architecture that was developed for the 6500. They used to be a research center. And there's published history about Travis and all that sort of stuff. I think if it's not a stated policy, it's very clearly understood implicit policy, by-policy, because we don't have people that could do research. So any way, that's gone away. In fact, we even ... Sperry was in MCC and we bailed out of that //

F: Right. I was very surprised to hear that.

C: I don't know. We consciously ... well, I was very ... I wasn't surprised because I was kind of close to the action when all that was happening. I was in Detroit at the time, and I knew, I saw the mentality in the discussions about that and I quickly saw ... they made a deliberate decision in the first place not to join MCC. Therefore, when they bought Sperry they just followed along their normal course. I knew they didn't believe in research. I've heard some of their ... some of the players, I've heard their opinions. And they don't, they're not going to do that.

Anyway, I stayed in that job and in 1977--a few dates I can remember-July 1, 1977 I was still in this building. However, I had nothing to do with this organization, and they had a serious problem with the program activity, and they weren't getting anything out--no schedules //, no products were coming out and they put me in charge of the programming group. And on the V-Series, the one called the V-Series. We developed COBOL-74, DMS II, they didn't have the DMS II on the product line. There was a COBOL compiler, DMS II, RPG. We had a pretty good couple of years. So good, in
fact, that I got promoted // to be general manager of Plainfield, which is now
Plymouth. And the reason I got that job, you might have seen //

F: Was work in Plainfield Components for a while, or not? //

C: There were two pieces. This was this thing that built in plasma displays. //
memory. Plainfield was a very interesting ... again, that was the place that
Burroughs a long time ago. When I was there they had kind of split off a
piece that was doing terminals. They did the PV830. By the time I went
there they had done the PV830 and it was decided to have a replacement
cultured end product called the MT983, 985, // terminal. And the guy that
was running the group decided to take the work that was done down in La
Jolla and make a product out of it. And that's how that came and that's how
I was kind of connected to that. So there was a product that was
engineered in La Jolla and we were going to build it in Plainfield. And we
also built printers there. Actually we assembled printers. They were
engineered in France and had been engineered in Plymouth. They shut
down the Plymouth engineering for // desktop-type printers. But they
things are recorded. That was a scattered ... that was another one of those
crowded deal, scattered over five facilities, // and for that reason we were
building, getting to build ... that was one of the enticements--we had a
brand new plant in a nice area called Flemington. I actually was there to
break ground for that plant. And during the year we were introducing that
product line, trying to get it into manufacturing, we built that plant //. And I
stayed there a couple years and I ... did you know Jake //?

F: No.

C: He got promoted to run the computer systems group, which is what Office
of Product Operations is. // That's the engineering side of OPO. Or was.
Because all the computers, all the main frame // except the engineering part
and that was when they split manufacturing and engineering at headquarters level. He asked me to come be your director of programming in Detroit, so I finally succumbed. It was in '70, I think it was in '82? Roughly. '82. And I worked for him as director of programming and he left the company. The day he left town, I left town. I got ...

F: Back to the West Coast?
C: No, I, actually I went to work in SED as director of engineering // which was in Devon. I had never moved to Detroit, actually, and that was the deal I made him when he enticed me there. I went there on temporary living expense deals with a two-year maximum duration. That was the deal that I could go for a couple years and then have to go find me another job because I had no real desire to move to Detroit. So I ... I don't know if you knew Ann Edmunds or not in Detroit. I roomed with her and commuted on the weekends back to New Jersey where I still maintained my home. And when Jake quit the company. I was about probably 18 months into my two years //. And I had been looking around anyway for where I was going to land. So I got this job back //, barely within commuting distance of my house in Jersey at SED. // and I enjoyed that. In fact, I was sitting there fat, dumb and happy thinking I had it made, enjoyed the people, had good people working for me, enjoying the work of learning a lot, because I was really getting to see the customers, the big customers, complex systems--that's what we did. I don't know how much you've looked into SED, what they did, but that was good. I had really lost touch really with the customer. Back in the early days we used to go out on sales calls to tell customers about the 6500 design before it was even on paper. And that was enjoyable.
Anyway, I was sitting there fat, dumb and happy in November, actually I guess it was still in September or October of '85. They rattled my cage again and said ... I had seen that there was problems out here ... when you make the Wall Street Journal, you know that you've got serious problems. There were some serious problems here. And they came after me. As they say, the rest is history. I came out here in November of '85 to do the job I've got now. We've had a good year. As a matter of fact, I tell the customers, in March, after I'd been here for three months, this is the schedule that I presented to Caswell after I got here and we assessed where we are, and this is the benchmark for what we said we were going to do. The red mark is when we did them.

F: So they're all early.

C: Yes. And we're trying to make ... well this one's on schedule. But we're trying ... this red mark doesn't point to a day, but it's going to be earlier than that. And so I think we've restored our credibility with the customer base. And even with the management, because now I'm working on the task force which // Stern and Blumenthal have said o.k., we realize we're not spending enough money on the B-Series. What would it take to do it right? And I'll tell them.

F: That's exactly what you want.

C: Right. So we've had a good year. And we're going to make the schedules or beat them.

F: You've certainly done a lot of different stuff, and worked on different products.

C: Well, I tell people I haven't had ... I've been with the company 25 years. I haven't had one year of experience 25 times, I'm happy to tell you that. I've been allowed to do all kinds of things.
F: Is there any one project you think was more satisfying or more that you really feel you contributed to most?

C: Did you pick up on that, the design. There never will be a period in my life like being in on the design of the 6500. I have to say a sense of accomplishment, certainly the ALGOL compiler. When you do something that you know is ... there are very few times in your life when you get a chance to participate in developing a product when you have absolutely no doubts that it's the best, and you ... we had goals, we had performance goals for that compiler and technology goals, but there's things that just weren't done. We did a one-pass compiler that was unheard-of. In fact, when they recruited me for the job, I said you can't do that. So the learning curve, the amount of new things learned was absolutely dramatic. As a matter of fact, I remember it was extremely difficult. I had to get to a breaking point one day when I realized that I could not think of things in the terms that I had been used to thinking of, and I had to throw away all my old ideas and completely erase my blackboard and look at it from a different perspective because this machine, B5000 was different, so the sense of accomplishment, when I knew what we did was the best, was very good with the ALGOL compiler, // the same way. I knew that was right //. Sheer enjoyment of ... I mean those were also very intense, hard-working, I mean night and day, really tough concentrated, long-term work ethic. The 6500 was hard work in a different way, like getting headaches trying to think of a lot of things // but it was so enjoyable because of the persons, you can't get away from the people involved, that you work with. Barton, whom you know, I used to say everything I know about machines I learned from Barton. And I had the opportunity to be around him, see him quite a lot. It
was a very pleasant experience designing something that we knew was right, and being able to design it right.

F: It seemed like years ago you had more sense of working on that, a bigger piece of the project. I don't know if that's necessarily true, but it seems like //.

C: Well, you get these kind of jobs, Luke, my boss Luke Wills is my boss now. We were in his office in Blue Bell a couple weeks ago, looking out the window. They're building, there's a track of housing going up behind the Sperry, the Blue Bell complex over there, and there are these graders out there making roads and stuff. He says, "You know what I'd like? I'd love to have one of those. You know what? If you had a job like that you'd go home and tell the wife, What did you do today? Well, I made, I cut this road ... You can see what you did. And nowadays you got much ... well, for one thing, there are a lot of things going on, and there's projects, the major items span years. The B500 that we're going to deliver has spanned over three years. Of course, I've only been here for the last ... I will be here for the last two years of it. But still, two years is a long time to see the results. // The way you see the results, we've got them up and running now. It's not over 'til it's over. It's not done until it's shipped. In fact, I tell my people their job is not done until they have a satisfied customer on the other end of the line. So, I don't know there's the force of that concreteness to these kinds of jobs.

I got quite a bit of satisfaction out of getting that Flemington plant up and running. When we moved, we moved the total operation, 800 people, factories. We moved the engineers on one weekend. We moved the factory on another weekend. There was not a stitch dropped, and there was no confusion on Monday when people went to work. They knew where
they were and they knew everything. It was a very // if I do say so myself. It was very well done, really. We got a nice factory, in fact that's one of the more modern ... we had a vision even then, although at that time we just basically moved what we had and we had to, to do it fast. But we had a vision of making that a modern automated factory, and those things are coming through just now.

F: It seems as though you've been appointed troubleshooter.
C: Well, trouble-maker, maybe?
F: No, I don't know about that. //
C: It's been fun.
F: Think the people you work with make a big difference, too?
C: They make all the difference. // They ... I don't know how many other people go through life thinking they're in the wrong ... maybe they're in the wrong business. See, I was going to be a teacher when I was through. And for whatever crazy sort of circumstances, I ended in this business.
F: What school did you go to?
C: I went to East ... in fact, I started at East Texas State Teacher's College because I was going to be a teacher. When I ... I took a couple of education courses and changed my mind real quick. Majored in math, went to the University of Oklahoma to study math in graduate school. Spent two years there and I was a teaching graduate assistant there, but I got hungry and I accidentally ended up in programming // job, and I was going to write my thesis after I went to work. But I can remember learning programming in this place where I went to work, when they didn't have programming in schools. And I used to go around, after I was working on a real programming application and I'd go around with this guy and I would say, "They pay you for this sort of stuff? I never had so much fun in my life."
And I tell you the truth, secretly I still ... if I'm not careful, if I get started on
that // today, the day just vanishes. I can get hooked on that very quickly.
A few years back when the Commodor Vic-20 came out, I got my son one
of those for Christmas and I was at his house, so naturally we had to figure
how to use it, right? Christmas Day. So I'm sitting down there, he's sitting
looking over my shoulder and I'm working on the keyboard. And the next
thing you know, I looked at my watch and it's 4:30 in the morning. I said,
my God, I didn't even know any time had passed. I remember saying,
something is wrong. I've been in this business 25 years and I still, and this
is fun. Why is that? I really honestly believe that I could go back to
programming and enjoy the heck out of it.

F: You like what you're good at, I guess.

END OF SIDE TWO

END OF INTERVIEW