Oral History of Francis O. (Fran) Underwood

Interviewed by:
Robert Garner

Recorded: November 9, 2009
Mountain View, California

CHM Reference number: X5549.2010

Robert Garner: Today is November 9th, 2009. I'm Robert Garner here at the Volunteer Computer History Museum. IBM [International Business Machines, Inc.] 1401 Project Lead, and I'm here to proudly interview Francis O. Underwood, who is the Chief, and only, Architect of the IBM 1401. At that time he knew it as the space machine, but we'll call it the 1401 now. Fran, I'm going to call you Fran. Everyone calls you Fran.

Francis Underwood: That's right.

Garner: To be honest, at first I didn't know whether it was a woman who had been the architect of the 1401. But everyone calls you Fran. And Fran, what I'd like to do is go through your amazing early history, starting from your—well, since you were a little boy. And I think that starts in Omaha, Nebraska. Do you want to start from there?

Underwood: I can go back farther. I come from a family of technically oriented people. My grandfather was a watchmaker and an engraver. My father was a architect, an engineer, and a teacher. Unfortunately, I lost my father when I was four. So I feel that my peculiar talents were the result of good genes. I was born in Omaha, Nebraska, and four years later, my father got appendicitis, and gangrene set in. There was no way to save him at that time, so he passed away. My mother and I went back to Binghamton, New York, to live with my grandparents. So I was brought up pretty much by my grandparents. My mother went back to school for further training as a nurse. So my grandparents raised me, and that was a pretty good thing. Great people.

Garner: May I ask when were you born?

Underwood: I was born in 1926. I'm pretty ancient right now. So being in Binghamton was very fortunate, because they had a technical high school, so that the last—in ten and eleven and twelfth grades, I attended the technical-mechanical course there, which was very, very strong in machine shop, drafting, mathematics and physics. The idea was that we were going to be trained as toolmakers, and eventually end up at IBM as toolmakers. Well, I completed the three-year course, and then the war came along, so I had to enlist in the Navy, so I never got to go to IBM as a toolmaker.

Garner: Was it a very prestigious institution, the technical school? Something about only 20 people were accepted at a time?

Underwood: Oh, yes, it was a normal high school, but they had this particular special course set aside as technical-mechanical. 2,000 people applied for this course in the city every year, and they only accepted 20, and they only graduated ten out of this course. It was pretty tough.

Garner: So you were one of those ten.

Underwood: Oh, yes!

Garner: What made it tough?

Underwood: Well, it's very intense in mathematics and physics. We had three-hours of machine shop every day, three hours of drafting every week, and physics and math every day. We just had a lot of technical stuff there.

Garner: Do you have any fond memories, or is it just...

Underwood: Oh, it was great! If I could remember it all, I'd love to. But I forget a lot.

Garner: Where were you when the war ended, do you remember?

Underwood: Yes, I went to—I joined the Navy V5 Program; the intent of that program was to train pilots. First they were going to send us to one year of college, and then off to flight school. Well, I got my first year of college in at Hobart College in Geneva, New York. Then the Navy discovered that they were not losing pilots at the rate they had anticipated. They didn't need any more at that particular time, so I took another screening test, and managed to stay in college. They sent me to Cornell University in Ithaca. And that was fine. I majored in mechanical engineering and minored in aeronautical engineering.

Garner: At Cornell?

Underwood: At Cornell. Then the war ended, and I was given a choice, "Either get out, or stay in. And if you stay in, you're going to go to sea for four years." And I said, "No, thank you. I prefer to leave without a degree."

Garner: So to stay in meant to get your degree. If you wanted to get your degree.

Underwood: If I wanted to get the degree, I'd have to stay in. I said, "No, thanks." So I went back to Binghamton, and got a job as a tool engineer with a small engineering company. We did a lot of work for IBM, and I, in particular, did tool design for IBM at the time that the 407 was just starting production. My job was to do all the tooling for the 407.

Garner: These are things like a drill jig?

Underwood: I did drill jigs and punches, and assembly fixtures and all kinds of stuff. But it didn't pay well. So I went directly to IBM to get a job as a tool maker.

Garner: In fact, how much did it pay? Do you remember how much you were paid?

Underwood: Seventy-five cents an hour! And it was pretty poor pay. I could do better at IBM. So I went down there and applied for a job as a tool maker.

Garner: Do you remember what year that was?

Underwood: Oh, that must have been ’47, somewhere around in there. Yes. Or early ’48. It just so happened that at the time I applied, IBM had decided they would like to try something new in training customer engineers. They wanted ten engineering-type people of diverse backgrounds to enter a new trial program. We would go into the factory, instead of the classroom, and we would actually build every product. We would learn how to build every product in the line, five examples of each product. And not only build them, but do the final tests.

Garner: Do you remember what some of those products were?

Underwood: Oh, it was everything in the line. Like the key punches, and the verifiers, and the interpreters. Particularly the counting machines like the 402 and the 405.

Garner: And those are very complex machines. How many parts were in a 405, do you think? 50,000?

Underwood: Thousands. Oh, yes.

Garner: 100,000 maybe.

Underwood: Oh, maybe. It was terribly, purely mechanical. We had about, oh, close to a thousand relays, and...

Garner: Complex printing mechanism.

Underwood: Oh, and the printing mechanism consisted of type bars. It moved up and down. And every little piece of type in every bar, there would be like 48 type...

Garner: Type.

Underwood: Yes, with a little spring behind them to return them, and then a hammer that would put the type plug against the ribbon and that against the paper. Then they would all collapse and line up, and then they would rise up again, and be stopped under control of a punch card and then the hammers would fire and print the line. It was very, very complicated.

Garner: One of the remarkable things about these machines is they had to be milled very accurately, to microns, almost to work properly.

Underwood: Oh, yes.

Garner: You had to hand-assemble these machines as well. Did they ever not work after you assembled them?

Underwood: Oh, no! <*laughs*> No, no, no. It was easy technology. It was so...

Garner: You make it sound easy.

Underwood: Well, it was. You had big parts, so you could get a hold of them, you could measure them, you could look at them, and evaluate them very nicely. It was easy.

Garner: What about the wiring diagrams?

Underwood: Well, like the 405, the wiring diagram was 40 pages long, all fan-folded together and very, very complex, but—and you know when I started this course, I didn't know anything about electricity. Maybe Ohm's Law, but that was about all. And here I had to learn all this stuff. I didn't know anything about accounting or industry in general. I was very naïve. I had to learn all this stuff. Well, I learned it, but I didn't have it all together. I didn't have the big picture until about the last day on the last machine with the last final test, the guy I was taking instruction from made some off-the-cuff remark, didn't amount to anything, but all of a sudden, everything that I learned just came together in my mind, and I understood everything. It was the most amazing moment of my life—to have that happen!

Garner: Did you have to get together on Saturdays?

Underwood: <*laughs*> Yes, these ten guys that were going through this training program were all separated. We were all working in different departments, but we would get together every week or so with the regular customer engineering classes. The first thing we did was sing songs. "Hail to Thomas G. Watson, leader of the IBM." And "Onward Ever IBM," and all that stuff.

Garner: With your white shirts and black ties.

Underwood: Oh, yes, we all had to dress up in dark suits and ties and white shirts.

Garner: Any fear about long ties around all this machinery, or you would stick the tie in your shirt or something?

Underwood: I don't recall. I should remember, but I don't.

Garner: So then one day an electronic machine came along, the 604?

Underwood: Well, yes, when one of these courses, or classes, got together, IBM had developed a new calculator called the 604. It had about 2,000 vacuum tubes, and I'd never seen one before. They wheeled that in, and demonstrated it. It would add, subtract, multiply and divide at very, very high speed. I was overwhelmed. So I said, "Pardon me, but where are you ever going to find people that can service this incredible machine?" And they said, "Oh, you guys will do it!" Well, I said, "Well, maybe." But I couldn't see it. Well, I finally graduated from that course, and they sent me to Washington, D.C. My first customer was the Veteran's Administration. They had big, long two-story buildings on either side of the reflection pool between the Jefferson [Lincoln] Memorial and the Washington Monument. Every building had 125 keypunches, 125 verifiers, and down at the end there were a few 402s and interpreters and stuff like that, collators. So I learned an awful lot about key punches. [I] got very proficient. I did that for maybe a year.

Garner: How much were they paying you?

Underwood: Oh, maybe a dollar-and-a-quarter an hour. Something like that. But I was getting along very well.

Garner: What were you doing in your spare time?

Underwood: I always wanted to be an engineer, not a customer engineer. Although, I enjoyed the work somewhat. So I had a couple of ideas about two new machines. So I built myself a drawing board, and evenings, I would design these two machines I had in mind. The first one was, I don't know why they never built such a thing, but it was a calculating keypunch. So that you'd key in the factors, and right away, the answer would be punched into the same card, instead of having to take the card, run it through other machinery to get the answer punched. I thought it was a good idea so I designed that.

Garner: So it could do add, subtract, multiply...

Underwood: Oh, yes, add, subtract, multiply and divide. I did all the certain design.

Garner: All relay-based?

Underwood: Yes, all relay-based.

Garner: Wow, that's hard.

Underwood: The other machine was an interpreter—the machine that reads the card, and then prints the information on the same card. At the time all we had was a machine called a 511. It was a clunky old machine, and dirty. Ugh! So I designed a new machine that used wheels instead of type bars. It could print anywhere on the card, and it was very fast. So I designed that. I took these designs to my manager, my customer engineering manager. He says, “No,” he wasn’t interested. And I said, “Well, I’d like to present these to engineering to see if I can get a job in the engineering.” He says, “No, they won’t be interested,” he says, “No way.” Turned out the week after that, IBM was looking for engineers. They sent a man out from the lab—several people out from the lab to the outlying offices and I got to be interviewed, and I took my drawings along to show this guy. One week later, my CE [Customer Engineering] manager had to come to me and say, “Well, they want to interview you in Endicott.” So I went up to Endicott with all my stuff. A couple days later, after I got back, he had to come to me again and say, “They made you an offer.” He wouldn’t tell me how much. He was embarrassed. So...

Garner: Undoubtedly, more than him, potentially. Undoubtedly very close to his salary.

Underwood: Oh, more. So of course I took the job, and it was great.

Garner: Do you remember what year you took that job with them?

Underwood: Well, it must have been...

Garner: 1951?

Underwood: ’51? Yes, I think it was about 1951. I think so.

Garner: And your first assignment?

Underwood: My first assignment was to redesign the 402 accounting machine so they could read the new 160-column card that had been proposed. I did all that mechanical design.

Garner: Basically twice the density of the existing cards.

Underwood: Exactly, they’d put two columns of information in each card column by using binary coding. All the rest of the equipment like keypunches and verifiers and so on, were being designed in Poughkeepsie. Well, that whole program fell through, like programs do.

Garner: Who were you working for?

Underwood: I was working for a bright guy named Ben Durfee. Ben Durfee had been a customer engineer with his offices in Russia.

Garner: Of all places!

Underwood: He spent years there in Russia servicing IBM equipment. When he left Russia and came back to the United States, he designed—well, relay circuits for computing. Large electro-mechanical relays circuits for add, subtract, multiply and divide. Same thing I had done, but he was good. He was good! But everything he did was seat of the pants. Ben Durfee, he would draw circuit diagrams on Vellum in ink, page after page after page of these things with these thousands of relay points, and every relay point had an open-sided, close-sided transfer point, and each one of those had to be drawn with a little circle. And he did that. Oh, my god! Tedious work, but he was good at it. I mention this, because I have a reason. While I was working for him...

Garner: Took one of these first—<*inaudible*>

Underwood: The Director of Engineering Education came to me and asked me to teach in addition to my regular job. Well, I'd never done any teaching. I'd never been able to talk in front of people—totally foreign to me. So I said, "No, I can't do that."

Garner: What did he want you to teach?

Underwood: He wanted me to teach the engineering training program.

Garner: You'd just become an engineer, and they wanted you to teach.

Underwood: I'd just become one, yes. And he wanted me to teach. Remember now, I had two years of field work behind me, and I was good at it. I'll admit, I was good at it. These 40 pages of circuit diagrams for the 402, I remembered it! I could service a machine without reference to it usually.

Garner: And I think you also took a class in Boolean Algebra?

Underwood: Yes, that's where I'm getting to.

Garner: Okay.

Underwood: So Perry Perrone, he...

Garner: Who was he? Perry Perrone?

Underwood: Perry Perrone, he was the Director of Engineering Education. He finally persuaded me to teach this course in engineering training. So I finally buckled under and said, "Okay, I'll try it." Well, it turned out, much to my amazement, that it was easy. I had no problem talking to these guys. They were my kind of people, and I knew more than they did, so I had no trouble talking to them. The engineering program students were engineers from other walks of life, and I had to teach them how to be an IBM engineer. Which is kind of different.

Garner: You had to teach them everything about IBM equipment basically.

Underwood: Yes, I had to teach them all about the equipment, and how we did things. You know, the concepts of IBM engineering. Well, so that was going along fine. Then there was a young man who had just graduated from MIT [Massachusetts Institute of Technology], and his name was Fred Fosse. He came to the Endicott lab and taught a course in "Switching Circuit—Mathematical Approach to Logic Design." I was fortunate enough to sit in that course. Well, I was intrigued. So I learned how to mathematically approach a switching circuit problem. This is what I had while I was working with Ben Durfee. I came to him, and I showed him these mathematical approaches to what he had been doing by intuition. He was totally amazed. He never thought there could be anything like that.

Garner: My understanding that Fred was—who was Fred's teacher?

Underwood: Well, Fred, I believe, learned what he knew from Claude Shannon, who was the father of information theory.

Garner: I understand Claude actually wrote his master's thesis on switching circuits.

Underwood: Yes.

Garner: So this was the first student to really learn that, and then you were the second person in line.

Underwood: That's right. I just took to the course, and so when IBM wanted to have a second session of the course, I volunteered to teach it! So that went well. Very well.

Garner: You were moving very fast.

Underwood: Apparently.

Garner: You taught it, so you knew it well enough just to go ahead and start teaching it.

Underwood: Oh, yes!

Garner: And you had a student in your class.

Underwood: I had a great student—well, they were all great. But this particular student that I'm going to talk about, his name was Mitch [Mitchell] Marcus. He took to the material, too, and the way things worked out, I essentially finished teaching the course and then turned it over to him. He continued, and he wrote a book on switching circuits that became the standard throughout the world on switching circuit design.

Garner: So your enthusiasm inspired him to write a book to become a teacher?

Underwood: Well, he was the kind of guy who would do it anyway. I didn't have to really inspire him.

Garner: You connected. You still know him today.

Underwood: Oh, yes! He's been my best friend ever since then.

Garner: You guys had a joint bond.

Underwood: Yes, oh yes.

Garner: And it was fun. The switching circuit stuff was fun to you?

Underwood: Oh! Yes! Great fun!

Garner: Because it was like math puzzles almost.

Underwood: Yes, it was.

Garner: Now there was no computer to do the switching circuits automatically. You had to go through all the things by hand to optimize a switching circuit. That was part of what you were teaching.

Underwood: Yes. So it was great. It applied to the level of technology that we enjoyed at that time. Relay-switching circuits. So.

Garner: One thing that I think a lot of people today don't appreciate is that in a relay circuit, the current can go either direction. It can either go left to right, or right to left. So you have to be very careful how you design your circuits, I understand.

Underwood: Well, of course. If you're careless, you're going to make an error. They have short circuits and open circuits and all kinds of problems. That's what the mathematical approach is supposed to solve. So after I was teaching all that stuff, I had a regular job still. IBM designed a new interpreter, not my design, but another design, called the 519, and I did some design work on that product. So also during that time, I taught a computer systems, or computer architecture course that I had to devise myself. I don't know where I got all this information from but I did an awful lot of self-education. I had to learn all about electronics on my own, and oh, as a matter of fact, in one of the engineering training programs I had to teach the 604, and it was duck soup. I'd read the material at night, and teach it the next day. It was nothing to it. It was easy! So I devised a computer architecture course and taught that. Eventually, I was assigned to the Advanced Systems Development Department.

Garner: In what year?

Underwood: Oh, let's see '55, '56?

Garner: Did you want to join?

Underwood: Oh, hey! It was a step up, you know?

Garner: Oh, it was a step up. You wanted to be there.

Underwood: Oh, sure!

Garner: The Advanced Research Group basically. New ideas. New frontiers.

Underwood: Exactly. Our assignment was essentially to anticipate or well...

Garner: Invent the future?

Underwood: Invent the—exactly! That's a good term! Invent the future. We were to try to look ahead ten years, and anticipate what was coming along in the way of technology. What the user—and we were fed information from another group, a marketing or a product planning group had to anticipate customer requirements that were going to show up ten years from now. So we had all kinds of opportunities to invent all day long. All we did was invent, and innovate and create. We had just a lot of fun!

Garner: Do you remember some of the projects?

Underwood: Well, I remember that there was one project called the transistorized accounting machine, but I don't know anything more than that. I know we worked on it. My specialty in this department was to lay out the flow diagrams, and develop the logic. And integrate the storage and the I/O [input/output] mechanisms. And just architect the whole system.

Garner: Was there a project about video? Video document reader or something?

Underwood: There was another one called the Vidor, V-I-D-O-R, that was a video document reader. And of course, that was 50 or 60 years ago. I don't remember what that was.

Garner: Did it work?

Underwood: Oh, we didn't build anything.

Garner: I forgot. It was just all on paper.

Underwood: It worked great on paper. They always do.

Garner: And it probably was plug-board control even. It probably wasn't-- do you remember if it was a stored program?

Underwood: No, I don't remember.

Garner: It probably wasn't then.

Underwood: Let me stop just a minute and say—I want to back up a minute and say a word about technology. There were two species of technology at this time. One was the accounting machine species, started back in 1890 with the 1890 Census. Herman Hollerith devised a scheme to capture data in punched cards, and he also developed the machinery to process these punch cards, to sort them, and to collate them, and to accumulate totals and print the results. That technology slightly improved through the years, stood for 65 years, with no real innovation. It just was the same old thing, only a little better, slight improvements over time. The other species started with World War II. There was a need to plot range tables for cannon, and the very first attempts to do this sort of thing were implemented by analog computers. That's where they had little gears and wheels all wound up, and resisters, and potentiometers and so on, and it would draw curves. That was the output. Not very good. Very hard to setup. Not programmable, really. Then the next requirement during the war was decryption. Encoding and decoding, particularly to decode enemy transmissions. That required digital means. At that time, this new species of computing machinery started to develop. Allen Turing being one of the first. Then the implementation was by very crude electronic means. That improved quite rapidly. In 1955, 1960, you had these two species side-by-side. One, the old electromechanical data processing equipment, implemented with relays, and machinery, and plug boards. That was the way things were done! They always had been, they always will be. The other species was developing, but they were single instance machines. That is, these machines that they envisioned for solving, oh, differential equations, and decoding and so on, were large, large machines. They'd fill a room, and then they'd have thousands and thousands of vacuum tubes. They were highly unreliable, very expensive, but for the few minutes you could get one to run, you could do an awful lot of computation. But they, too, were based on plug-boards, and that's the way that was done. There was a big gap between the two species, and yet there was a similarity. They both processed data. So on the other side, IBM got into this business to develop a couple large machines, the Mark I, and so on. And the 701, 702, kinds of machines.

END OF TAPE 1

START OF TAPE 2

Garner: And then when you're done with that story, I'll ask you about your experience with Von Neumann in ASDD—.

Underwood: Oh.

Garner: Yes. And then any—

Underwood: Not time yet.

Garner: No, when you're done. When you're done with this story, then I'll ask you. Was there any other mach—

<*audio cuts out*>

Garner: Is my volume okay on my voice? Okay. It'd been down low. So start with that line.

Underwood: In the middle fifties, IBM experienced some competition in the accounting machine species of equipment, and from Bull Gamma in Europe had come out with a pretty powerful machine. IBM had no plan, so they devised one, and—

Garner: Can we stop for a second?

Underwood: There was a contention between the two species at the juncture of the proposal of WWAM [World Wide Accounting Machine]. WWAM was picking up certain aspects of the computer line and certain aspects of the accounting machine line and trying to merge them.

Garner: When you were in ASDD?

Underwood: Yes.

Garner: Okay, so why don't you just do that? Finish that thought, and then just end it there, then talk about the rest of your work in ASDD.

Underwood: All right.

Garner: The patterns, and do the Von Neumann story and then how you moved out of ASDD, and then we'll get into the whole SPACE story.

Underwood: Okay. We had the two species; the electronic computers, big, expensive, very fast, and the accounting machine line, which was slow but not so expensive. There came a time when certain aspects of the computer line and certain aspects of the accounting machine line were trying to be merged into some sort of a beast that would make a good product for IBM. It turned out that the instigators of this thing were the French and the Germans, who had proposed a worldwide accounting machine that looked like a unit record machine, but was built with electronic components from the electronic species. Well, I want to finish the discussion about my participation in the Advanced Systems Development Division. As I said, my particular contribution was to do the architecture of these advanced systems, to draw the flow diagrams and work out the logic, and just invent and innovate to my heart's content. It was great fun. During that time—Oh! I worked on a sorting machine, a tape sorting machine. Why would I work on that? Well, the department just happened to have two very good mathematicians. Their names were Nelson and Armstrong. These guys taught me a lot of mathematics. They were just great. But they had devised a sorting scheme, a way to fast sort tape data. It had to be on tape and they had these two tapes; they would just merge them and sort the stuff very rapidly. They had a mathematical approach. They had no idea how you could implement such a thing, but they had this mathematical proof that it was a much better way to sort. So I got the job working out a way to implement such a thing and it was called a 'tape sorter' <*laughs*>. It was totally worthless <*laughs*> but I got a patent for it.

Garner: So it was all controlled by transistors or relays? Do you recall?

Underwood: It was all transistorized.

Garner: Transistorized?

Underwood: Yeah. No control panels <*laughs*>.

Garner: No control panel, but no control store either?

Underwood: No. Well—

Garner: No program? There was no stored program?

Underwood: No. Well, there was a store.

Garner: There was a store.

Underwood: You had a record or two into the storage, a record or two from the other tape and do some manipulation with the data and write it back out, sorted.

Garner: You put in each step of the algorithm in transistors, in hard wire transistors?

Underwood: Yes, I did that. Well, it was terribly expensive and it was interesting. It got me a patent <*laughs*>. But during this process, there was some mathematical aspect that hadn't been solved. Fortunately, IBM had a contract with John von Neumann and he would come to us at various times and we would be able to meet with him and discuss certain aspects of this and that. This one day, von Neumann showed up and we made an appointment to see him and discuss this sorting technique. So we asked him something about it and he listened and he went, "Ahh," and then he'd pop back up and he'd say, "The way you do this is the following." <*laughs*> It was totally [insane?] to everybody else, but he figured it out in a few seconds.

Garner: So he'd go quiet first and then all of a sudden out would come, like, an oracle that would come the answer? Wow. Did he talk a lot or was he generally a quiet man?

Underwood: No, he was quiet. He was quiet. He wasn't verbose at all.

Garner: Because he had a reputation for being sometimes very opinionated, I guess, would be—

Underwood: Well, that may be, but when he was at IBM in these meetings, he just listened and gave his opinion <*laughs*>.

Garner: Did he ever talk about what he was working on at IBM? No?

Underwood: No.

Garner: No. It was very secretive back then. I think people did not tend to share a lot of what they were working on; was that generally true? People tended not to share what they were doing? I've heard that.

Underwood: I can't recall. So here I am at ASDD and I've designed all these myriad systems and I began to hear about the WWAM [World Wide Accounting Machine] machine. Somehow I got involved in it; I don't remember how, but I know that IBM needed such a machine because the competition was getting pretty bad in this area. Ralph Mork, who was the director of Military Division in IBM, was assigned the task of leading a new development program for an advanced accounting machine. So he needed to staff up, so the first thing he needed was an architect, so he got a hold of me and I had some ideas and I said, "Okay, I'll take it on." I worked for Ralph for some time and during that time, I spent a lot of time looking at various other portfolios, ways of doing things. Ralph Mork knew an engineer in Military Division whose name was Russ Rowley, and Russ was developing super secret machines for the Navy, but there was one aspect he could talk about, and that was core memory. So I learned about core memory from Russ.

Garner: Was Russ the Manager or a circuit guy, or maybe both, because he understood—

Underwood: The Manager.

Garner: He was a Manager. But he also knew circuits, did he?

Underwood: Oh, I guess he did. Sure. Well, yes. Oh, yes.

Garner: So he'd been managing this large computer design for some government agency?

Underwood: Well, again, that's the military and I wasn't supposed to know. *<laughs>*

Garner: Didn't know what he was doing, yeah. But it sounded like it could've been a stored program machine if it had core memory.

Underwood: Well, it sounds like it, doesn't it?

Garner: *<laughs>*

Underwood: So anyway, I looked at what had developed so far in the WWAM machine; looked at their printer, looked at their electronics, their data flow, their storage, their I/O. I pretty much concentrated my study on that machine by looking at the circuitry, and one day, I was overwhelmed by a discovery I had made. I was looking and counting components in various parts of the machine—how much did it cost to add and subtract and to multiply, and to move data, and store—and I came across a big chunk of electronics that had nothing to do with logic. It all was connected with driving the plug board. Driving the plugboard with electrical or mechanical relays is no problem. You can mis-wire to your heart's content; you won't hurt anything. But with an electronically-driven plugboard—

Garner: A transistor driving—

Underwood: Yes, transistor-driven plugboard, you couldn't make a mistake.

Garner: Because you might blow out a transistor.

Underwood: Oh, yes. So they had a safeguard against any mis-wiring, and that took an awful lot of electronics.

Garner: Interesting.

Underwood: And I was just astounded. If you get rid of the plugboard, you get rid of all these electronics.

Garner: It's like the old world trying to meet the new world and they weren't compatible.

Underwood: That's correct.

Garner: The transistors were low voltage things and easy to kill and a plugboard could hook two outputs together, for instance.

Underwood: Yes. And simultaneously, I realized that if you got rid of the control panel, you were left with a pure stored-programming machine, stored-program machine, that you could completely specify the process or the problem to be solved—

Garner: The control in the software.

Underwood: —without being limited by the control panel. The control panel was not only expensive to implement, but the poor customer had to buy the expensive plugboards and plug them up for every single task that they had to do—huge amount of money.

Garner: It's like razors, right? Razors and razor blades; the poor customers needed razors and handles. Extra handles.

Underwood: Yes, or printers and print cartridges. But it was terrible, just awful. And the WWAM control panel was, in my mind, very difficult to understand. So I realized that—

Garner: It was based on variable length data fields, somehow a beginning of field and end of field and some kind of weird thing.

Underwood: That's right. They did eliminate a lot of wires by this concept of plugging only the start or the end of the field. You didn't have to plug all the other wires but still, it was difficult. But you couldn't specify how to do it by programming. I mean, you could run a program to solve a job, but this would be a completely separate task. It couldn't be automated and it was still error-prone and expensive. If you got rid of all that, you could completely specify how to solve the job.

Garner: But could you achieve the cost? How did you know whether it would be cheap enough?

Underwood: Well, I said 45 per cent thrown out, it's got to be cheaper <*laughs*>. I started with that. You know, I just knew that that was the right thing to do.

Garner: Yes, the amount of core memory, I understand, in the WWAM could be expanded up to 1,900 characters, just data, not for instructions, so that's actually pretty big, 1,900 characters in that original machine. Just they needed to put instructions in there <*laughs*>.

Underwood: Oh. Well—

Garner: What did you do first? Did you tell Ralph? Do you remember what—

Underwood: Well, I very seldom saw Ralph Mork. I think he was protecting me, and bless him for that, because I know a lot of times, getting into a project there's always somebody there saying, “Don’t do that, do this, don’t—” you know, micro-managing. But Ralph kept these people away from me, and he would feed, quote, “information”, to these people. I got some feedback one day about the brand new electronic counters that I had invented. Well, I hadn’t invented any new counters, but Ralph told these people that things were coming along great <*laughs*>. I lost my train of thought here.

Garner: Somehow, you decided to write the first SPACE memo.

Underwood: Oh, well—

Garner: It wasn’t called that initially.

Underwood: No. I started to write some descriptions of what I thought this new accounting machine ought to be, and I called it by a very long acronym; the name of the machine was Stored Program Magnetic Core Memory Variable Instruction Length Variable Word Length Accounting and Calculating Machine <*laughs*>.

Garner: The longest acronym I've known <*laughs*>.

Underwood: Yes. So that was SPMCMVWLVILCAM <*laughs*>. I did that just to be facetious. We had a lot of fun in those days and I could get away with that. But then I began to get serious and it was just about that time, 1957, when Sputnik went up. The space race was the rage of the day, so I managed to come up with the acronym SPACE, which was the Stored Program Accounting and—

Garner: Accounting and Calculating Program, right?

Underwood: The Stored Program Accounting and Calculating Equipment; SPACE. The SPACE machine. That caught people’s imagination, I guess. People started getting interested. The lab managers—

Garner: That’s a touch of marketing there, a touch of good marketing <*laughs*>.

Underwood: Well, yes. The marketing people liked it and the lab managers and various other engineers became aware of it.

Garner: How did you go about designing the instruction set? How did you do that?

Underwood: Well, that's another thing <*laughs*>. Remember I mentioned two species of machines; the electronic calculators; they were binary-oriented and because they were binary, you could specify operands in many ways by some binary number. But the accounting machine species, they were all alphanumeric. Alphanumeric. They wanted to present a different face to the user. He wanted to see words, letters, numbers, decimal numbers.

Garner: A human-friendly interface. Friendly to humans.

Underwood: Yes, a human-friendly interface. That's what he, in my opinion, wanted to see. He didn't know anything about binary; he couldn't care less about binary.

Garner: Didn't know about compilers.

Underwood: No, didn't know about compilers. So I wanted to put a face on this thing that would appeal to the new user. He would see this friendly face beaming at him saying, "A for add, S for subtract, M for multiply, B for branch."

Garner: So he wouldn't be intimidated, so maybe he would feel comfortable writing the actual instructions and seeing them come out on paper.

Underwood: Exactly, be comfortable and find it friendly and easy to do.

Garner: You didn't have to do that. You could've made it hard and complex, but you knew that to be successful, you had to do this. You knew in your heart for this to be successful or—

Underwood: Well, that's the way I wanted it.

Garner: Yes. And you knew it would be right for the customer. You had all that field training from earlier on—

Underwood: Exactly.

Garner: —and you knew what that was like.

Underwood: Yes, that's another thing. Most of the engineers had never been in the field and it was only the field-trained people that could really produce good products. Like, Chuck Manscom [ph?]; he was the manager of the product, of the 141 product line eventually, and he had been a customer. He understood what I was trying to say; he embraced it <*laughs*> enthusiastically.

Garner: So who were the first people to come on board the project after Rowley? There was yourself, Rowley and there was a third person, wasn't there? Was that very early on, no? Was Jim Ingram—or did George come on early? Do you remember?

Underwood: Well, Jim was there. I don't remember. You know, today, I don't remember. But anyway, so here I am at ASDD and I'd conceived the SPACE machine and of course I didn't conceive of it in its final form; I had to go through some iterations. I looked at a three-address instruction and a two-address instruction but I ended up with a variable length instruction. I wrote memos covering all of these different aspects of the architecture that I was proposing; a series of four memos.

Garner: Did you hand-code algorithms to see which instruction and coding was best? Did somebody code something up to see during that process?

Underwood: Oh, how I learned all this stuff was working... Well, I coded it myself. I had a target application in mind and this application was the French banking problem, which the WWAM program had used as a target and—

Garner: Do you remember anything about the algorithm at all, how it worked or—

Underwood: Well, it's different than the way we bank here. It has something to do with the accumulation of interest and so on. I don't remember the details. But I would work out the flow diagram to solve that problem with a current version of SPACE, and it directed me very rapidly to what SPACE finally ended up to be. I was able to completely encode the solution of the French banking problem in 1,400 characters of storage. Totally <*laughs*> ridiculous today.

Garner: That's amazing.

Underwood: But one aspect of the SPACE machine was its particular data flow. Being a variable word length machine, the only really good way to attack this problem was a serial-by-character implementation; one character process at a time, rather than 16 or 32 or 64.

Garner: Like in the 650. I think the 650 was a parallel work machine.

Underwood: Probably. Most every machine was, except the accounting machines species. They were all single character machines implemented with electrical mechanical relays.

Garner: And the WWAM had continued that tradition of character serial in the WWAM, and you basically continued that on.

Underwood: That's correct, I did. Today, that would be a ridiculous thing to do, but back then with the statement from IBM management that from now on, all machines were going to be transistorized; no more vacuum tubes for any reason, all transistorized. Well, with transistors, as fast as they are, much faster than relays, you can afford to implement a serial-by-character implementation. It's slow, but really relatively very fast.

Garner: Compared to the hardware peripherals? Compared to the mechanical peripherals?

Underwood: Yes, that's right. It was magnitudes of speed, faster, very rapid. You could implement the logic for a single data path with very little logic. It just repeatedly uses it over and over and over again for as long as required. You start to process a word and you keep on going until something told you to stop, but what was that thing? Well, heretofore in the electronic computer species, that was a field mark or a record mark which actually occupied a position in memory. So the number of positions of memory devoted to this function was a function of the number of words that were defined in memory, which would be all over the place.

Garner: It was a special character that indicated the end of the variable length field?

Underwood: Yes.

Garner: Yeah, or maybe a special BCD [binary-coded decimal] encoding or something, yes.

Underwood: But in the data processing species, what we had to deal with was alphanumeric characters; alphabetic characters and decimal numbers with variable lengths. Predominantly, you had single column fields. You had lots of single column fields. You'd have a 10 column amount field or a 12 column amount field, or whatever the size of the National Debt is today, I don't know. It's huge. Or you had maybe 24 or 30 characters for a name, an address, but that was the range; from one to 30, predominantly very, very few characters in a field. Implementing the end of a field or the end of a record with a record mark consumed a lot of memory, because there were a lot of fields, very short. I didn't think that was very good.

Garner: Yes. You were saying that most of the data, even though the field on the punch card may have been wide, the data itself was pretty narrow, and so in memory—in other words, a punch card might have a very wide field, but the data in it was narrower typically. Most fields were short.

Underwood: Yes. Oh, yes.

Garner: And so when it went into memory on those electronic machines, you had to add an extra character for end of—and that took extra memory.

Underwood: Yes.

Garner: If the fields had all been long, that wouldn't be such a big deal, but because a lot of the data coming in was short, that took a lot of extra memory, and you didn't like that.

Underwood: True. And the fields were not defined in the card.

Garner: Right.

Underwood: There was no field marker, record marker, in the card, so how did you handle that? Well, I didn't know. So it so happened that there was another individual in the department and his name was Ed Grenchus and he was wrong on some sort of a processor which had some level of secrecy surrounding it. We didn't really know very much about his machine, but he did get a patent on a thing called a 'word mark'. This was an extra plane of cores that could be set or reset by certain means. They were activated by a plug-wire. So you had to have a plug-board to activate the word mark in his machine. Well, I didn't have any plug-board, so I had to devise another way to use word marks. Of course, by this time, I had convinced myself that word marks were the proper way to delineate fields in an accounting machine. I was sure of that. So I devised a new way to set and reset word marks and use them to control processes. So he has the patent and I get to use it; my version.

Garner: So basically, the word mark was an extra bit.

Underwood: Oh, yes. An extra—

Garner: In each character, there was an extra bit.

Underwood: Yes. So people would look at that and say, "Oh, one out of eight? That's 12 and a half per cent of the memory wasted." And I <*laughs*> said, "Show me a better way." There wasn't a better way. Subsequently, I found out about high-level programming, high-level languages where you have a data declaration, a section of the program and there you can define all the field characteristics and you refer to that as you're compiling for the length of the words; you don't need word marks, or delineators. So anyway, I didn't know about that at the time, so that was the only choice left to me, was word marks. I liked them. They were so—

Garner: Intuitive?

Underwood: They were so intuitive, yes. And I just—

Garner: You know Fran, one of the problems of not having a word mark, you say a compiler could know the width of everything and that had to be true for the 705 and the big computers, but the problem, though, is that the programmer has to keep track of how big everything is in his head, and he can make mistakes and accidentally take data and move it the wrong way, cutting it off. By having the word mark, you actually eliminated the possibility of mistakes <*laughs*>.

Underwood: Yeah, it was a pretty good machine.

Garner: *<laughs>*

Underwood: The thing that impressed me, when you came out to visit me, you brought along a printout of the powers of two. You print a one and on the next line you print two and the next line four and then eight and then 16, and you keep going until you've got a number 132 characters long and then you stop. It takes nine 1401 instructions to do this whole job. Today, it takes tens of thousands of instructions to do the same job, but of course they are executed at enormous speeds, that it compensates in a way. But it points out how simple the 1401 instruction code was. You could sit down and just write nine instructions and away you go. It was easy to do. You didn't have to compile or convert or anything of that sort. So the machine code was accessible to the user, and I figured that was important. Well, after I had pretty well defined the SPACE machine, as is the 1401 processor, or incidentally, the other thing I designed was the operator's console, it was a simple data flow on the 1401 and I thought it would be useful to display that data flow on the operator's console with the actual characters lit up in the various places for the—appeared data flow. So I designed the data flow operator's console; never saw it before.

Garner: It's a very beautiful design. It's very emblematic of 1401. Very few computers ever did that again. Very artistic. It's very artistic.

Underwood: Yes, well, that's the other thing. This machine was not designed by a scientist; it was designed by an artist. This design required a lot of art. You had to invent stuff and take it on faith it was going to be all right *<laughs>*. That's what I did; I put a whole lot of art into the design to make it appeal to the user.

Garner: Yes, I've seen many of your drawings and you have a very beautiful way of drawing diagrams and that shows up in the 1401 panel and your patterns, and probably all the hundreds of other drawings that did not survive the ages of time *<laughs>*.

END OF TAPE 2

START OF TAPE 3

Underwood: Well, I wanted to tell you about the—a little bit more about the instruction set. I mentioned that it was mnemonic in that the alpha character for add was A and S for subtract and M for multiply. But there was another new instruction called Edit and this is for printing out primarily numeric amounts in money: floating the dollar sign, or filling in with leading zeros, putting in commas, adding a credit sign or a minus sign on a debit. These are things that were new. You couldn't do that on a normal accounting machine because there was no facility for printing all these characters in variable positions. But the WWAM machine had invented a thing called edit where they could by plugboarding control the appearance of a printed amount with a floating dollar sign or leading zeros or commas or decimals. It was good, complicated. I don't know how limited it was. But I needed to have an edit instruction that was at least as good or if not, better, implemented by purely stored programming, no control panel, no plugboard. I worked on this thing for days struggling with how to do it and wasn't getting anywhere. Then all of a sudden, one morning I was in there shaving and the whole thing just popped into my mind how to do it. I knew everything about it. So I rushed into work and I grabbed one of my logic designers and laid it out for him and I said, "Now, you implement it." *<laughs>* He did and we both got the—I got the—well, we both got the patent and we got a $10,000 award for that. That was pretty good. So that was edit. Now…

Garner: Main memory.

Underwood: There were holes in the instruction set. I hadn't defined an instruction for every character or every combination of characters. It was more artistic than scientific. I didn't anticipate any more than 4,000 bytes of storage. I couldn't see any need for it. It's like Bill Gates says, "There'll never be any need for more than, what 432 [640k] bytes [of memory].

Garner: Yes. Well, your image was that people were going to be hand coding all the instructions. And 1,400 was quite a bit just to hand code, so 4,000, that seemed like a lot.

Underwood: Yes, it was. Subsequently, as things go, there was a demand for much more storage up to 16K of memory. Well, I hadn't anticipated this. So there was nothing in the architecture to take care of it, to address these additional positions. So I had to cobble something up. What I did was use the zone punches in the card or the zone bits in memory as a binary indicator of the 4K increment of memory. That was kludgey but I saw no other way to do it. I apologize for that, it was a—I just didn't see far enough ahead. The other thing, as I said, was that I hadn't anticipated people being bothered by undefined opcodes or combinations.

Garner: Yes, who would know, years later they would be emulating the 1401 instruction set on IBM 360 machines?

Underwood: Never anticipated that either.

Garner: So we have the instruction set pretty well defined. Jim Ingram is managing the engineering group. Then what happens? How did we get this machine done?

Underwood: That I don't know. I don't remember. *<laughs>*

Garner: You were part of it in some way.

Underwood: Yes, I was in there some way. There was a reorganization or the start of a new organization. I just vaguely remember. I was the architect. Jim Ingram was put in charge of implementing and moving the program into production. Chuck Branscomb replaced Ralph Mork, and he was the new accounting machine manager. Great choice, because Chuck and I were both veterans, you might say, of unit-record accounting machine usage. We both worked with—in fact Chuck was a user. He knew what the problems were and recognized that what I had was a great solution. He really just fell right in line with the whole thing. Then we gradually got other talent. I don't know where they came from but they showed up all coming together with a great deal of talent and enthusiasm and they just made this program just roll right along. It was a wonderful thing to see happen.

Garner: I understand that at one point they worked three shifts?

Underwood: Well, yes, I didn't have to do that. *<laughs>*

Garner: You kept your smile.

Underwood: But, I mean these guys were dedicated. They'd work hours and hours and hours of overtime and three shifts and anything to get this product together and out the door, because they could see—well, they believed, because I sort of spread the word—that this was going to be a great machine.

Garner: How did you know it was going to be a great machine?

Underwood: Oh, well, I liked it and I knew lots of other people would like it. Then thank God we found Shel Jacobs. He was the product planner, young, new, but he knew the customer. He'd been in sales for some years and he came in and he looked at this program and he just—he just took it over *<smiles>*. The usual corporate product planning group viewed the 1401 as just another machine and it might sell a few hundred here and a few hundred there. Shel came in and said, "My God, this thing is a blockbuster. We're going to sell thousands of these things. It's going to sell like hotcakes. It's going to go." Shel was one of the major, major driving forces for the success of the 1401. He saw the future and he convinced product planning—or the forecasters, "This is going to be huge." So they finally conceded that we'd sell 5,000. And based on 5,000 it could be a growing program. One day, Tom Watson walked into my office and asked me if this was all true, "Are we going to sell that many?" I said, "Oh, easily." And he said, "Well, all right." So he walked out and made the program official.

Garner: Isn't that remarkable that the…

Underwood: Yes, it was. I never saw him again.

Garner: …CEO [Chief Executive Officer] of all IBM, the famous Tom Watson Jr. came into your office—so he knew you were the architect…

Underwood: Yes.

Garner: He asked you, "Is this going to sell like hotcakes?"

Underwood: Yes. And I said, "It sure is."

Garner: Was he expecting a different answer? That's a remarkable story.

Underwood: I don't know. He just wanted a little confirmation.

Garner: He wanted assurance. Yes. Because you guys had to productize it, which would involve more resources and…

Underwood: Oh, yes.

Garner: Did he have a sense it would be a new machine, it would be different from the old, that it was stored program? Do you remember that at all?

Underwood: Well, I think he must have because he certainly had some reservations. But his information came from the lower ranks, of course, and all these people on up to him really believed in it. So, he just wanted one final good word.

Garner: Do you remember where you were on announcement day when they announced it? Do you remember seeing the announcement video or the day they announced it?

Underwood: I do not remember seeing the announcement. I may have, but I don't remember. What I do remember, is *Time Magazine* on the 5th of October [9th of November], 1959 came out with an issue based on the 1401 and the cover was a three page fold out of the 1401 System on a mountain top saying something like "A New World" or "A New View" or something. <*laughs*> It was great. So, what was I supposed to—?

Garner: I think after the 1401 was announced, it went into product engineering; there were probably lots of bugs. I don't know, did you help with that process, do you recall?

Underwood: Actually, I don't remember too much after that. They had all the glory of this announcement and that was in '59. I certainly stayed on the program for a while and did mundane things, I don't remember what anymore. But the…

Garner: Well, actually, I had one question for you. One thing the 1401 did was months before announcement, so in early 1959, Chuck decided to add the tape control unit.

Underwood: Oh, yes.

Garner: From what I can tell, Shel Jacobs and Chuck and maybe you, did not anticipate that so many customers would buy the tape system so they could stop storing data and punch cards.

Underwood: That's right. We didn't anticipate that either. So we were just lucky I guess.

Garner: Yes. So at some point then, you came out west, you left Endicott.

Underwood: Well, I came out west in 1962 to work on a process control computer and left the 1401 all behind me, all the records, all the memories.

Garner: Clean sheet of paper.

Underwood: Yes, cleaned the decks and go on to something new. So I worked on the 1800. Interesting thing is the instruction set for the 1800 was created by a man named Arnold Spielberg; that was Steven's dad. So I got to at least know him.

Garner: Wow.

Underwood: I worked close with him for a while.

Garner: So you worked on the data path of the 1800s.

Underwood: I did the data path work. I didn't really understand enough about process control because I had never been in the field. If I had been in the field and had worked with it, I would have done a better job.

Garner: You also worked on the 1130? Or maybe they were so related that…

Underwood: Well, that was peripheral work. I don't think I was directly involved. I may have been, but I don't recall.

Garner: Okay.

Underwood: For some reason that period of time in there is kind of vague.

Garner: But then there was this instructional computer.

Underwood: After I left the 1800 program, I went into the instructional…

Garner: Computer aided, is that what it was called?

Underwood: Well, computer-aided instruction, yes. There were two phases to this program. One was a large scale central processing machine with lots of peripherals, carrels for the students. They were all audiovisual carrels, they had a typewriter, earphones, tape reader and supposedly a student would pick up a cartridge and stick in a machine and put on earphones, get instruction and type notes or type answers and could read. Then there was the standalone carrel, the secondary education machines that did not rely on a central processor, they were standalone and they also had audiovisual capability. But they needed something that the other ones didn't; they needed a high capacity storage. Well, I was working in this area, not on a big one, but on a little one. So one of my cohorts, Bob Treseder and I we invented a floppy disk just for this machine.

Garner: Tell me more. How did you do that?

Underwood: Well, the tape wasn't viable because you had to scan back and forth to look for data. We wanted something that would spin, be magnetic and hold all of 32 kilobytes, 32 kilobytes. So we built these things. It was about 10 inches in diameter.

Garner: How did you get that 10 inch piece of magnetic material?

Underwood: Well, magnetic tape is made in a wide roll just like newsprint and it's all mylar and it goes through a coding process and then it goes through a splitting process and then a rewind and the reels for the individual tapes. Our plan was to stamp out circles and to spin these circles and to read and write. And that’s what we did.

Garner: So you built a prototype and…

Underwood: Built a prototype.

Garner: …some kind of magnetic head.

Underwood: Yes. Implemented the read/write electronics with audio, read/write heads. So they weren't very fast, but it was still faster than anything else we had. We put in an operating system on the disk.

Garner: APL [A Programming Language] based I understand?

Underwood: It was APL based. Now APL is “A Programming Language.” It was invented by a mathematician named Ken Iverson and it became a very popular programming language, just loved it. There was a lot of art in the design of that program. Once again the logic was all serial by character. I had a very bright engineer working for me and he wrote 10 pages of logic, Boolean algebra to describe the logic. It was perfect. Perfect. We built according to his spec and it worked exactly the first time.

Garner: What happened to this floppy then? You were saying you had—

Underwood: Well, of course back then all this stuff was a great idea, but we didn't have the right technology. It was too expensive. The program was killed and the money diverted to the other big program. This spelled the end of the floppy disk. However…

Garner: Yes, not quite the end.

Underwood: However, we had an engineer who saw the value of a floppy drive and he took the idea, left IBM and started his own company. Multi-billion dollar company, by the way.

Garner: Was this Al Shugart?

Underwood: Al Shugart. Yep.

Garner: Now who had the idea of putting the floppy inside of a paper carrier? Was that before Al took over?

Underwood: When we stopped the small secondary education system—and that was the end of the floppy— the disk drive, the hard drive people picked it up and made some improvements on it. They are the ones that conceived of a smaller disk inside an envelope. Great idea. But IBM said, "No, we're not going to build such a thing. It's going to ruin us on our punch card business."

Garner: But already, customers were transitioning to tape storage. So…

Underwood: Yes, they were…

Garner: Those people must not have seen that happening or they weren't thinking, right? Because the floppy was just a better tape and it came directly from tape, which is even more ironic.

Underwood: There were a lot of corporate decisions that us peons didn't understand, you know. There must have been a good reason for it, but I don't know what it was.

Garner: Yes.

Underwood: So Al goes off and makes a fortune with floppy disk drives.

Garner: You told some story about Merlin being left at the loading dock once. I don't know if that's a generally known story or not.

Underwood: I guess Al thought he was going to get away with it and he did. But there was an incident preceding Al's departure. IBM had developed a brand new hard drive with multiple platters that was removable and it was called Merlin, very high speed, very high capacity. They had built—you know, they had complete documentation in the files and they built a model, a prototype and they set it on the back loading dock one night intending to move it somewhere the next day. Well, during the night somebody took it right off the dock and the documentation turned up missing the next day. Well, IBM found out who did it and they called them "The Dirty Dozen." And they took this stuff and they started their own company and of course made billions again.

Garner: Do you remember the name of the—?

Underwood: And IBM couldn't touch them. Can you believe this? They couldn't touch them because they couldn't show that they had exerted due diligence in protecting the documentation or the model. They just set them out there in the public's hands and so there was no recourse for IBM, amazing. Well, things changed at IBM after that. You couldn't know what the guy in the next office was doing anymore. <*laughs*>

Garner: Wow. Then you continued to work in San Jose until about '69, I think, and <*inaudible*>.

Underwood: Well that's right, '70. I think in 1970 I was approached by an entrepreneurial group. It had a grand idea for a business and it sounded good at first and I was enthralled and I thought it was my turn to earn some money. So I left IBM and went with this group. The thing is it was a—the idea for the company was to do something in the electronic field that would support them while we developed our own product. When we did that we would have a manufacturing facility already in place and we could just go. That sounded great to me. Unfortunately, the electronic business they got into was back panel wiring. That was the level of technology—that's how back panels were made at that time, no printed circuits. Back panels were wired automatically by a wiring machine called the Gardner Denver. What one would do would write out a wiring list, process it on a PDP-11 [Programmed Data Processor] and then input the program to the Gardner Denver and it would automatically establish all the wiring routes. Well, we had two machines and two girls, four hands, four girl hands to run these things and make money to support five well paid executives. I finally realized what was going on. Nobody else seemed to know. They failed. So I had to find a job. So I went back to Lou Stevens at the Los Gatos lab and said, "I made a mistake. I want to come back." He said, "Well, you know, nobody's coming back to IBM." It came right down from the president. "Anybody that leaves in this—after the Merlin…

Garner: Out for good.

Underwood: …fiasco, can never come back." I says, "Well, yes." Lou said, "Well, I'll go talk to him." So he talks to the president and the president says, "You mean, Fran wants to come back?" He says, "Yes." He says, "Okay." <*laughs*>. So…

Garner: You were loved.

Underwood: So I got back into IBM.

Garner: And you worked on the 3800?

Underwood: At that time IBM was developing the 3800 laser printer. Oh, God, it was a huge machine. The printing mechanism was six feet long and about four feet high, had a big drum in it, about 30 inches in diameter and about 16 or 18 inches wide with a coil of photosensitive mylar inside the drum then out through a slot, around the outside and back into the take up reel. As the printer did its thing, every 20,000 pages, a new surface would automatically be taken off of the supply roll and taken up by an intake roll. That would cause a shift in the balance. This drum always had to be balanced, so there was a mechanism in there to do that. When I came back, they assigned me to be liaison engineer between the three development departments. One was mechanical, one was electronics and the third was the photo process. So I did my thing; coordinating. I did some design work. I actually redesigned the inside of that drum so it was easy to balance, but the mechanism they had was ridiculous. I worked on the stacker it was fanfold paper in and when it came out it had to be burst and stacked. So I worked on that stacker.

Garner: And then after that program you went to Rochester.

Underwood: Yes, that's right. That project—my work on that thing came to the end where it was announced—it was something to see the halls of the laboratory aligned with thousands and thousands of boxes, cartons of print paper, it consumed us. It would take 14 minutes to chew through a box of paper. And <*makes a "vroom" sound*>, it was really fast. So then I went to Rochester to work on…

Garner: Was it called Triple Play?

Underwood: Triple Play. Yes, Triple Play. I guess that was some kind of a terminal and I don't remember much about it.

Garner: Then you worked on some kind of communication project after that?

Underwood: Yes, after that I was transferred to work advanced communications which I had no experience with, knew nothing about it.

Garner: That hadn't stopped you before.

Underwood: Well, no, but remember, I did know an awful lot when I started on the 1401.

Garner: So finally, the last straw was they ask you to go to Austin.

Underwood: Exactly. They were going to transfer this Triple Play to Austin and they wanted me to go with it. Well, I had 32 years in at that time. I said, "No, I'm going—I think I'll retire."

Garner: You had been a California boy all this time.

Underwood: Yes.

Garner: Now, ironically, you live near Austin today. So then what happened next, you decided to leave IBM?

Underwood: I decided to leave IBM and retired back in California in Auburn, just north of Sacramento. Oh, it was a lovely little town.

Garner: You didn't exactly retire.

Underwood: I started my own business as an engineering consultant. That was great. I picked up—or I hooked up with a guy who was building scrap metal recovery plants. I hooked up with him and I started to design these things. They were comprised of conveyer belts that would convey all the scrap through a tumbler to separate out all the stuff and then onto another conveyer belt, past a magnet, would filter out the ferrous material and then into a cyclone which is eight feet diameter and 60 feet long with a fan powered by a 250 horse power motor sucking air and separating all this fluff out. I designed 15 of those machines and installed them in Ireland and England and the United States.

Garner: Did you travel to those places?

Underwood: No I didn't.

Garner: So that client finally wanted to move though or something to Nevada, right? That client wanted to move to Nevada?

Underwood: Oh, yes. He couldn't stand the tax structure in California so he moved his business to Nevada. So then I started working on disk drives, testers and did a lot of programming there, I designed a tester, electronics and the packaging and wrote the code all in machine language—Assembly language actually…

Garner: How many lines?

Underwood: …for the 8085 processor. Half a million lines of code.

Garner: Five hundred thousand lines of Assembly language. Now compared to programming the 1401, was that kind of painful because the 8080 instruction set is not as simple as the 1401 instructions?

Underwood: Well, true, but I had no trouble with it.

Garner: Yes, yes.

Underwood: And then I started—I thought I would try the C language. I got C sick. I couldn't hack it.

Garner: You wanted to stay in Assembly language.

Underwood: Yes.

Garner: Wow.

Underwood: So I quit that and I found a job in the paper for a structures engineer. Well, the thing is all throughout my career in IBM I had a separate tangential career as a machine designer outside of IBM. I had friends who ran small factories or machine shops or what have you and I would design tools and industrial machinery for these people. I did that for years. All the time I was at IBM I always had this work on this side. So the work I did in scrap metal recovery was largely structural engineering. So I saw this ad and I went for it. It turned out the guy wanted a ski lift designer. He looked at my drawings for this scrap metal recovery, hired me like that. So I worked in that field for 15 years.

Garner: Designing ski lifts.

Underwood: Designing ski lifts. I designed an awful lot of them all over the world.

Garner: The company was called Cable Transport Engineering or something like that?

Underwood: It was called Cable Transport Engineering, yes, C-Tech. They were bought by a Swiss company and then that company was bought by another Swiss company and I finally retired at age 78 or something like that.

Garner: You started working at C-Tech in 1988.

Underwood: Right.

Garner: So you worked there for how many years?

Underwood: Fifteen.

Garner: For another 15 years. Wow. How would you compare that part of your life to your…

Underwood: Oh, I think it was the best.

Garner: Wow. Did you ever actually sit on one of your ski lift designs?

Underwood: Oh, yes.

Garner: Did you ski? Were you a skier?

Underwood: Oh, no, I didn't ski, but I went to many of the sites and rode them.

Garner: Okay.

Underwood: I had a grandson, about four years old and his dad took him skiing. His dad was a skier. Maybe he was eight, I don't know. But he took him along and while they were in among the ski lifts, his dad mentioned that I designed ski lifts. The little kid looks around and he says, "Wow, I sure hope he designed this one."

Garner: Wow. And Fran, I wanted here to show this to the camera. <*Showing photo of a small aerobatic pane at San Jose runway*> Is that you in that—?

Underwood: Yes.

Garner: And show us towards the camera. What kind of plane is that?

Underwood: This—another aspect of my life. I used to fly a lot and I was a one fourth owner of this Stearman Trainer, it's a biplane, two open cockpits and it was a World War II trainer. Today they use it as crop dusters and as aerobatic airplanes. Well it was an aerobatic airplane, so I learned to do aerobatics and I had just a ball in this airplane. Unfortunately, when I had it for a couple of years I was transferred to Rochester. There was no way this airplane was going to go to Rochester, so I sold my part and that was the end of my flying. I loved that airplane.

END OF TAPE 3

START OF TAPE 4

Underwood: I was designing the 1401 operators console, I had to work with an industrial design department and the manager of the industrial design department at that time was a very competent artist and he taught me how to paint.

Garner: Okay, why don't we do that—stop because we'll do that on tape.

Underwood: I learned to paint with gouache or tempera and acrylics. Gouache and tempera are opaque watercolor. I just had a ball with that, painting airplanes and old wheels and all kinds of stuff. But that takes a lot of concentrated, devoted time and finally I got to the point where I didn't have time anymore. I had to give that up. So there's been a lot of different lives that I've lived.

Garner: You know, Fran I was thinking back, I had one more question about the 1401 Do you remember when you came to San Jose they were working on a follow on to the RAMAC [Random Access Method of Accounting and Control] like the 310 or something like that and you came to San Jose.

Underwood: Yes.

Garner: You had to somehow convince them that the 1401 was going to be the future and not the RAMAC, how did you—do you remember this?

Underwood: Well, yes, you're right, there was a competitor program ongoing in San Jose based on the RAMAC and I had worked with these people in the past, so they were friends. So Jim Troy, the lab manager, who's a big champion of the 1401 said "We've got to squash that thing out in San Jose." So we went out en masse, myself and Chuck and Jim Ingram and Jim Troy, we went out there.

Garner: To get to here, where we are now.

Underwood: This SPACE machine is the way to go and they said no it's gotta be this way and so we started comparing and soon, very soon became very clear that they were way over priced and weren't really hitting the market. So Lou Stevens and Jim Troy had to battle this thing out and Jim Troy being pretty strong, he won...

Garner: Well, Jim Troy had the energy to win but your design allowed for a low-cost machine.

Underwood: That's right. Well, I he had something to work with, he didn't do it just by tricks.

Garner: I think basically, Fran, I like to characterize that story as you brought to an end the development of processors in San Jose, they kept doing the disk part of the RAMAC, so you turned San Jose from a complete computer engineering site to just disks and that was because of your 1401.

Underwood: Exactly right.

Garner: That made a big difference in the future of San Jose.

Underwood: Yes I did, I made a big difference at IBM.

Garner: You did indeed; you invented the future where it became the future.

Underwood: Yes but you know, I was only one of many, all the other people that worked on that program made—like Shel Jacobs—made significant contributions. Without Chuck there'd have been no program, without Shel there would have been no program. No matter how good the 1401 design was, if those guys weren't there to support it, it wouldn't have happened. They were just as important as I was.

END OF TAPE 4

END OF INTERVIEW

