Interview with Marcian (Ted) Hoff

Interview with Marcian (Ted) Hoff

March 3, 1995

Los Altos Hills, California

Hosted by Rob Walker

Co-Founder, LSI Logic Ltd.

Transcription by Dag Spicer

Program in History and Philosophy of Science

Department of History

Stanford University

[Start of Tape: 0:00]

[Garden tour omitted approximately 3 minutes.]

[Start of Interview: 3:04]

RW: Today weíre visiting Ted Hoff in his Los Altos Hills, California home. Ted is often credited as being the inventor of the microprocessor. I knew Ted while I was at Intel from 1971 to 1980. I considered him one of the most brilliant semiconductor scientists of our time. This [interview] is a part of the Silicon Genesis program and is part of Stanfordís Silicon Valley program. So, Ted, thanks for taking the time to give us an interview.

First, could you tell us a little bit about your early years, before you went to Intel.

TH: Well, I donít know how far back you want me to go but I was born in 1937 in Rochester, New York, and I got interested in science at a fairly early age, primarily because my fatherís brother, who was just 12 years older than I was, was studying chemistry and chemical engineering. So I thought that looked like a marvelous subject. I loved the magic you could do with chemistry and pretty much decided to follow in that career until my uncle advised against it. He said that unless I went into chemical engineering, as opposed to chemistry, he thought the job market didnít look very good.

Now, partly because when I was twelve years old my uncle had given me a subscription to *Popular Science* magazine and I answered an ad for an Allied Radio catalog, Iíd gotten interested in electronics. So that was sort of my second choice and so, well, when it came time to go to college, maybe *thatís* what I should pursue...

So I went to Rensealler Polytechnic Institute. I studied electrical engineering there and I graduated in ë54. And then, not ever having been west of Niagara Falls, I thought it was time to see a little bit more of the country and decided to do graduate school and I was fortunate to be accepted at Stanford University. So I came out to Stanford, got my Ph.D. there in 1962 and I stayed on working with Professor Bernie Widrow in the area of neural networks. And while I was at Stanford, I got interested in integrated circuits. I had been interested in computers for some time and we used computers in our work on neural networks.

So one day, I think it was around some time in early 1968, I got a phone call. It was from a fellow who I didnít really know, but I had heard of him, and I met him once I believe--his name was Bob Noyce. And he told me he was starting a company and would I be interested in possibly becoming an employee of his new company? Well, I thought it might be fun to try and I interviewed, and got the position, and became employee number twelve at Intel Corporation.

RW: Now, what was ìIntelî short for?

TH: Well, I guess ëINTegrated ELectronics.í I donít think I ever heard it explained. But, Intel started out with the idea of building semiconductor memory and, in fact, when I was being interviewed, Bob asked me the question: ëWhat did I think the next area for semiconductor development should be?í... and I had been thinking about this a bit and had talked to a few people and I said ëmemory,í and I guess that was the right answer because that was what Intel was planning to do.

RW: Thatís interesting. Now in your class at Stanford, there were a lot of other people that... was Shockley teaching there at that time or had he left?

TH: I donít know whether Shockley was teaching there at that time. I never had any courses from him but when I came to Stanford, there was a fellow there, Dr. John Moll, who had been at Bell Labs and when I was at RPI I had heard of him through some papers which were very important references for I thesis I did. So I was pleased to find that he had joined the Stanford faculty when I came to California.

RW: Well, Paul Low was there... went on to fame at IBM...

TH: Yes, Paul was in the same group that I was in working with Bernie Widrow in the area of neural nets and adaptive systems, and adaptive filters.

RW: Jim Kofort and Ed Jones who were the guys who essentially invented computer-aided design for ICs.

TH: Yes, there were also in our group and we were all working in this area of neural networks at the time.

RW: Which kind of went away for a while.

TH: Well, it went away for a while but itís popped microprocessor again recently. In fact, a few years ago I was invited by Bernie Widrow to sit in on a study that was sponsored by DARPA of the state of neural net research.

RW: Well anyway, so youíre at Intel... so what were your first projects there?

TH: Well, my duties were primarily what they... they gave me the title: ìmanager of applications research,î and it really had two... two areas. One was to help customers use products developed by the company, which is the traditional ëapplicationsí role, but the other part of it was to talk to customers and find out more about what they were trying to accomplish so that I could play a role in helping to define the next generation of products.

RW: So, how did the microprocessor project get going?

TH: Well, thatís... One of the characteristics of most of the products that were being developed at Intel is... they were what we call ëproprietaryí products. The [customer's] company had ideas, defined the product, and started the development. And Intel also had a policy of not really saying very much about products that were in development. So that meant there could be a very long design cycle. From the time you start to do the engineering until you release the product, then the customers start to look at it, it takes *them* a while to design the product in, and that can mean there is large investment being made in engineering and very little in the way of revenue being derived from it.

So, at one point it was felt it would be desirable to undertake some *custom* developments where you work closely with your customer, develop the product to *his* specification. And one of our first programs which started... letís see Intel was started in... well, officially, I guess it got started about September of ë68... and about April of 1969, we undertook to develop a family of chips for a Japanese calculator company. And there were several names that this company used, different corporate entities, but the calculators were to come out under the name ëBUSICOM.í

So, the contract was signed, the engineers came over from Japan to work with Intel, to wrap up the last details of their design, and they arrived sometime around June of 1969. Now the only role I was supposed to take was to work with them and just help them find whatever supplies and so on they might need and sort of act as an interface, or liaison for them.

But, in the course of that, I began to see that there could be some *real* problems with the design they were proposing. There were some pretty aggressive cost targets, part of the problem was that they were going to be using, like 40-lead packages--those were expensive packages at the time. And so, I was concerned that it was going to be very difficult to meet the cost objectives for the program. So, it seemed to me that there were some characteristics of their original design that were quite positive. For example, they used read-only memory to customize it, but it didnít seem to me that they were making very effective use of the read-only memory. The level of program they put in the memory was a very high level program which meant that there was an enormously complicated logic to implement the actual functions of the calculator family.

So, my first suggestion was just ëwhy not make better use of that memory by simplifying the functions that youíre going to perform in the calculator and do more with the read-only memory?í They werenít very interested in making any changes, the Japanese engineers were committed to the design they had and they essentially told me to stay out of their way. But Bob Noyce was encouraging, and so I continued to work on that and actually in the course of that, over a relatively short period of time, felt that the way to do it would be to make a very simple, general purpose computer. And I started to develop the instruction set for that computer and that instruction set was essentially what became the instruction set for what we announced as the ì4004î microprocessor.

But with that instruction set I was able to show that it was possible to scan a keyboard, debounce the keyboard--in other words, you do multiple scans and check to make sure that you get consistent results on successive scans--to maintain a display, and to do the calculator arithmetic. And that would be done, and my thinking was to do it in binary, but with a binary-coded decimal correction so that you could binary or binary-coded decimal arithmetic. And again, I worked out that instruction set and a configuration which was initially to have two chips that would make up the central processor. There would be essentially a central processor and there was a timing chip that was going to generate sort of timing for the whole system. And then there was a read-only memory chip, to be developed, and a read-write memory.

And again, the idea was basically rejected by the Japanese engineers but again, Bob Noyce and others at Intel, encouraged the development. And eventually it went to the point where... I believe it was in September, actually September 16th and somewhere I have a letter, in which we made--our marketing department made--essentially a formal proposal to the Japanese that they consider this Intel design as opposed to their own engineersí design.

And Japanese management came over from Japan, and I believe that was in October--I donít know the exact date--but sometime in October ë69 and essentially both groups presented their approach and my group, which at that time consisted of myself and Stan Mazur, essentially proposed this general purpose computer approach. And one of the advantages we pointed out was that it wasnít limited to just calculators but it had other applications as well. And so, it turned out much to our surprise and delight that the Japanese management agreed and chose the Intel design as the one to be pursued, and so as of October of ë69 we were committed to build what would be a computer on a chip. And in the course of the subsequent refinements of the design, the timing chip was combined with the central processor to make a fully stand-alone central processing unit. They only thing it required were a couple of clock signals and power supplies.

RW: So, how long did it take to get silicon?

TH: Well, the big problem was I was not an MOS designer. And it would have involved a lot of training and I had other obligations as well. I *did* do a brief study from what little I did know about MOS to estimate the number of transistors that would be required--to make sure that we were within reason and it looked like it would be somewhere around 2,000 transistors.

But Intel only had a few people who were capable of MOS design and so they started to do some recruiting. And it took almost... well something like six months before they found somebody and then, I believe it was April of 1970, when Federico Faggin finally joined Intel. I believe he had been at Fairchild before that. Federico worked at just a furious pace, I mean, he really, really went all out and in the space of about nine months, designed the three major chips. And by that time we actually had four chips that were defined. One was a very simple static shift register that was just used as an I/O-expander. But there were three complicated chips to be designed: a central processor, a read-write memory, and a read-only memory. And both the read-write and read-only memory had I/O ports on them, so they were more complicated than just a simple memory chip would have been.

I believe the last one that Federico did was the 4004, and I think he had first silicon on that sometime in January of 1971. There were some problems and he fixed those and second silicon, I believe, came out in February of ë71 and the processor worked at that time.

RW: And *that* was the microprocessor.

TH: That was the first microprocessor, yes.

RW: How exciting. Of course, you guys didnít realize you were making history at that time...

TH: Well, probably not that we were making history but that we felt that it was important. And one of the things that we did realize was that we were... a lot of us... that included some of the engineers, Federico, and myself, we had various projects going on in the lab and we realized that if these chips were available it would make our ability to design things like test equipment and so on... it would make it a lot easier to have those chips available. And, in fact, one of the projects I had was to build a ... I guess what youíd call a ëPROM burner,í but a device for blowing the fuses in a bipolar PROM or programming the floating-gate devices... of course about that time the floating-gate EPROM was being developed by Dov Froham-Benzkovksy who was at the next bench at Intel. It turned out that was a very fortuitous development because those EPROMs turned out to be very useful devices to help complete developments that used microprocessors. The microprocessor, we felt, was very important in the sense that *engineers* would find it a useful device, and it would make design easier, and allow you to do things that you couldnít--or would be reluctant--to do in a design done with conventional logic.

Things like the interface to the human being. We could do things with the microprocessor like not allow you to set the switches in something in a way that would be destructive to the equipment. Itís much harder to do that when you are just, you know, providing switches to, you know, to somebody and they can flip them in any manner.

RW: Well, today when we think of a microprocessor we think of something that sits on a desk, that is, a computer. Yet the first ones were really *controllers* were they not?

TH: Thatís correct, they *were* controllers and our imagined use for them was in the area of controllers. In fact, the kinds of things that we talked about--applications for microprocessors--were things like elevator controllers were the example we used to give ... but we did have applications of that type. People, I believe, were using them for ... like, gas pumps, in other words for gasoline stations where they were starting to have centralized gathering of information--what was going on at the pump--so, in effect, having a small processor available for that application was really ideal. And things like, you know, traffic control, traffic signals, the counters and so on for control in parking lots... So those were the kinds of applications that we envisioned the parts for.

At that time, we werenít really looking to replace the general purpose computer. For one thing, the microprocessors of that first generation were *very* slow devices and if you were to go to an application that required a large amount of memory, you were going to have to have a big investment in memory, because memory was quite expensive then. So it wouldnít necessarily make sense to use a microprocessor in that environment because you werenít using your memory effectively, you might do better to spend a few dollars more and use a processor that was built out of TTL such as youíd find in a minicomputer of the day. So, for that reason, we tended to limit our imagined use of these devices to applications that could be done with very modest amounts of memory.

RW: Now did BUSICOM ever build the calculator?

TH: Well, BUSICOM I believe did build some calculators but BUSICOM had financial problems. They were always cash-short, and I believe they were not able to really successfully market the product so they didnít use the volumes that they expected. And in fact, one of the aspects of that... originally, Intel marketing was very reluctant to offer the products to anyone other than BUSICOM. The feeling was that BUSICOM *would* use the entire production that Intel would be capable of manufacturing. But there were a number of us there who felt ... I was one, Federico was another ... who felt that this product was much more useful than just for calculators, that other people would find it, and that Intel should offer it as a proprietary... you know, an Intel proprietary product. The only problem was that when the contract was written with BUSICOM, even though it was totally an Intel, in-house design, the contract was written in such a way as to give BUSICOM full rights to the product. So it was actually necessary to go and negotiate a new contract, and, in fact, BUSICOM opened the door in May of 1971 when they said that the calculator business had gotten *so* competitive that they wanted to see if they could get *lower prices* for the parts than they had originally negotiated.

At that time, I know I was one who went to the marketing people and said: ëIf you canít get any other concession from them, at least get the rights to sell it to other people.í And the marketing people were *very* reluctant to even consider that as an option, but they did. And they did get the rights--with certain restrictions, there was a restriction that said ëcouldnít be sold to other people who were manufacturing calculators.í

RW: Now, was that Ed Gelbach, when youíre talking marketing?

TH: No, this was before Ed Gelbach came to Intel, in fact, the director of marketing at that time was Bob Graham. And, I believe the... I donít know his exact title but sort of like, uh, you know, very high up in the marketing group... was also a fella named Bob OíHare. And, I mean, both of them, essentially told me that it was difficult enough to find salesmen to sell memory products and to hire them into a semiconductor company and they felt that it would be impossible to hire qualified people to sell computers. And so for that reason it was not reasonable to consider having the company go into the computer business.

And furthermore, they felt that the, in fact, one told me... I donít remember which one it was who told me this but it was one of the two gentlemen, said something to the effect that ëlook, weíre *late* coming into the computer business... weíd be lucky to get 10% of the market. Now they sell maybe 20,000 minicomputers a year. That would be 2,000 chips a year and for 2,000 chips a year, it was not worth all the headaches that you would have if you had to support computers.

RW: And of course that goes back to the first mainframe computer. The initial estimates were that you could sell somewhere between ten and a hundred.

TH: I might point out that one of the three engineers from Japan, a fellow named Matsotoshi Shima, gave a talk a few years ago and he quoted the use of microprocessors and microcontrollers in Japan and the number he quoted was 600 million...

[video edit drop out approx. 0.5s]

RW: ...to get the 4004 to market?

TH: Well it took actually quite a while for us to get the 4004 to market. Now, we had the rights to sell it as of May of 1971 but there was great reluctance, not just in marketing but elsewhere in the management of the company to take it to market, and one of the big concerns was how we would support our customers. In fact, it was felt that this could be a real problem. There was a concern about, you know, how even the sales force would handle the products.

So we were developing some tools, things like assemblers and the like, and we managed to have some contacts with some universities that also provided some tools for us, and we went outside also for their development. We were developing some design aids for our own use in-house, and we felt that those would be useful for our customers. But, I finally took the position that weíd probably just have to let some of our customers be on their own but we could probably support maybe a dozen or two dozen of the largest of them.

One of the factors, though, that I think really helped get the microprocessor announced was a change in the marketing department. As of, I believe it was August of 1971, a new director of marketing came on board. That was Ed Gelbach and he came from Texas Instruments... and apparently was not nearly as afraid of the microprocessor, and he may have been aware of some developments that were going at Texas Instruments that we didnít know about at the time but subsequently found out about those developments. But anyway, Ed was much more open and, as of, I believe November of 1971, a formal announcement was made. And in that announcement--it was not a modest ad--the statement was made ëAnnouncing a new era in integrated electronics ... a microprogrammable computer-on-a-chip.í

Now we debated a lot about using the word ëmicroprogrammableí because microprogrammable normally meant that the order code was something that could be adjusted... but there were those that said that ëno, ìmicroprogrammableî really meant that the code was cast in some form of firmware.í And in that sense, we figured, well we could use the term.

The ad was placed and there was a Fall Joint Computer Conference at which Intel had a suite, and Stan Mazur went to that conference and was at the Intel suite and he said customers came in and they were indignant having seen the ad, and he would show them a copy of the data sheet that Intel had produced and the customer thought it was, you know, probably just another 4-bit slice. And the customers would see the data sheet and say ëIt really is a computer,í and they were satisfied. In fact I have a copy of the first data sheet that we produced and this was also considered... it may seen very trivial today... but the design and the layout of this data sheet was considered, by our marketing department anyway, to be revolutionary. And the reason is that the first part of the data sheet is all application information and it isnít until you get to the back part of the data sheet where you get to the electrical specifications. And up until that time, it was traditional that the data sheet *started* with the electrical specifications and *ended* with the application information. But the feeling was that when you are producing a computer central processor on a chip, itís the application information thatís more important than the electrical information.

RW: Yes.

TH: And thatís why it was put first.

RW: Well, of course, earlier semiconductor products--their use was more obvious.

TH: That is true.

RW: You know, if itís a RAM, itís pretty obvious what it does but something like this, itís not clear what it would be used for initially. By the way, one of the other ramifications of that announcement was it put me out of the ASIC business because I was producing calculators and various application-specific ICs that the customers saw that the microprocessor could replace that, and so my group got canceled and I had to go to a different type of work and everything... So youíre responsible for that!

TH: Well I think of that as somebody overreacted. We always felt that the microprocessor wouldnít replace the ASIC business itself but maybe it would cut into the bottom end of it. Because where we saw it as primarily having application was in the realm where you couldnít *afford* to do an ASIC design where the engineering cost would just wipe you out. And in that case we felt the microprocessor will make a ... will find a niche for itself. But it still looked like it would be well worth it when you get into very high volumes to go to the ASIC ... so I think somebody overreacted in that case.

RW: [laughs] Well, now, were you involved in the 4040?

TH: Uh, not in the 4040 but there was a second, parallel development actually that went on at Intel and I was somewhat involved in that, and that led to the 8-bit processor, the 8008. And, in fact, it started very shortly after we had that meeting in October of ë69 that got us committed to the 4004. I think it was around December of ë69 ... we had a contact from--actually, Victor Poor of Computer Terminals Corporation--and I believe they were a user of shift-registers and they were building what they sometimes referred to as ëglass Teletypes,í computer terminals that used a cathode-ray tube instead of paper to, you know, present the information. And his initial request was for the *registers* of a machine that they were to build and I believe his original contact was primarily to Stan Mazur who was working for me at the time.

In the course of that, it seemed that to understand, first of all, how the registers worked, you needed to understand the processor and the processor was really not that much more complex than the 4004 processor. So, we made essentially a formal proposal that they let us do their whole processor as a single chip. Stan described it to me, he said Victor Poor nearly fell off his chair ... Subsequently to that, Victor Poor has said that *he* brought the idea for the microprocessor to Intel and that it was his intention all along to do it as a single chip.

But considering that the 4004 had already been defined, and *we* considered the 4004 to be the first microprocessor, and it was really... my firm belief is that the original request was for registers and it was our *counterproposal* to do the eight... well, what was ultimately known as the 8008 ... it had different number in the early days--it was called the ì1201î within Intel for throughout its development cycle. And that essentially was launched officially as a project around, maybe around February of 1970, and the engineer who worked on that was a fella named Hal Feeney, but he was also working with Federico Faggin. And so the work on that product went a little bit slower and, at the time, Hal was not as experienced an engineer, so I think Federico was working faster, moving along faster on the 4004.

So, then Computer Terminals got into some financial difficulty and so they apparently were not even likely to use the product---in fact, they never did use the product--but when they had written the contract with Intel to develop the product, they *did* give us the rights to sell *that* to other people. And we did have, I believe, contacts with Seiko of Japan as one of our early customers.

That product led to some rather interesting side issues having to do with intellectual property. In the case of the 4004, we felt *that* was an Intel proprietary design and therefore we thought it would reasonable to try to protect it. We spoke to our patent attorney, a fellow whose name was Stuart Lubitz, but Lubitz said he did not want to write a patent on a computer. He said they werenít worth it and essentially he refused at that time to write a patent.

And I can see why. Subsequently I came across the patent that was written on the IBM 360 computer and it runs something like, I think itís around 900 pages of drawings and another 900 columns of text, or something in that order... itís an enormous document. We said that itís too easy to design around a patent of that type. And I think he felt the idea of putting the computer on a chip was a fairly obvious thing to do. People had been talking about it in the literature for some time, itís just... I donít think at that point anybody realized that the technology had advanced to the point where if you made a simple enough processor, it was now feasible.

RW: Well, thatís right, the idea had been kicking around but the technology wasnít there and in fact...

TH: Thatís right...

RW: ... in fact, Hal Feeney came to me at... I was at Fairchild... and he said ëwe were working on this microprocessorí which turned out to be the 8008, ëand itís going to take us a long time--weíre doing this design all by hand... Can you do essentially a silicon breadboard using your standard cell technology.í And I looked at it and I couldnít... I couldnít do it... it would have been four or five chips...

TH: Yeah.

RW: ... to do, because the efficiency... and we also didnít have as good a process. So while the idea was there, you had to have a ... Intel really had the hottest process around.

TH: Yes, we figured our process was probably good for maybe twice the transistor count of any other MOS process around and, of course, the MOS process had about a four-to-one advantage on logic density... so no that... definitely... we felt we were... we were ahead in that sense and thatís where we felt we had a good shot at being the first to make a microprocessor. There were other papers, I mean, that were written... I think Iíve seen one as early 1965 talking about a computer on a chip, but it was recognized that until yield problems were solved, it was a long way off.

And, in fact, Iíve published an article in March of 1970... this is the 1970 International Convention Digest and this, as far as I know, is the first mention of the *feasibility* of the microprocessor in which, I said in here that ... um, the article is called ìImpact of LSI on Future Minicomputers,î and it said ìan entirely new approach to the design of very small computers is made possible by the vast circuit complexity possible with MOS technology. With 1,000 to 6,000 MOS devices per chip, an entire central processor may be fabricated on a single chip.î So that was essentially announcing there the ability of the technology to *do* a one-chip microprocessor.

RW: Itís also interesting how fast things move. That was in 1970, weíre filming this ë95, and now there are so many billions of these around the world. One third of American families have a personal computer in their home.

TH: And now people routinely put several *million* transistors on a chip and we thought we were lucky to have from 1,000 to 6,000. Thatís where the technology was in those days.

RW: So the 8008. Were you involved with that?

TH: Well, to a degree in the definition. In fact, Stan and I one day were talking and the 4004, you see, did not have interrupt capability. Now an interrupt is something where you have a computer and itís doing a task and you have this interrupt circuit which allows an *external even*t to signal the computer, the computer essentially stops what its doing, goes off and does a new task, and then picks up where it left off. So, we thought ëwouldnít it be nice if we could put interrupt capability into a processor?í

The 4004 didnít have it. The only way the 4004 could deal with I/O was through *polling* and by that it means that it keeps going out and sort of doing a conditional test that says ëis there anything out there for me to do?í And then if the answer is ënoí it goes back to what it was doing but it has to have that instruction programmed into the loop to check to see if thereís some I/O operation pending.

RW: Of course, thatís appropriate for a calculator.

TH: *That* would be reasonable in that environment. So we thought it would be nice to put in an interrupt but we did not want to add significantly to the complexity of the chip. So the question is: ëwhatís the least that you put on the central processor chip so you can add interrupt later on as an afterthought. And we came up with what we thought was a really novel solution. The idea was that we felt that we could put a switch between the processor and its memory. And on interrupt, we would, essentially, flip the switch and have it read a different instruction. In other words, instead of reading from memory, weíll force, force a CALL instruction, thatís a jump to subroutine. So once you forced a CALL to a subroutine, then the subroutine supposedly can save the state of the machine and it can, you know, do the interrupt task and then RETURN like any other subroutine RETURN does and it goes back and picks up where it left off.

Only one problem: the processor was asking for something from memory when we flipped that switch. And itís got a program counter thatís keeping track of where itís fetching from memory. And the program counter has been advanced, so itís been advanced as it fetched the CALL instruction, so itís going to miss a few bytes out of memory, what it shoves on the stack as part of the subroutine CALL is not gonna be the right number. The solution is just donít advance the program counter when youíre forcing this CALL instruction.

On interrupt, at the end of the current instruction, interrupt is acknowledged, during interrupt acknowledge the program counter isnít advanced. We didnít say what it was to be used for... thatís all we wrote into the spec.

Well, this spec was essentially for our customer, Computer Terminals. And Computer Terminals Corporation, we didnít know at the time, was looking to Texas Instruments as a second source for their products. So they took the design to Texas Instruments, which was fine--I mean, it was their right to do so--but what I do have a problem with is Texas Instruments proceeded to file for a patent on it.

Now we would have considered that unethical, for one thing the architecture of this machine was supposedly Computer Terminalsí with a few, you know, assists from Intel in terms of how it interfaced and so on to get it down to... it ended up in an 18-lead package.

But anyway, itís interesting... Texas Instruments included that business of the interrupt handling by not advancing the program counter. But, it turned out in subsequent work with this product at Intel ... as Stan and I (and our responsibility was applications, you see, not the design of the chip) ... we began to realize that even though we had this nifty little interrupt technique, it was going to take a lot of logic to force a CALL instruction because a CALL instruction was three bytes and that meant logic that had to take over from memory and then figure out how to sequence through these three bytes to set up the CALL.

But there was another instruction in the repertoire that was called RESTART. Now, the RESTART, as originally defined in that first data sheet, ah, first you know, target spec, was not a CALL instruction. It was just like a jam-load of the program counter to an initial value. But we thought about it and we said ëwhat would be the harm in making RESTART a CALL instruction, where the program counter is saved on the stack?í It just means that if you want to use it like a RESTART, you just never going to do the RETURN that would have gone with the CALL. So that doesnít hurt anything. You use it like a JUMP and you just donít bother to RETURN from it.

And the way the ah... it wasnít like it was going to waste memory because the stack, where you saved these RETURN addresses was actually a ring, like a toroidally-connected ring of memory. So, um, it essentially didnít matter where it started in the ring.

So we made that change to the spec. Now apparently, that piece of information never got to Texas Instruments. So when they wrote their patent application, they did find out that Intel serviced an interrupt by forcing a RESTART instruction. But the RESTART instruction in the TI chip is *not* a CALL. Itís a jam-load of the program counter--so itís non-functional--yet TI wrote that into their patent. So, in effect, itís fairly obvious that the people that wrote the TI patent didnít even know what they were copying.

But TI did assert some of those patents in recent litigation. And, however, it has come out that information showing that a lot of the information on some of those patents was actually copied from Intel. And I understand, as that information came out, they got very reasonable in their licensing negotiations! So as far as I know, it has not gone into court but it has been in to negotiation and it has gone very close to going to court.

RW: Well TI, in the last recession, the difference between their profit and loss was their licensing fees.

TH: Thatís right.

RW: They became very aggressive.

TH: Iíve heard that also, yes.

RW: Well, so on the 8008 so you were ... you had some ideas in it and then Hal Feeney implemented it. When was that introduced?

TH: Um, the product I believe was announced about April of 1972. Actually, there was more of like a pre-announcement of that maybe a few months earlier. When we came out with some of the literature for the 4004, we did mention this 8-bit processor that was coming along. And the 8008, and... with those exceptions of things like the interrupt and the RESTART instruction, was primarily an architecture that had been developed by Computer Terminals Corporation. They brought us the bulk of the instruction set.

And our customers began to... they tried to use the product... in other words, if they had a little application theyíd consider the 4004 but if it was a bigger application, they felt they should use the 8008. And it wasnít necessarily the most reasonable choice because in many cases the 4004 would do 8-bit arithmetic faster than the ... or multiprecision arithmetic, faster than the 8008 could do, because the 8008 was a pretty slow unit.

But customers wanted more processing power and it turned out that it was kind of short of registers. So, it was ... even with our interrupt scheme, it was difficult to really handle an interrupt. You couldnít... you had to reserve registers, it didnít have the capability for saving the state of the machine easily, and one day Federico Faggin came around and said that Intel had this new n-channel process and that he was going to lay out the 8008 in n-channel. And I asked him if it was just like a, you know, a simple, re-photographing or whether it required a new layout.

No, he said, it was going to require a totally new layout, all different design rules. And so I said ëWell how about if we fix some of these problems weíre hearing from our customers?í and so he was receptive fortunately to that. And so, really Stan and Federico and I all contributed to what ultimately became the 8080. And I consider that the first microprocessor that *really* had performance comparable to a minicomputer and I still think of it as one of the first really successful microprocessors... I had lots of second-sources, in fact, I heard even that the Russians built a copy.

RW: OK, letís stop for a moment.

[Pause.]

RW: Well, certainly you became officially known as ëthe guy,í the guy that invented the microprocessor, and you were in *Fortune* magazine and so on and so forth. And yet, Federico Faggin and Stan Mazur were also involved. Was there any hard feelings from their standpoint?

TH: Well, I know Federico has had hard feelings. One thing... Iíve tried, most of the times, when I have been recognized, to give credit for both the role that Stan and Federico made. I mean Federico, thereís no question, he did a fantastic job of layout and nobody should take that away from him.

However, I think if you look at some of the time frame, I believe originally it was probably Gordon Moore who was responding to *Fortune* magazine... You know, ëhowíd this thing get started?í And it was started by me, before either Federico or Stan was an employee of Intel. Stan joined in September of 1969. This thing got started by me before that time frame and, in fact, it was like the 16th of September when we made essentially a formal proposal to Japan. But the work had been going on, essentially, from about the beginning of July and it was throughout much of late-July and throughout August that I was working on this basic idea and the instruction set and so on for the 4004.

Now, it was difficult however, working alone and especially ... I had the Japanese engineers who were *hostile* to what I was doing so it was wonderful when Stan joined the group--which now was two people instead of one--and began to make contributions. And Stan did make a lot of contributions to the basic ideas for the processor. But, I think, why I got singled out in this by the people at Intel was because I really did start the thing before either of these other people joined.

RW: Sure...

TH: Federico didnít join until the following April. On the other hand, Federico *did* do the detailed circuit design and the full logic design and layout for the part... he did *a lot* of work and he deserves the credit for that. But, as far as the architecture... which I feel... the architecture is a crucial factor in getting the transistor count down to where it would be *compatible* with the process of the day. And in fact, that part of it... you know people were projecting something like thousands and thousands of gates that would be required. Now figuring something in the order of three or four transistors per gate, youíd be talking 20 or 30,000 transistors for many of the architectures that might have been considered.

People were looking at using LSI for *aircraft computers* and they were talking about doing it in LSI but maybe twenty or thirty chips... so the idea of getting it on... an architecture that would get it to *one* chip was, I believe, a crucial factor and I still believe that thatís one of the simplest architectures around. The actual transistor count ... when you count the physical transistors on the die, I believe itís a little over 2,100 ... I think something like 2108. When Federico counted the transistors, I think he counted it as 2,300 but he said in that he had some regular arrays and he was counting transistors that he didnít actually implement--sort of like a read-only memory chunk in which he counted all the *sites* as a transistor even if a transistor wasnít placed there.

But I mean this was a very small number. One other point on this is that the basic definition of what became the 8008--its instruction set, its register complement and so on--that device ended up, I believe around 3,400 transistors and, as far as I know, the TI device was designed from that same spec. In other words, as far as I know, there was no flow of *layout* information, or logic detail from Intel to TI. It was just an overall architectural viewpoint ... bus specifications and things like that that went to them. I believe that their transistor count ended up just a little over 3,000 as well and there were a few features that we had added to that original instruction set, for example, making the RESTART a CALL.

So, the architecture is really what determines the number of transistors and I feel *thatís* what really made it possible to do the device at that time.

RW: Well, and I think most people would say that the architecture is the machine. Because it can be implemented in various technologies and thatís sort of a turn-the-crank kind of thing...

TH: Yeah.

RW: ...that the real definition of a computer is the architecture and instruction set.

TH: And along those lines, I might mention thereís a situation with a patent granted to a fella named Gilbert Hyatt. Now he filed nine months after this IEEE article appeared [holding up previously-mentioned article]. He had a processor that he was describing built with TTL... Iíve seen the patent application, the original application, the processor was actually working with a core memory but he said it could have been replaced with semiconductor memory. There is nothing in the disclosure of the patent that says anything about single-chip technology. The patent had something like 140 claims and *one* claim in the patent application said ëI claim this processor implemented as a single chip,í which would be reasonable in the light of the article that had been published nine months before.

Now, the only problem is that if you look at the design thatís published there and go back and do a transistor count--and I made that effort--using what I consider a very conservative design effort in which every gate was a complex gate and that way you minimize the transistor count, I still came up with something like almost 8,000 transistors for the processor. But then Hyattís patent claim says that the memory is *also* going to be on that same chip so if you make *that* assumption and the type of memory that was available then, if it was read-write memory would be... was six transistors per bit. So in effect... and that would have been something like 32,000 bits of memory I believe was the number. So it would have been a very, very large amount.

RW: It would have been impossible.

TH: Yeah. And even... I believe even the seven thousand or eight thousand transistors for the processor would have put it out of the realm of possibility for this time frame...

RW: It would have been a number of years before that would have been practical...

TH: Yes. But that probably would not... and itís not realized by the patent office, probably nobody in it would appreciate it...

RW: Whatís happening with that now?

TH: As far as I know, that patent has been asserted and royalties are being paid. In fact, not long ago, I was looking at a small notebook computer and happened to notice that it said on the computer itself, it said ëlicensed under the following patentsí and there was a list of patents. So I looked up the patents and they were all Hyatt patents, and including the one that supposedly covers the microprocessor.

RW: So...

TH: The other thing thatís also interesting, some of them covered the idea of refreshing a dynamic RAM. It was actually put in... and whatís interesting, thereís not a word about dynamic RAM. And in fact, it seems fairly obvious from the wording in the original patent application that when it was filed, Hyatt didnít know that dynamic RAM existed. He talks about read-only memory or flip-flop memory and dynamic RAM is neither.

RW: So, have you ever looked up this attorney that wouldnít patent your design?

[Laughing.]

TH: There have been several times when Iíve almost crossed paths with him. Iím not sure he would admit at this point to his role in the matter...

RW: Well, you went on to do a bipolar bit-slice...

TH: Yes.

RW: and that was the first bit-slice...

TH: Well, I donít know...

RW: ... AMDís followed.

TH: OK, I didnít know that. We attempted to do a bit-slice and Intel did not have a strong position in its bipolar technology at that time. We had an early start with the Schottky process but it really didnít develop and progress as much as the MOS processes did.

So, in looking at it, we felt... in fact, we never intended to offer a bit-slice. In fact, we considered that the worst nightmare would be to try to support a microprogrammed architecture. Because then, in effect, youíre trying to support everybody doing everything. In fact, we had an architecture, an actual processor defined, that would have essentially directly implemented the code that was put out by Intelís PL/M compiler and *thatís* what we thought we would build with our bipolar family. We had optimized the design for that particular processor and I guess there must have been a recession sometime around 1974, or about the time this was coming out, and Intel decided no way did it want to support another processor family and so the chips were designed and ... but there was no intention to do an upgrade. Because the idea was this would have been an upgrade path... if they 8080 wasnít powerful enough for you, then you go to the bipolar processor. So that, it was decided, would not be announced. And then we... Now we were in that nightmare scenario: what do we *do* with these chips?

And so they were announced. In fact, I had written a... what I considered a crude assembler for the microcode so that I could, and the engineers working for me, could generate microcode for this family. But it was not a clean layout. I mean, I threw this thing together and it worked fine for us but I felt that before it was offered as a product, someone should clean it up. There was a software development group at Intel that was supporting the microprocessors and so we sent the product over there... or this assembler.

Well, they decided that they would not only re-write it but they would totally change the assembly language. So now we were in the situation where we had the product on the market and we didnít even have an assembler for it. I mean, we had one working in-house but what they published as an assembly language was not what we could run.

And of course, they were months late on getting the product out, so that did not help. And of course people came out with 4-bit slices... we figured about the best we could yield was a 2-bit slice. We *did* try to compensate by putting some of the interfaces that you do from that 2-bit slice... you know, the bus interfaces so we could have the address bus and the data bus from that same slice. Where in some cases with the 4-bit slice, youíd need to have separate chips outside that would buffer an address for an address bus. But, generally, one of the things I found: you canít really tell the customer what he doesnít want to hear. And he just knows that a 4-bit slice is better than a 2-bit slice, even if it needs a bunch of logic to go around it to accomplish the same function!

RW: Well, also at that time, Intel was exiting the bipolar business. There werenít putting the R&D into bipolar.

TH: ...Our major effort was in n-MOS.

RW: Correctly so, ... it [bipolar] was sort of a lame duck.

TH: Yeah.

RW: So you left Intel in what, ë83?

TH: Well I had a sort of very different career at Intel for some years. In other words, I really got out of the microprocessor game in about ë75 and started with telecom activity. And I feel fairly... quite proud of that activity because as far as I know, we produced the first commercially-available monolithic CODEC...

RW: Yes.

TH: And our group, with a little help from Paul Grey from Berkeley who came down and spent a sabbatical with us, we had I believe the first commercially-available switched capacitor filter, thatís an important product that supports the CODEC. And then we had another product which... it wasnít the right product at the right time but it was still, I think, a pioneering device, a digital signal processing chip. In this case, it had analog I/O on it, we called it the 2920, and we were looking to try to do things like DTMF receivers, and MODEMs and so on with this device. But as I say, it wasnít a successful product. In this case, the ASIC implemented with, you know, switched capacitor techniques seemed to be more attractive to the people. We never found the right market for the product that had a small enough volume to justify a programmable part. Most of the things like MODEMs ended up being high-volume things where they were well able to go and do custom or semi-custom logic for.

RW: Yeah, and of course in those days telecom was primarily copper wire was it not?

TH: Well there was copper wire but there was a developing interest in switching and the use of ... well the CODEC was originally so that you could multiplex conversations over a twisted-pair wire which originally would have been installed so itíd carry one conversation, then by digitizing and sending over that same wire, youíd go in and at each of the places where there were loading coils youíd replace those with repeaters to handle the digital signal and re-establish it, and now you put 24 conversations on that same wire pair.

So, thatís where the CODEC was to be used originally. And I had the idea that it could also be used in a switching environment, in other words, in a PABX environment where the signal coming in from the telephone is digitized and then you do your switching digitally and then send it...

RW: Which, of course, is how itís done today.

TH: Yes. And so, designed into our first CODEC was some of the control that one could use to do that kind of switching. Then, as you ask, I left Intel in the beginning of 1983. I mean Iíd been there for 14 years ... which I think is a long time in this valley... and Atari had made a very interesting offer. They were working on a wide range of very interesting things. I mean, some having to do with video games, some having to do with home computers and in other applications of computers. They were looking at things like picture telephones and ... one of their gadgets was a... they called it a ësports meterí for runners and joggers and so on--making use of Doppler sonar--and a whole range of looking at the ways you interact with a computer and trying to get data in more quickly or mechanical reactions... things even like... for things like driving simulators to get the feeling of the wheel and so on.

So there was a lot of engineering going on in that area and Alan Kay was at Atari and in this case, I would be working fairly closely with Alan Kay and some of these new R&D ideas and new ways to look at computer applications and that sounded like it could be a very, very interesting field to be in. The only difficulty was that, I think at the time I joined Atari, I donít think anybody realized how much trouble the company was in. And over the following year, that just magnified.

One of the things I had remembered about Intel was that when we came out with a microprocessor, there was a major concern, which at first I didnít understand but finally understood: a lot of microprocessors were sold through distribution. And one of the problems was that distributors were kind of lax about getting information about sales back. You didnít know whether the product went out the door or whether it was sitting on the distributorsí shelves. So there were some major efforts being made to improve the feedback about whatís actually happening out there under ëdistributorís inventory.í

RW: And also, not to actually count it as a sale until a distributor sold it...

TH: Yeah, until itís in a customerís hands...

RW: Right.

TH: Well, one of the things I found out was the attitude of Atari was since most of their products were sold through distribution, there was *no way* that you could have knowledge of what the final sales were. So essentially they washed their hands of it and *that* came back to haunt very badly because apparently there were a lot of games and so on, out there on distributorsí shelves that were *not* moving and that the company didnít know about and assumed that they had sold and they were making more. And so that was one aspect of...

RW: The company collapsed really rapidly.

TH: It ... the other thing was the market had probably just saturated. I mean, there was, thereís probably a certain number of people or families that have--I guess the main market were teenage boys between what, a certain range of ages--which were the primary customer, and at some point every one of those families had, you know, the video game equipment. And unless the next generation offered a significant improvement, nothing much was going to... you know, they werenít going to buy another one.

And so in some sense, I think, they continued to project the same growth, you know, sort of neglecting the fact that the market was... they hadnít bothered to stop and count the number of kids that really *were* out there.

RW: Well we did that with watches and calculators. Traditionally we just keep extending that growth... which doesnít happen.

TH: Itís nice once you put the straight line on the chart just to project it out there... you like the way the numbers look!

RW: ...On log paper!

TH: Yeah. So, I think that was a factor and so when it did start to shrink, it was a very painful time... and necessitated a lot of cutbacks and under those circumstances, the future-thinking... R&D and, you know, these activities that Allan Kay was associated with and that I was to be associated with, were really not considered high priority anymore.

I was given responsibility for some of the more immediate tasks to be done at the company. But even those were seen as a problem. And as I heard, the concerns began to grow at the parent company, Warner, and so finally they decided to sell the company and it was purchased by Jack Tramiel who had been associated with Commodore.

And, in fact, I was away in New York when the word came that the company had been sold. In fact, Iíd heard rumors about it before... I even had spoken to some of the people there, you know, and the management, and they sort of you know, pooh-poohed the whole thing. I donít think anybody really knew that this was happening... except back at Warner [laughs.]

And so anyway, when I came back, I met with Jack Tramiel and who was very pleasant but, essentially, we pretty much agreed that what he wanted to do and what my interests were were probably not very close. And so, at that point, they just bought out my contract and I became independent.

Shortly after that, maybe it was about a year, Gary Summers who had been in charge of Atari Semiconductor and who also left about that same time, decided to start a company in the ...in fact, he called it ìTECLICONî for TEChnical LItigation CONsultants... in which the idea is to provide technical expertise to people who are involved in patent litigation or patent prosecution. And it turns out that while the attorneys are very sharp, thereís a limit to how much of the technology they can follow because they have so many *legal* aspects that they have to worry about.

So we found this niche and I found it fascinating because I often get to ... sort of like playing detective, in other words, digging through and looking for ëprior artí and the experience Iíve had helps me a great deal in this. In other words, knowing where to look for a piece of prior art or what type of technology *might* be using something thatís relating to a topic thatíll be at issue in a patent trial.

RW: Well did it ever come out that Babbage had really ...was really the inventor?

[Laughter.]

TH: I donít think weíve gone back that far.

RW: OK, well Ted, thanks a lot. Weíve got to wrap it up here. But itís been great fun talking with you.

[End.]

[End of Interview: 1:19:18]

[Word Count: 10 418]